

Unidade II

5 SÍNTESE DE CORES

A síntese de cores em computação gráfica é o processo de combinar diferentes cores para criar uma imagem ou cena visual. Existem dois principais métodos de síntese de cores: aditivo e subtrativo.

A síntese de cores aditiva é usada em dispositivos que emitem luz, como monitores de computador, televisores e projetores. Nesse método, as cores primárias (vermelho, verde e azul) são combinadas em diferentes intensidades para criar uma ampla gama de cores. Por exemplo, se o vermelho e o verde são combinados em intensidades iguais, surge o amarelo. Se as três cores primárias são combinadas em intensidades iguais, surge o branco.

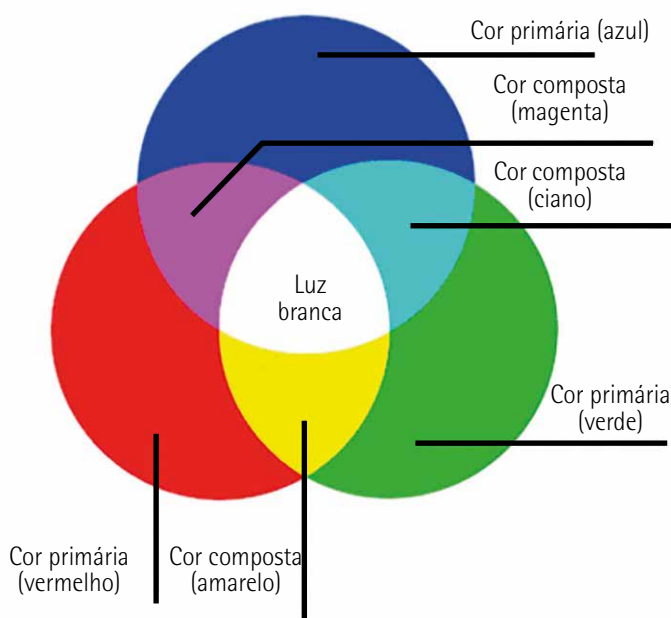


Figura 39

Por outro lado, a síntese de cores subtrativa é usada em dispositivos que refletem a luz, como impressoras e tintas. Nesse método, as cores primárias (ciano, magenta e amarelo) são combinadas em diferentes quantidades para criar uma ampla gama de cores. Por exemplo, se o ciano e o magenta são combinados em intensidades iguais, eles criam a cor azul. Se todas as três cores primárias são combinadas em quantidades iguais, elas formam o preto.

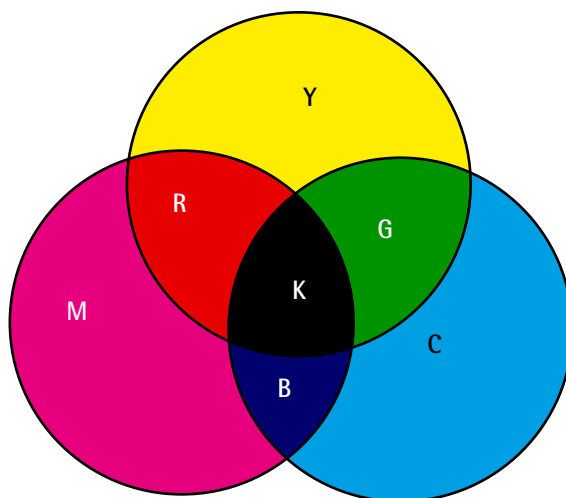


Figura 40

Na síntese de cores em computação gráfica, as cores são representadas por valores numéricos de intensidade de cada canal de cor, que podem ser combinados em diferentes proporções para formar as cores desejadas. As cores também podem ser manipuladas usando técnicas como ajuste de brilho e contraste, aplicação de filtros e efeitos, e a aplicação de texturas e materiais a objetos em uma cena 3D.

5.1 Luz e cores

O sistema de luz e cores em computação gráfica é responsável por criar a aparência visual de objetos e cenas em uma imagem digital. Existem diferentes modelos de cor e sistemas de iluminação que podem ser utilizados para simular a interação da luz com os objetos.

O modelo de cores mais comum é o RGB (Red, Green, Blue), que utiliza as cores primárias de luz para representar todas as outras cores. Cada pixel em uma imagem é composto de uma combinação de valores de intensidade dessas três cores, que variam de 0 a 255. Por exemplo, um pixel com valor de intensidade (255,0,0) representa a cor vermelha pura, enquanto um pixel com valor de intensidade (0,255,0) indica a cor verde pura.

Outro modelo de cores comum é o CMYK (Ciano, Magenta, Amarelo, Preto), aplicado principalmente para impressão. Nele as cores primárias são as cores secundárias do modelo RGB. Para criar a cor preta, é adicionado um canal separado de intensidade de tinta preta, já que a combinação completa das três cores primárias não resulta em preto puro.

O sistema de iluminação mais usado em computação gráfica é o modelo de iluminação de Phong, que simula a iluminação de objetos em uma cena 3D. Esse modelo leva em consideração três componentes principais de luz: ambiente, difusa e especular. A luz ambiente representa a luz que é refletida em todas as direções e é geralmente usada para simular a iluminação de fundo em uma cena. A luz difusa

expressa a luz que é refletida em todas as direções de maneira uniforme, e a luz especular indica a luz que é refletida de maneira mais concentrada, criando reflexos brilhantes em superfícies lisas.

Os objetos em uma cena 3D são modelados com materiais que podem ser atribuídos a diferentes valores de reflexão e absorção de luz, bem como texturas que podem ser mapeadas sobre a superfície dos objetos para criar efeitos visuais mais complexos. A combinação de modelos de cores e sistemas de iluminação permite a criação de imagens digitais realistas e complexas em computação gráfica.

A seguir uma cena com adição de iluminação e reflexões.



Figura 41

Disponível em: <https://bit.ly/3meNTT0>. Acesso em: 30 mar. 2023.

5.2 Luz, cores e o sistema visual humano

A luz e as cores são percebidas pelo sistema visual humano através dos olhos, que funcionam como uma espécie de câmera que capta a luz que entra no globo ocular e a transforma em sinais elétricos que são interpretados pelo cérebro.

O olho humano é composto de várias estruturas que trabalham juntas para permitir a visão. A córnea é a primeira estrutura que a luz atravessa ao entrar no olho, e é responsável por focar a luz na retina, que é a camada sensível à luz na parte de trás do olho. A retina contém células fotossensíveis chamadas de cones e bastonetes, que são responsáveis por detectar a luz e transmitir sinais elétricos para o cérebro através do nervo óptico.

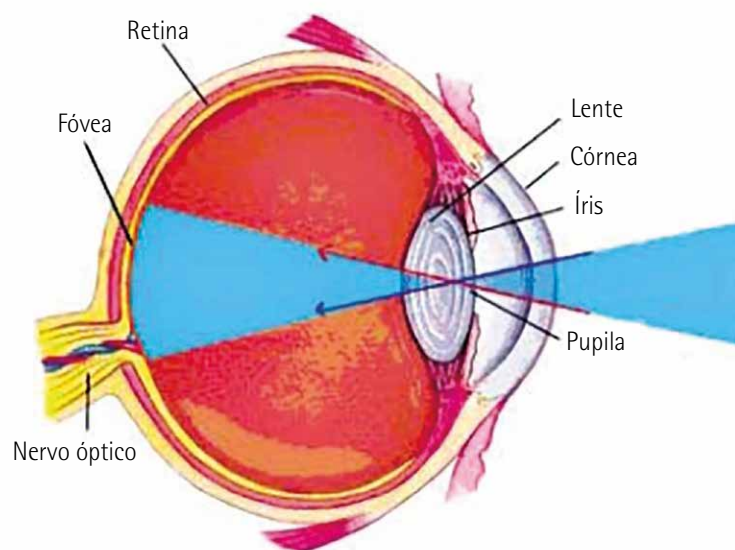


Figura 42 – O sistema visual humano

Fonte: Azevedo e Conci (2003a, p. 183).

Os cones são responsáveis por detectar as cores, enquanto os bastonetes detectam a luz em condições de baixa luminosidade. Existem três tipos de cones no olho humano, que são sensíveis às cores vermelho, verde e azul, respectivamente. A combinação de sinais dos cones permite que o cérebro interprete uma ampla gama de cores.

O cérebro processa as informações recebidas dos olhos e cria uma percepção da cor com base nos sinais recebidos dos cones. Além disso, o sistema visual humano é capaz de perceber a luminosidade e o contraste, que são interpretados com base nos sinais recebidos dos bastonetes.

A percepção da luz e das cores pelo sistema visual humano é resultado da interação entre a luz que entra no olho, as estruturas do olho que a transformam em sinais elétricos e o processamento desses sinais pelo cérebro. Esse processo complexo é fundamental para a percepção do mundo ao nosso redor e é essencial para a compreensão e apreciação de imagens e obras de arte.

Aspectos técnicos sobre cores e luz

Existem diversos modelos matemáticos para representar cores e luz em computação gráfica. Um dos mais utilizados é o modelo RGB (Red, Green, Blue), que foi mencionado anteriormente. Nesse modelo, as cores são representadas por um conjunto de três valores que indicam as intensidades das componentes de luz vermelha, verde e azul em um determinado pixel de uma imagem.

Para criar cores adicionais, é possível combinar as intensidades das três cores primárias (vermelho, verde e azul) em diferentes proporções. Por exemplo, a cor amarela pode ser criada com uma combinação

de intensidades de vermelho e verde, enquanto a cor roxa pode ser criada com uma combinação de intensidades de vermelho e azul.

Outro modelo matemático comumente utilizado em computação gráfica é o modelo de cores HSL (Hue, Saturation, Lightness), que separa as cores em três componentes diferentes: matiz, saturação e luminosidade. A matiz (Hue) é a característica que define a cor em si; a saturação indica a pureza da cor; já a luminosidade (Lightness) destaca a quantidade de luz presente na cor.

Em relação à representação da luz, um modelo matemático comum é o modelo de iluminação de Phong, que leva em consideração a luz ambiente, a difusa e a especular para simular a iluminação de objetos em uma cena 3D. Esse modelo utiliza equações matemáticas para calcular as intensidades de luz que são refletidas pelos objetos com base em suas propriedades físicas, como reflexão e absorção de luz.

Além disso, existem modelos matemáticos para simular outros efeitos de luz, como sombras e reflexos. A criação de imagens realistas em computação gráfica envolve a aplicação de diferentes modelos matemáticos para representar a luz e as cores de maneira precisa e realista.

5.3 Sistema de cores aditivas

É um modelo usado em computação gráfica e outras áreas, como a fotografia e a televisão. Nesse sistema, as cores são criadas pela adição de luz de diferentes cores primárias em um fundo escuro ou preto.

As cores primárias no sistema aditivo são vermelho, verde e azul, conhecidas como RGB (Red, Green, Blue). Cada uma dessas cores é representada por um valor que varia de 0 a 255, indicando a intensidade da luz em cada componente de cor. Quando essas cores são combinadas, elas criam outras cores, como amarelo, ciano e magenta.

A adição de luz das três cores primárias em iguais proporções resulta em uma luz branca. Isso pode ser facilmente observado em monitores de computador ou televisores, em que a imagem é criada a partir da combinação de luz emitida pelos pixels de RGB.

O sistema de cores aditivas é oposto ao sistema de cores subtrativas, que é utilizado em impressão de documentos e imagens. No sistema subtrativo, as cores primárias são ciano, magenta e amarelo, e a combinação dessas cores resulta em uma cor preta.

Em resumo, o sistema de cores aditivas é um modelo utilizado em computação gráfica e outras áreas para representar cores a partir da adição de luz de diferentes cores primárias, como vermelho, verde e azul.

Os exemplos de cores aditivas mais comuns são:

- **Vermelho:** é uma cor primária no sistema de cores aditivas. Pode ser criado através da combinação das cores vermelha e verde, com intensidades máximas, resultando em uma luz amarela que pode ser percebida como vermelho.

- **Verde:** também é uma cor primária no sistema de cores aditivas. Pode ser criado através da combinação das cores verde e vermelha, com intensidades máximas, resultando em uma luz amarela que pode ser percebida como verde.
- **Azul:** é a última cor primária no sistema de cores aditivas. Pode ser criado através da combinação das cores azul e verde, com intensidades máximas, resultando em uma luz ciano que pode ser percebida como azul.

Outros exemplos de cores aditivas incluem:

- **Amarelo:** pode ser criado através da combinação das cores vermelha e verde, com intensidades iguais ou próximas.
- **Ciano:** pode ser criado através da combinação das cores verde e azul, com intensidades iguais ou próximas.
- **Magenta:** pode ser criado através da combinação das cores vermelha e azul, com intensidades iguais ou próximas.
- **Branco:** pode ser criado através da combinação das três cores primárias em igual intensidade.

Diversas outras cores podem ser criadas através da combinação de diferentes proporções das três cores primárias.

5.4 O sistema RGB (Red, Green, Blue)

É um modelo de cores aditivas utilizado em computação gráfica e as cores são criadas a partir da combinação de luz de diferentes intensidades das cores primárias vermelho, verde e azul. Quando essas cores são combinadas em variadas proporções, elas criam uma ampla gama de cores.



Observação

Cada cor é representada por um valor numérico que varia de 0 a 255 para cada componente. Assim, o vermelho é (255, 0, 0), o verde (0, 255, 0) e o azul (0, 0, 255).

O sistema RGB é amplamente aplicado em monitores de computador, televisores, câmeras digitais e outros dispositivos eletrônicos que usam luz para produzir imagens coloridas. As imagens digitais são armazenadas como uma grade de pixels, em que cada pixel é composto de três valores RGB, os quais representam a cor daquele pixel.

Em resumo, o sistema RGB é um modelo de cores aditivas utilizado em computação gráfica para representar cores através da combinação de luz de diferentes intensidades das cores primárias vermelho, verde e azul.

5.5 Sistemas de cores subtrativas

Trata-se de modelos de cores que representam as cores através da subtração de cores de luz branca ou de uma fonte de luz. Diferentemente do sistema de cores aditivas, em que as cores são criadas através da adição de luz, no sistema subtrativo as cores são criadas pela subtração de cores de luz.

Um exemplo comum de sistema de cores subtrativas é o CMYK (Ciano, Magenta, Amarelo e Preto), usado na impressão de documentos e imagens. Nesse sistema, as cores primárias são ciano, magenta e amarelo, e a combinação dessas cores resulta em uma cor preta. No entanto, como a combinação de cores subtrativas nem sempre é perfeita, uma cor preta adicional é adicionada para garantir uma cor preta mais sólida.

Outro sistema de cores subtrativas aplicado em computação gráfica é o sistema RYB (Vermelho, Amarelo e Azul), que é comumente utilizado em artes visuais. Nesse sistema, as cores primárias são vermelho, amarelo e azul, e a combinação dessas cores cria outras cores.

Embora o sistema RGB seja o mais comum em computação gráfica, o CMYK é amplamente aplicado para a impressão de documentos e imagens, enquanto o sistema RYB é usado em artes visuais.

Os exemplos mais comuns de cores subtrativas são:

- **Ciano:** é uma das três cores primárias do sistema CMYK, com magenta e amarelo. É produzido pela subtração de vermelho da luz branca, e é frequentemente usado como cor base na impressão colorida.
- **Magenta:** também é uma das cores primárias do sistema CMYK. É produzido pela subtração de verde da luz branca e é comumente usado na impressão de tons de pele e em outros elementos gráficos.
- **Amarelo:** a terceira cor primária do sistema CMYK. É produzido pela subtração de azul da luz branca e é usualmente usado na impressão de tons de verde e marrom.
- **Preto:** é uma cor não primária do sistema CMYK, adicionada para melhorar a qualidade da impressão. É feito pela subtração de todas as três cores primárias (ciano, magenta e amarelo) da luz branca.
- **Marrom:** é uma cor que pode ser criada pela combinação de ciano, magenta e amarelo em proporções adequadas.

- **Roxo:** pode ser criado pela combinação de ciano e magenta em proporções adequadas.
- **Laranja:** pode ser criado pela combinação de amarelo e magenta em proporções adequadas.

Essas cores são usadas principalmente em processos de impressão devido a sua capacidade de absorver a luz para criar cores mais escuras.

CMYK (Ciano, Magenta, Amarelo e Preto)

É um sistema subtrativo de cores utilizado principalmente em impressão. As cores primárias ciano, magenta e amarelo são combinadas para criar outras cores. Quando as três cores são combinadas em proporções iguais, elas formam uma cor escura, geralmente marrom ou preto, devido à subtração de luz branca.

O modelo CMYK é uma variação do modelo CMY, com a adição de um quarto componente, o preto (K). O preto é adicionado ao modelo CMYK para melhorar a qualidade da impressão, pois a combinação completa das cores ciano, magenta e amarelo para criar o preto pode levar a cores escuras impraticáveis e a uma qualidade de impressão mais baixa.

Assim, o modelo CMYK é frequentemente utilizado em impressoras e outros dispositivos de impressão para a criação de documentos e imagens coloridas. No modelo CMYK, as cores são representadas por valores numéricos que variam de 0 a 100 em cada componente de cor. Por exemplo, uma cor ciano puro seria representada como (100, 0, 0, 0); a cor magenta pura, como (0, 100, 0, 0); a cor amarela pura, como (0, 0, 100, 0); a cor preta pura, como (0, 0, 0, 100).



Lembrete

O modelo CMYK adiciona um componente preto para melhorar a qualidade da impressão, e as cores são representadas por valores numéricos que variam de 0 a 100 em cada componente de cor.

5.6 Profundidade de cor (bit por pixel)

As imagens digitais são representadas na forma de matrizes de pixels, e a cada pixel está associado a uma cor.

Denomina-se profundidade de cor (color depth) a quantidade de bits necessários para se representar um certo conjunto de cores. A profundidade de cor é, normalmente, estabelecida em bits por pixels (bpp).

Por exemplo, para representar uma imagem em branco e preto (duas cores), é necessário apenas um bit para cada pixel ($2^1 = 2$): bit = 0 => pixel apagado ou bit = 1 => pixel aceso. Assim, cada byte (8 bits) de memória pode armazenar 8 pixels. Considerando uma imagem de 320 x 200 pixels nesse sistema, vemos que a memória ocupada será de $320 \times 200 \times 1 = 64000$ bits = 8000 bytes (8 kbytes).

Por outro lado, com um único byte podemos representar $2^8 = 256$ cores. Nesse caso, as cores que serão representadas dependerão do hardware ou de uma paleta de cores armazenada com o arquivo. Imagens assim são chamadas de indexadas.

Vimos que o sistema de estabelecer cores para cada pixel é o da síntese de cores a partir de um conjunto de cores primárias aditivas e subtrativas.

Tanto o sistema aditivo RGB como o CMY possuem, cada um, três canais de cores primárias (tricromia).

Aqui temos a profundidade de 8 bpp (grayscale), que emprega apenas um canal, totalizando 256 tons de cinzas, 16 bpp (highcolor), que permite a representação de 65536 cores através de uma síntese que utiliza 5 bits para o canal da cor vermelha (R), 6 bits para o da cor verde (G) e 5 bits para representar o azul (B). Também há o sistema de 24 bits por pixel (truecolor), que emprega um byte para cada canal, totalizando $2^8 \times 2^8 \times 2^8 = 2^{24} = 16.777.216$ cores.

Formatos de arquivos de imagens

Uma vez atribuída a propriedade cor a cada um dos pixels da matriz que forma uma imagem digital, esta é armazenada em formatos específicos, identificados por suas extensões. Por exemplo: BMP, JPG, TIF, PNG, PCX, PIX, DDS, FIT, GIF, PCD, TGA, RAW, RLE, WMF, CDR, PDF, PS, SVG, AVI.

Cada formato possui uma utilização específica, seja ela própria de um aplicativo (formatos proprietários), como o PSD, CDR, DDS, de um ambiente operacional, como o BMP, GIF, JPG, seja ela de uma mídia, como o PCD, AVI, MPG, WMV.

Os formatos que são próprios de um certo ambiente acabam por assumir características de uso geral e, alguns, tornam-se muito populares, como o GIF e o JPG, que se tornaram padrão na internet.

Alguns armazenam dados de imagens na forma matricial (pixels), como BMP, GIF, TIF, JPG, e outros na forma de vetores (vértices, arestas e faces), como WMF, CDR, SVG. Já outros na forma de streamers (sequências de imagens), como AVI, MPG, MOV etc.

Outra característica muito importante a ser destacada é a forma de armazenagem, profundidade da informação (cores, luminosidade) em bits, por exemplo, 2, 8, 16, 24, 32, 48, 64, 98-bits..., e a compressão, com perda de informação (lossy) ou sem perda de informação (lossless).

Por exemplo, arquivos BMP (bitmap: mapa de bits), em geral, são arquivos de imagens diretas, mapas de bits na memória, armazenadas sem qualquer perda de qualidade (resolução) de cores ou da distribuição espacial da imagem. Já os arquivos de tipo GIF e JPEG são formatos compactos obtidos a partir de transformações que reduzem a quantidade de informação em favor de um arquivo menor.

No caso dos arquivos GIF, as cores são **indexadas**, representadas na forma de uma tabela de algumas poucas cores (paleta), e cada pixel guarda referência a uma cor da tabela. Trata-se de um arquivo compacto com perda da qualidade de cor.

Os arquivos JPEG, por outro lado, preservam a profundidade de cor de uma certa maneira, mas seu algoritmo de compressão por blocos, usando transformadas discretas de cossenos (DCT – Discrete Cosine Transform), substitui a representação por pixels por uma de coeficientes de uma série matemática truncada. Esse truncamento produz uma distorção na imagem que a faz perder qualidade espacial.

Apesar de perda de qualidade, tanto GIF e JPEG são preferíveis para uso na internet por causa do tamanho reduzido de seus arquivos, que economiza em transmissão e o armazenamento em disco. Entretanto, ao representarmos as imagens que armazenam no dispositivo gráfico de saída, elas são decodificadas (decode) para o formato de mapa de memória, ou bitmap, cujo tamanho depende da resolução espacial da matriz de pixel e da profundidade de pixel. Isso também vale para o caso quando o arquivo de imagem é vetorial, já que a saída gráfica mais comum é feita em um dispositivo raster (monitor, impressora matricial).

Como dissemos anteriormente, a imagem digital colorida é codificada em um certo número de canais (cores primárias ou filtros), de acordo com especificações próprias de cada modelo. Os modelos de cores mais comuns são: RGB, CMY, XYZ, HSB, OSA, HVC, HSV, HSL, YIQ, OPP. Destaca-se que existem variações (LRGB e CMYK) e outros modelos com aplicações específicas (modelos de cores falsas).

Basicamente, os modelos dividem-se em dois tipos de sistema: **aditivo** e **subtrativo**.

O principal sistema aditivo é o RGB, usado nos monitores de vídeo, cujas cores primárias são vermelho (R), verde (G) e azul (B). Esse é o mesmo modelo usado pelo olho humano para enviar sinais ao cérebro, que sintetizará a imagem cromática por meio das células da retina, denominadas cones (cones R, G e B). Nos sistemas digitais, ao contrário do olho, que é analógico, cada canal (R, G e B) possui 256 níveis de intensidade (8 bits). Na verdade, o olho humano apresenta mais um canal, que podemos codificar como L (luminância), que é processado por células da retina denominadas bastonetes.

A luminância, por sua vez, também é um canal de um modelo de síntese de cores denominado HLS (Hue, Lightness, Saturation), ou seja, matiz, luminância e saturação.

Já o sistema subtrativo CMY, usado nas impressoras, se baseia nas cores primárias que filtram certos matizes da luz branca, refletindo as cores ciano (C), magenta (M) e amarela (Y).



Observação

Não confunda as cores primárias do CMY com aquelas que aprendemos nas aulas de arte do Ensino Fundamental; note que o ciano não é azul, e magenta não é vermelho.

O universo de cores que um dado modelo é capaz de representar é denominado espaço de cores e não é completamente equivalente. Ou seja, não é possível destacar o espaço de cores do RGB no modelo CMY, mas o inverso sim, pois o CMY é um subespaço do RGB.

As especificações técnicas do processamento de cores são uns dos assuntos mais complexos em computação gráfica, tema que não será aprofundado neste livro-texto. Agora usaremos apenas os modelos RGB e CMY.

A síntese de cores por esses modelos teóricos pode ser obtida com expressões do tipo:

$Cor = r.R + g.G + b.B$, onde r , g e b são, respectivamente, as quantidades (percentuais) de R , G e B . Considerando valores integrais (0 ou 1), temos:

$$1.R + 1.G + 1.B = \text{branca}$$

$$0.R + 0.G + 0.B = \text{preta}$$

Já no modelo subtrativo CMY, o canal C tem a propriedade de absorver (filtrar, subtrair, zerar) a componente vermelha da luz branca ($R + G + B$); o canal M absorve a componente verde da luz branca; por fim, o Y absorve a componente azul.

$$0.R + 1.G + 1.B = C$$

$$1.R + 0.G + 1.B = M$$

$$1.R + 1.G + 0.B = Y$$

Observe na tabela a seguir as possíveis combinações (teóricas) que transformam um modelo em outro.

Tabela 2

Cor	R	G	B	=>	C	M	Y
Preto	0	0	0	=>	1	1	1
Vermelho	1	0	0	=>	0	1	1
Verde	0	1	0	=>	1	0	1
Azul	0	0	1	=>	1	1	0
Amarelo	1	1	0	=>	0	0	1
Magenta	1	0	1	=>	0	1	0
Ciano	0	1	1	=>	1	0	0
Branco	1	1	1	=>	0	0	0

Um exercício fundamental é descobrir qual é a cor da luz refletida por um determinado objeto quando iluminado por uma luz incidente de dada cor e demais combinações (por exemplo, qual é a cor do objeto? Qual é a cor da luz incidente?). O raciocínio simplista é:

- luz incidente (modelo aditivo);
- objeto pigmentado (modelo subtrativo);
- luz refletida (modelo aditivo).

Para ser verde, por exemplo, um objeto deve ser pintado com a mistura dos pigmentos ciano e amarelo ($G = C + Y$). Enquanto o pigmento ciano absorverá o vermelho da luz incidente, o amarelo absorverá a componente azul, restando à luz refletida o canal verde! Contudo, quando iluminado por uma luz, digamos, de cor vermelha, azul ou magenta, sua coloração aparecerá preta, pois não refletirá cor alguma (lembre-se de que o magenta absorve o verde e que as cores primárias vermelha e azul não possuem o verde). Por outro lado, se a luz for verde, amarela, ciano, a cor aparente será verde.

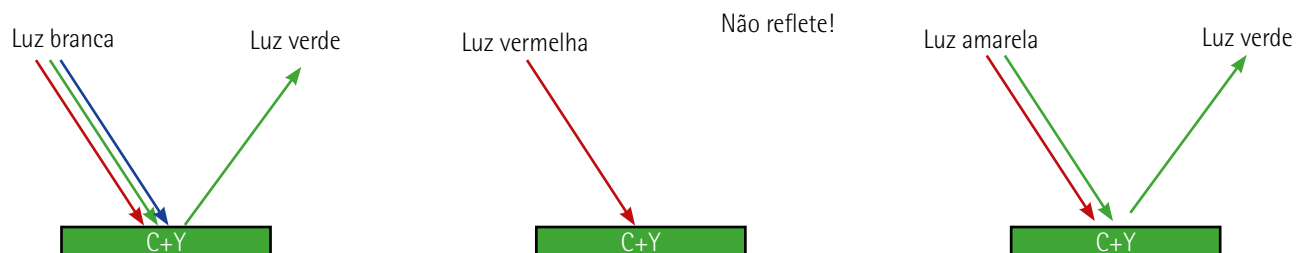


Figura 43



Lembrete

No modelo subtrativo CMY, o ciano absorve a componente vermelha (R) da luz branca; a magenta, a componente verde (G) da luz branca; o amarelo, a componente azul (B).

Exemplos de aplicação

Exemplo 1

Um objeto de cor verde iluminado por uma luz azul refletirá a cor:

- A) Azul.
- B) Vermelha.
- C) Amarela.
- D) Preta.
- E) Branca.

Resolução

Esse problema de lógica pode ser resolvido de várias maneiras. Descreveremos apenas as três mais imediatas.

- Para se pintar um objeto de cor verde, devemos combinar as cores primárias subtrativas (pigmentos) ciano e amarela. Enquanto o ciano tem a propriedade de absorver a luz vermelha, o amarelo absorve a luz azul. Como o objeto é iluminado por uma luz azul, está será absorvida pelo pigmento amarelo contido na mistura do verde, não refletindo nada, portanto, o preto.
- Para percebermos um objeto verde, é necessário que essa componente primária aditiva esteja presente na luz incidente. Como a luz incidente é azul, que é uma cor primária aditiva, não há como refletir o verde, logo não refletirá nada.
- Podemos pensar na relação luz incidente no RGB, cor do objeto no RGB = luz refletida no RGB, como uma operação de multiplicação de arranjo matricial (que não é o mesmo que multiplicação de matriz):

$$\text{Luz incidente} \cdot \text{Cor do objeto} = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0.0 \\ 0.1 \\ 1.0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} = \text{Preto}$$

Assim, deduz-se que a alternativa correta é a D.

Exemplo 2

Qual das opções a seguir contém o número de bits que permite, em um terminal de vídeo colorido, representar uma opção de 1024 cores?

- A) 8 bits.
- B) 6 bits.
- C) 10 bits.
- D) 11 bits.
- E) 12 bits.

Resolução

Lembrando que:

$2^{\text{nº de bits}}$ = quantidade de cores, então:

$$2^x = 1024$$

$$2^x = 2^{10}$$

$$x = 10$$

Conclui-se que a alternativa correta é a C.

Exemplo 3

Qual é a quantidade de memória necessária (em bytes) para representar uma imagem com 65.536 cores por pixel (high color) em resolução de 1280 x 1024 (SXGA)?

Resolução

O sistema de 65.536 cores usa 16 bits por pixel ($2^{16} = 65.536$). Sabendo-se que 16 bits = 2 bytes (cada byte equivale a 8 bits), então esse sistema usa 2 bytes por pixel. Considerando que o dispositivo de saída tem $1280 \times 1024 = 1.310.720$ pixels de resolução, então:

$$1.310.720 \times 2 = 2.621.440 \text{ bytes}$$

Exemplo 4

Highcolor e truecolor são padrões de diferentes profundidades de cores (bits/pixel). O highcolor é capaz de representar 65.536 cores, já o truecolor quase 17 milhões (16.777.216), embora seja mais comum referi-lo como um padrão de 16 milhões de cores. O olho humano é capaz de distinguir até 10 milhões de cores. Qual é a profundidade de cor do padrão truecolor?

A) 4 bits.

B) 8 bits.

C) 16 bits.

D) 24 bits.

E) 32 bits.

Resolução

$$2^x = 16.777.216$$

$$2^x = 2^{24}$$

$$x = 24$$

Assim, a alternativa correta é a D.

Exemplo 5

Um operador de luz de teatro teve um de seus três holofotes de luz primária (RGB) danificado (queimado). Ao iluminar um fundo de cor branca, ele só conseguia fazê-lo refletir duas cores primárias e o ciano. De que cor era a luz do holofote danificado?

- A) Azul.
- B) Vermelha.
- C) Amarela.
- D) Preta.
- E) Branca.

Resolução

Já que um dos holofotes está queimado, o funcionário só conseguirá refletir duas das três cores primárias do RGB.

Considerando as misturas de cores, temos:

$$R + G = Y$$

$$R + B = M$$

$$G + B = C$$

Como o funcionário refletiu o ciano ($G + B$), logo o holofote queimado é da cor vermelha (R).

Conclui-se que a alternativa correta é a B.

6 TRANSFORMAÇÕES GEOMÉTRICAS EM DUAS E TRÊS DIMENSÕES

Transformações geométricas em computação gráfica são operações matemáticas aplicadas sobre objetos geométricos em um espaço bidimensional ou tridimensional, a fim de alterar sua posição, tamanho, orientação ou forma. Essas transformações são realizadas através de matrizes de transformação, que permitem representar uma transformação em termos de uma única matriz.

As transformações geométricas em duas dimensões mais comuns são:

- **Translação:** desloca um objeto em uma determinada direção, especificada por um vetor de translação (t_x, t_y).
- **Rotação:** gira um objeto em torno de um ponto de rotação (c_x, c_y) em um determinado ângulo (θ).

- **Escala:** aumenta ou reduz o tamanho de um objeto ao longo de um determinado fator de escala (s_x, s_y).
- **Cisalhamento:** distorce um objeto em uma determinada direção, deslocando um dos seus lados em relação ao outro.

Já as transformações geométricas em três dimensões mais comuns são:

- **Translação:** desloca um objeto em uma determinada direção, especificada por um vetor de translação (t_x, t_y, t_z).
- **Rotação:** gira um objeto em torno de um eixo específico (x, y ou z) em um determinado ângulo (θ).
- **Escala:** aumenta ou reduz o tamanho de um objeto ao longo de um determinado fator de escala (s_x, s_y, s_z).
- **Cisalhamento:** distorce um objeto em uma determinada direção, deslocando um dos seus lados em relação ao outro.

As transformações geométricas são amplamente usadas em diversas aplicações de computação gráfica, como animação, modelagem 3D, jogos e visualização científica.

A modelagem matemática das transformações geométricas em duas e três dimensões pode ser realizada por meio de matrizes de transformação homogêneas. Essas matrizes são compostas de uma matriz de rotação, uma de escala, uma de cisalhamento e uma matriz de translação, dependendo da transformação específica que se deseja realizar.

Translação em 2D

A matriz de transformação homogênea para a translação em 2D é dada por:

$$\begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix}$$

Figura 44

Onde t_x e t_y são os valores de translação em x e y , respectivamente.

Rotação em 2D

A matriz de transformação homogênea para a rotação em 2D é dada por:

$$\begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Figura 45

Onde θ é o ângulo de rotação em radianos.

Escala em 2D

A matriz de transformação homogênea para a escala em 2D é dada por:

$$\begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Figura 46

Onde s_x e s_y são os fatores de escala em x e y, respectivamente.

Cisalhamento em 2D

A matriz de transformação homogênea para o cisalhamento em 2D é dada por:

$$\begin{bmatrix} 1 & sh_x & 0 \\ sh_y & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Figura 47

Onde sh_x e sh_y são os fatores de cisalhamento em x e y, respectivamente.

Translação em 3D

A matriz de transformação homogênea para a translação em 3D é dada por:

$$\begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Figura 48

Onde t_x , t_y e t_z são os valores de translação em x, y e z, respectivamente.

Rotação em 3D

A matriz de transformação homogênea para a rotação em 3D é dada por:

$$\begin{bmatrix} R_{11} & R_{12} & R_{13} & 0 \\ R_{21} & R_{22} & R_{23} & 0 \\ R_{31} & R_{32} & R_{33} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Figura 49

Onde R_{11} , R_{12} , R_{13} , R_{21} , R_{22} , R_{23} , R_{31} , R_{32} e R_{33} são os elementos da matriz de rotação.

Escala em 3D

A matriz de transformação homogênea para a escala em 3D é dada por:

$$\begin{bmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Figura 50

Onde s_x , s_y e s_z são os fatores de escala em x, y e z, respectivamente.

6.1 Transformações de pontos

A transformação de pontos em computação gráfica refere-se à aplicação de uma matriz de transformação a um conjunto de coordenadas de um ou mais pontos, com o objetivo de realizar uma operação específica, como translação, rotação, escala, cisalhamento, entre outras. Essa técnica é usada para realizar mudanças em objetos gráficos, permitindo a criação de animações, efeitos visuais e outras aplicações em diversas áreas.

A transformação de pontos geralmente é representada por uma matriz de transformação homogênea, que é uma matriz 4x4, em que as três primeiras linhas representam a rotação, a escala e o cisalhamento, e a quarta linha indica a translação. Cada ponto é representado como um vetor de coordenadas homogêneas, que consiste em um vetor 4x1, em que as três primeiras linhas expressam as coordenadas x, y e z, e a última linha é sempre 1.

Ao aplicar a matriz de transformação homogênea em um conjunto de pontos, a matriz multiplica cada vetor de coordenadas homogêneas, resultando em um novo vetor de coordenadas homogêneas transformado. As coordenadas transformadas podem ser convertidas em coordenadas cartesianas normais, dividindo cada componente x, y e z por w, que é a última coordenada homogênea.

Ela é amplamente usada em diversas áreas, como jogos, animação, design gráfico, modelagem 3D, visualização científica, entre outras

Destaca-se que ela é representada matematicamente por meio de matrizes de transformação homogênea. Cada tipo de transformação, como translação, rotação, escala, cisalhamento, entre outros, é indicado por uma matriz específica.

A matriz de transformação homogênea geral para a transformação de pontos é dada por:

$$[X'] [M] [X]$$

$$[Y'] = [M] \times [Y]$$

$$[Z'] [M] [Z]$$

$$[W'] [M] [W]$$

Onde:

X, Y e Z são as coordenadas do ponto original.

X', Y' e Z' são as coordenadas do ponto transformado.

W é a coordenada homogênea do ponto original e W' é a coordenada homogênea do ponto transformado.

M é a matriz de transformação homogênea.

As matrizes de transformação homogênea para cada tipo de transformação são dadas por diversas maneiras, conforme acentuado a seguir:

$$\begin{bmatrix} 1 & 0 & 0 & T_x \\ 0 & 1 & 0 & T_y \\ 0 & 0 & 1 & T_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Figura 51 – Translação

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\theta) & -\sin(\theta) & 0 \\ 0 & \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Figura 52 – Rotação em torno do eixo X

$$\begin{bmatrix} \cos(\theta) & 0 & \sin(\theta) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(\theta) & 0 & \cos(\theta) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Figura 53 – Rotação em torno do eixo Y

$$\begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 & 0 \\ \sin(\theta) & \cos(\theta) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Figura 54 – Rotação em torno do eixo Z

$$\begin{bmatrix} S_x & 0 & 0 & 0 \\ 0 & S_y & 0 & 0 \\ 0 & 0 & S_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Figura 55 – Escala

$$\begin{bmatrix} 1 & Sh_x & 0 & 0 \\ Sh_y & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Figura 56 – Cisalhamento em XY

$$\begin{bmatrix} 1 & 0 & Sh_x & 0 \\ 0 & 1 & 0 & 0 \\ Sh_z & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Figura 57 – Cisalhamento em XZ

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ Sh_x & 1 & 0 & 0 \\ 0 & Sh_y & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Figura 58 – Cisalhamento em YZ



Saiba mais

Para saber mais sobre transformações lineares, consulte:

CALLIOLI, C. A. *Álgebra linear e aplicações*. 6. ed. Santos: Atual, 2009.

6.2 Transformações lineares especiais

Já estudamos as matrizes associadas a transformações lineares, agora vamos entender como obter algumas dessas matrizes através de conceitos da álgebra linear.

Primeiro, deve-se saber a definição de uma transformação linear a partir das imagens de vetores que compõem uma base de seu domínio.

Uma base de um espaço vetorial V é um subconjunto ordenado $B = \{v_1, v_2, \dots, v_n\}$ que satisfaz às condições:

- $V = [B]$ (isto é, B gera V).
- B é linearmente independente – $D \neq 0$

Portanto, a partir da imagem e da base, é possível encontrar a transformação linear.

Reflexão em torno do eixo x

Observando o gráfico a seguir, notamos que ponto A , após sofrer uma reflexão em torno do eixo x , teve como consequência a mudança do sinal da componente y . Supondo que você queira converter esse ponto em um pixel aceso no dispositivo gráfico de saída, é necessário transformá-lo em elemento de uma matriz. Dessa forma, determine a transformação linear e a matriz associada à reflexão em torno do eixo x .

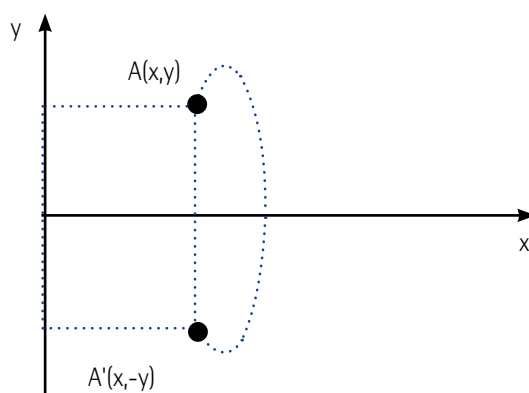


Figura 59

A transformação linear da reflexão é: $T: \mathbb{R}^2 \rightarrow \mathbb{R}^2$, dada por $T(x, y) = (x, -y)$.

Observe a seguir a determinação da matriz associada:

A base canônica do \mathbb{R}^2 é $C = \{(1,0), (0,1)\}$ e $\dim \mathbb{R}^2 = 2$, logo a matriz será 2×2 .

Calcula-se T a partir da base canônica:

$$T(1,0) = (1,0)$$

$$T(0,1) = (0,-1)$$

Escrevendo T como combinação linear da base canônica, temos:

$$a_{11}(1,0) + a_{21}(0,1) = (a_{11}, 0) + (0, a_{21}) = (1,0); \text{ temos } a_{11} = 1 \text{ e } a_{21} = 0$$

$$a_{12}(1,0) + a_{22}(0,1) = (a_{12}, 0) + (0, a_{22}) = (0,-1); \text{ temos } a_{12} = 0 \text{ e } a_{22} = -1$$

Logo, a matriz associada é:

$$\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

Reflexão em torno do eixo y

Observando o gráfico a seguir, notamos que o ponto A , após sofrer uma reflexão em torno do eixo y , teve como consequência a mudança do sinal da componente x . Supondo que você queira converter esse ponto em um pixel aceso no dispositivo gráfico de saída, é necessário transformá-lo em elemento de uma matriz. Dessa forma, determine a transformação linear e a matriz associada à reflexão em torno do eixo y .

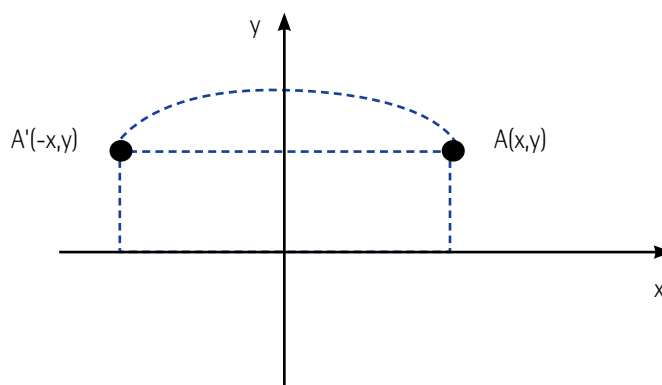


Figura 60

A transformação linear da reflexão é: $T: \mathbb{R}^2 \rightarrow \mathbb{R}^2$, dada por $T(x,y) = (-x,-y)$.

Observe a seguir a determinação da matriz associada:

A base canônica do \mathbb{R}^2 é $C = \{(1,0), (0,1)\}$ e $\dim \mathbb{R}^2 = 2$, logo a matriz será 2×2 .

Calcula-se T a partir da base canônica:

$$T(1,0) = (-1,0)$$

$$T(0,1) = (0,1)$$

Escrevendo T como combinação linear da base canônica, temos:

$$a_{11}(1,0) + a_{21}(0,1) = (a_{11}, 0) + (0, a_{21}) = (-1,0); \text{ temos } a_{11} = -1 \text{ e } a_{21} = 0$$

$$a_{12}(1,0) + a_{22}(0,1) = (a_{12}, 0) + (0, a_{22}) = (0,1); \text{ temos } a_{12} = 0 \text{ e } a_{22} = 1$$

Logo, a matriz associada é:

$$\begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix}$$

Reflexão em torno da origem

Observando o gráfico a seguir, notamos que ponto A , após sofrer uma reflexão em torno da origem, teve como consequência a mudança do sinal da componente x e da componente y . Supondo que você queira converter esse ponto em um pixel aceso no dispositivo gráfico de saída, é necessário transformá-lo em elemento de uma matriz. Dessa forma, determine a transformação linear e a matriz associada à reflexão em torno da origem.

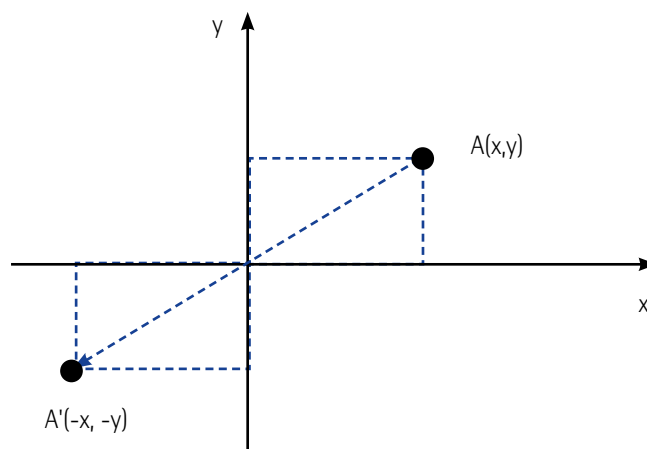


Figura 61

A transformação linear da reflexão é: $T: \mathbb{R}^2 \rightarrow \mathbb{R}^2$, dada por $T(x,y) = (-x,-y)$.

Observe a seguir a determinação da matriz associada:

A base canônica do \mathbb{R}^2 é $C = \{(1,0), (0,1)\}$ e $\dim \mathbb{R}^2 = 2$, logo a matriz será 2×2 .

Calcule T a partir da base canônica:

$$T(1,0) = (-1,0)$$

$$T(0,1) = (0,-1)$$

Escrevendo T como combinação linear da base canônica, temos:

$$a_{11}(1,0) + a_{21}(0,1) = (a_{11}, 0) + (0, a_{21}) = (-1,0); \text{ temos } a_{11} = -1 \text{ e } a_{21} = 0$$

$$a_{12}(1,0) + a_{22}(0,1) = (a_{12}, 0) + (0, a_{22}) = (0,-1); \text{ temos } a_{12} = 0 \text{ e } a_{22} = -1$$

Logo, a matriz associada é:

$$\begin{pmatrix} -1 & 0 \\ 0 & -1 \end{pmatrix}$$

Reflexão em torno da reta $y = x$

Observando o gráfico a seguir, notamos que o ponto A , após sofrer uma reflexão em torno da reta $y = x$, teve como consequência a mudança da posição da componente x com a componente y . Supondo que você queira converter esse ponto em um pixel aceso no dispositivo gráfico de saída, é necessário transformá-lo em elemento de uma matriz. Dessa forma, determine a transformação linear e a matriz associada à reflexão em torno da reta $y = x$.

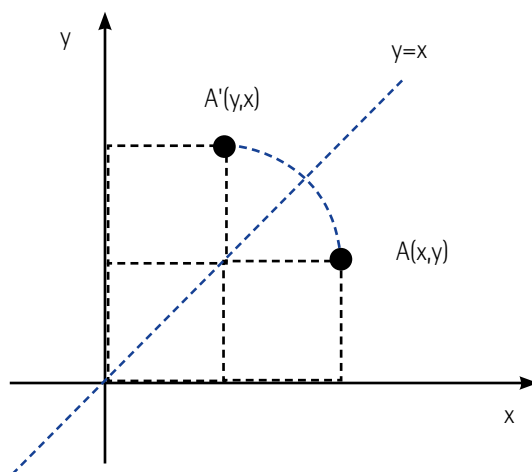


Figura 62

A transformação linear da reflexão é: $T: \mathbb{R}^2 \rightarrow \mathbb{R}^2$, dada por $T(x,y) = (y,x)$.

Observe a seguir a determinação da matriz associada:

A base canônica do \mathbb{R}^2 é $C = \{(1,0), (0,1)\}$ e $\dim \mathbb{R}^2 = 2$, logo a matriz será 2×2 .

Calcule T a partir da base canônica:

$$T(1,0) = (0,1)$$

$$T(0,1) = (1,0)$$

Escrevendo T como combinação linear da base canônica, temos:

$$a_{11}(1,0) + a_{21}(0,1) = (a_{11}, 0) + (0, a_{21}) = (0,1); \text{ temos } a_{11} = 0 \text{ e } a_{21} = 1$$

$$a_{12}(1,0) + a_{22}(0,1) = (a_{12}, 0) + (0, a_{22}) = (1,0); \text{ temos } a_{12} = 1 \text{ e } a_{22} = 0$$

Logo, a matriz associada é:

$$\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

Reflexão em torno da reta $y = -x$

Observando o gráfico a seguir, notamos que ponto A , após sofrer uma reflexão em torno da reta $y = -x$, teve como consequência a mudança da posição da componente x com a componente y além do sinal de ambas. Supondo que você queira converter esse ponto em um pixel aceso no dispositivo gráfico de saída, é necessário transformá-lo em elemento de uma matriz. Dessa forma, determine a transformação linear e a matriz associada à reflexão em torno da reta $y = -x$.

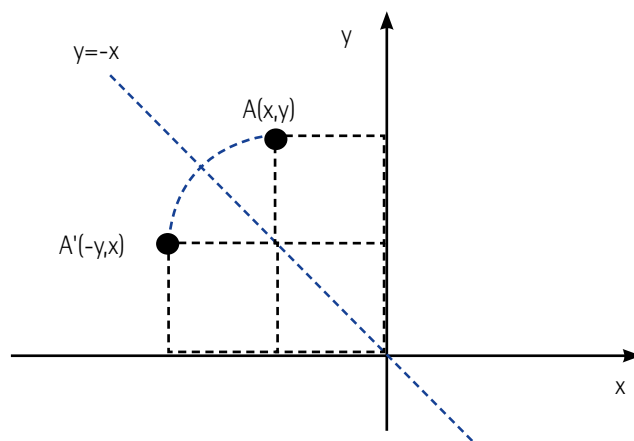


Figura 63

A transformação linear da reflexão é: $T: \mathbb{R}^2 \rightarrow \mathbb{R}^2$, dada por $T(x,y) = (-y,-x)$.

Observe a seguir a determinação da matriz associada:

A base canônica do \mathbb{R}^2 é $C = \{(1,0), (0,1)\}$ e $\dim \mathbb{R}^2 = 2$, logo a matriz será 2×2 .

Calcule T a partir da base canônica:

$$T(1,0) = (0,-1)$$

$$T(0,1) = (-1,0)$$

Escrevendo T como combinação linear da base canônica, temos:

$$a_{11}(1,0) + a_{21}(0,1) = (a_{11}, 0) + (0, a_{21}) = (0,-1); \text{ temos } a_{11} = 0 \text{ e } a_{21} = -1$$

$$a_{12}(1,0) + a_{22}(0,1) = (a_{12}, 0) + (0, a_{22}) = (-1,0); \text{ temos } a_{12} = -1 \text{ e } a_{22} = 0$$

Logo, a matriz associada é:

$$\begin{pmatrix} 0 & -1 \\ -1 & 0 \end{pmatrix}$$

Dilatação

Observando o gráfico a seguir, notamos que ponto A , após sofrer uma dilatação, teve como consequência o incremento de K vezes o seu módulo. Supondo que você queira converter esse ponto em um pixel aceso no dispositivo gráfico de saída, é necessário transformá-lo em elemento de uma matriz. Dessa forma, determine a transformação linear e a matriz associada à dilatação desse vetor.

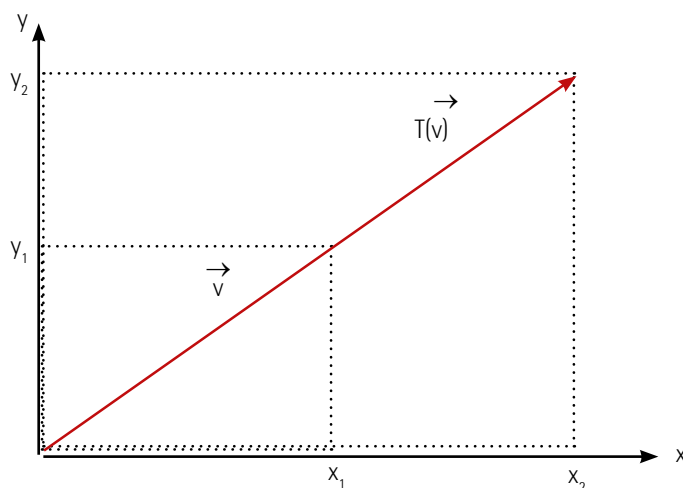


Figura 64



Observação

Caso o valor de k seja $0 < k < 1$, significa que o vetor sofreu contração.

A transformação linear da dilatação é: $T: \mathbb{R}^2 \rightarrow \mathbb{R}^2$, dada por $T(x,y) = (kx,ky)$.

Observe a seguir a determinação da matriz associada:

A base canônica do \mathbb{R}^2 é $C = \{(1,0), (0,1)\}$ e $\dim \mathbb{R}^2 = 2$, logo a matriz será 2×2 .

Calcule T a partir da base canônica:

$$T(1,0) = (1k,0)$$

$$T(0,1) = (0, 1k)$$

Escrevendo T como combinação linear da base canônica, temos:

$$a_{11}(1,0) + a_{21}(0,1) = (a_{11} + 0) + (0 + a_{21}) = (k, 0); \text{ temos } a_{11} = k \text{ e } a_{21} = 0$$

$$a_{12}(1,0) + a_{22}(0,1) = (a_{12} + 0) + (0 + a_{22}) = (0, k); \text{ temos } a_{12} = 0 \text{ e } a_{22} = k$$

Logo, a matriz associada é:

$$\begin{pmatrix} k & 0 \\ 0 & k \end{pmatrix}$$

Dilatação na direção x

Supondo um objeto 2D que sofre dilatação ou contração apenas na direção x , notamos como consequência o incremento de K vezes o valor de sua abscissa, acarretando um alargamento do objeto. Supondo que você queira converter esse objeto em pixels acesos no dispositivo gráfico de saída, é necessário transformá-lo em elemento de uma matriz. Dessa forma, determine a transformação linear e a matriz associada à dilatação desse objeto.

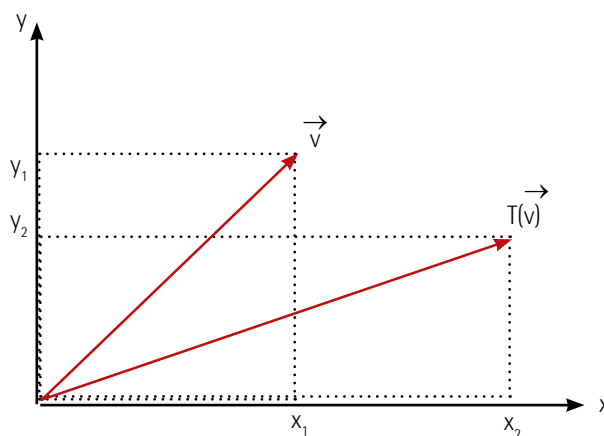


Figura 65

A transformação linear da dilatação é: $T: \mathbb{R}^2 \rightarrow \mathbb{R}^2$, dada por $T(x,y) = (kx,y)$.

Observe a seguir a determinação da matriz associada:

A base canônica do \mathbb{R}^2 é $C = \{(1,0), (0,1)\}$ e $\dim \mathbb{R}^2 = 2$, logo a matriz será 2×2 .

Calcule T a partir da base canônica:

$$T(1,0) = (k,0)$$

$$T(0,1) = (0, 1)$$

Escrevendo T como combinação linear da base canônica, temos:

$$a_{11}(1,0) + a_{21}(0,1) = (a_{11} + 0) + (0 + a_{21}) = (k, 0); \text{ temos } a_{11} = k \text{ e } a_{21} = 0$$

$$a_{12}(1,0) + a_{22}(0,1) = (a_{12} + 0) + (0 + a_{22}) = (0, 1); \text{ temos } a_{12} = 0 \text{ e } a_{22} = 1$$

Logo, a matriz associada é:

$$\begin{pmatrix} k & 0 \\ 0 & 1 \end{pmatrix}$$

Cisalhamento na direção x

Este transforma o retângulo $OBCD$ no paralelogramo $OBC'D'$. Cada ponto (x,y) é deslocado horizontalmente até $(x+ky,y)$. Observe que os pontos sobre o eixo das abscissas não são deslocados, pois possuem a ordenada nula. Por essa razão, os quadriláteros $OBCD$ e $OBC'D'$ possuem a mesma base, OB .

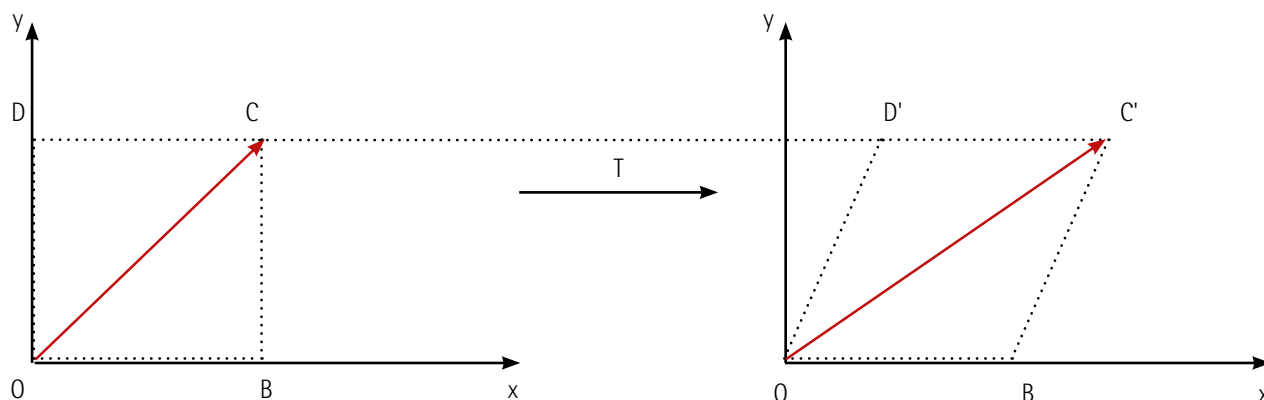


Figura 66

A transformação linear do cisalhamento na direção do eixo x é: $T: \mathbb{R}^2 \rightarrow \mathbb{R}^2$, dada por $T(x,y) = (x+ky,y)$.

Observe a seguir a determinação da matriz associada:

A base canônica do \mathbb{R}^2 é $C = \{(1,0), (0,1)\}$ e $\dim \mathbb{R}^2 = 2$, logo a matriz será 2×2 .

Calcule T a partir da base canônica:

$$T(1,0) = (1+k \cdot 0, 0) = (1,0)$$

$$T(0,1) = (0+k \cdot 1, 1) = (k,1)$$

Escrevendo T como combinação linear da base canônica, temos:

$$(1,0) = a_{11}(1,0) + a_{21}(0,1) = (a_{11}, 0) + (0, a_{21}) = (a_{11}, a_{21}); \text{ temos } a_{11} = 1 \text{ e } a_{21} = 0$$

$$(k,1) = a_{12}(1,0) + a_{22}(0,1) = (a_{12}, 0) + (0, a_{22}) = (a_{12}, a_{22}); \text{ temos } a_{12} = k \text{ e } a_{22} = 1$$

Logo, a matriz associada é:

$$\begin{pmatrix} 1 & k \\ 0 & 1 \end{pmatrix}$$

Exemplo

Uma aplicação simples na computação gráfica são as letras vistas na tela do computador. Considere a letra maiúscula I , desenhada em um sistema de coordenadas em \mathbb{R}^2 , como na figura a seguir. Sabendo que a constante $k = 1$, desenhe a letra I resultante do cisalhamento no eixo x .

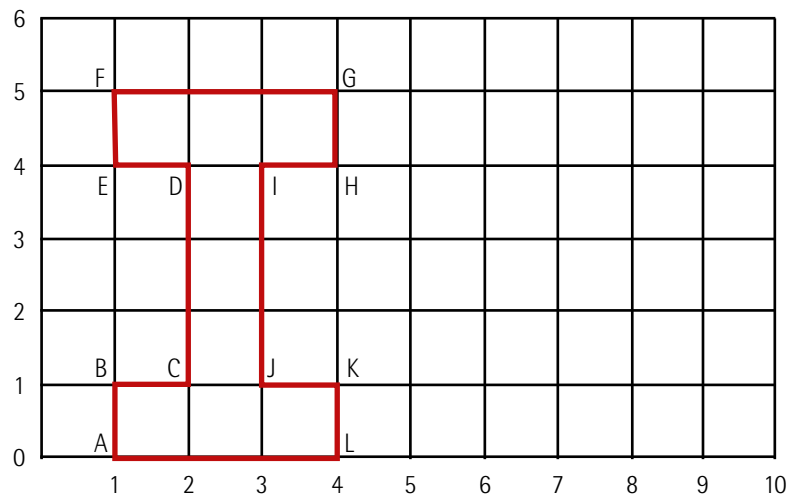


Figura 67

$$\begin{pmatrix} 1 & k \\ 0 & 00^1 \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 0 & 00^1 \end{pmatrix}$$

$$\begin{pmatrix} 1 & 1 \\ 0 & 00^1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 1 & 2 & 2 & 1 & 1 & 4 & 4 & 3 & 3 & 4 & 4 \\ 0 & 1 & 1 & 4 & 4 & 5 & 5 & 4 & 4 & 1 & 1 & 0 \end{pmatrix}$$

$$= \begin{pmatrix} 1*1+1*0 & 1*1+1*1 & 1*2+1*1 & 1*2+1*4 & 1*1+1*4 & \dots & 1*3+1*1 & 1*4+1*1 & 1*4+1*0 \\ 0*1+1*0 & 0*1+1*1 & 0*2+1*1 & 0*2+1*4 & 0*1+1*4 & \dots & 0*3+1*1 & 0*4+1*1 & 0*4+1*0 \end{pmatrix}$$

$$= \begin{pmatrix} 1 & 2 & 3 & 6 & 5 & 6 & 9 & 8 & 7 & 4 & 5 & 4 \\ 0 & 1 & 1 & 4 & 4 & 5 & 5 & 4 & 4 & 1 & 1 & 0 \end{pmatrix}$$

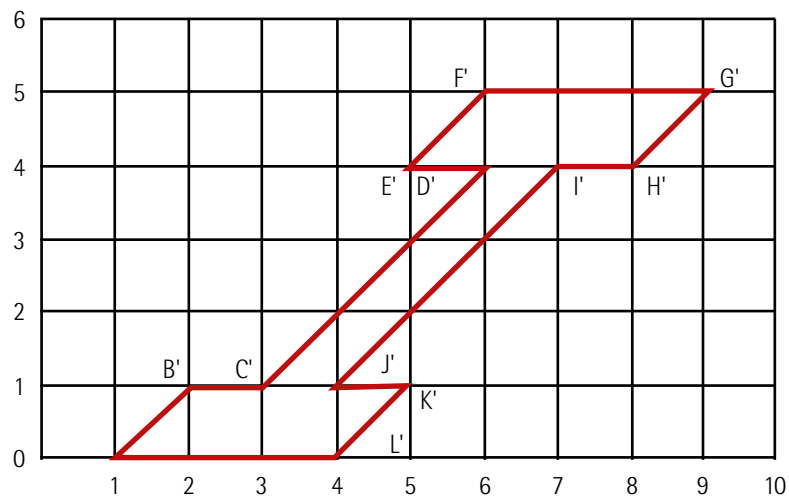


Figura 68

A figura a seguir ilustra o cisalhamento na direção do eixo y . A transformação $T : \mathbb{R}^2 \rightarrow \mathbb{R}^2$, dada por $T(x,y) = (x, y+kx)$, é denominada cisalhamento na direção do eixo y .

O cisalhamento na direção do eixo y transformou o retângulo $OBCD$ no paralelogramo $OB'C'D$. Cada ponto (x, y) foi deslocado verticalmente até $(x, y + kx)$. Observe que os pontos sobre o eixo das ordenadas não foram deslocados, pois possuem a abscissa nula.

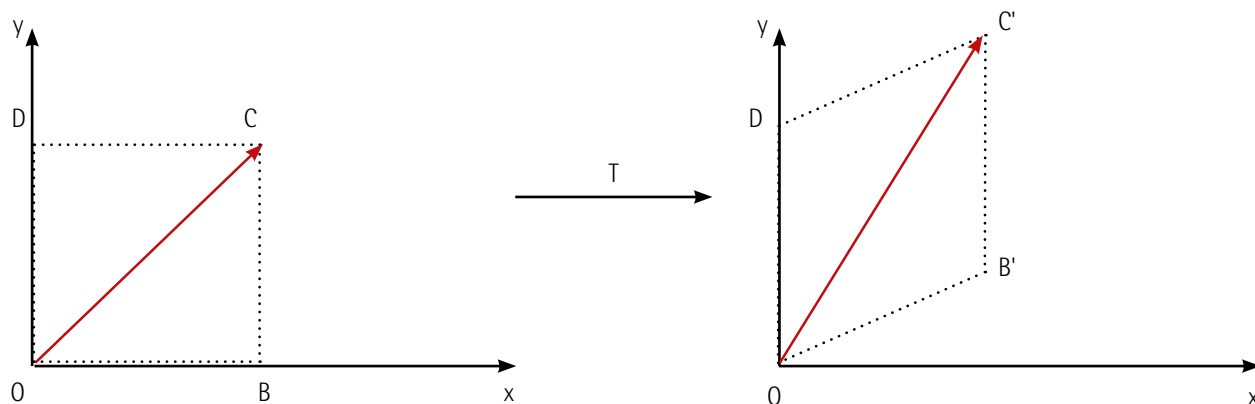


Figura 69

A transformação linear do cisalhamento na direção do eixo y é: $T: \mathbb{R}^2 \rightarrow \mathbb{R}^2$, dada por $T(x,y) = (x, y+kx)$

Observe a seguir a determinação da matriz associada:

A base canônica do \mathbb{R}^2 é $C = \{(1,0), (0,1)\}$ e $\dim \mathbb{R}^2 = 2$, logo a matriz será 2×2 .

Calcule T a partir da base canônica:

$$T(1,0) = (1, 0+k \cdot 1) = (1, k)$$

$$T(0,1) = (0, 1+k \cdot 0) = (0, 1)$$

Escrevendo T como combinação linear da base canônica, temos:

$$(1, k) = a_{11} (1,0) + a_{21} (0,1) = (a_{11}, 0) + (0, a_{21}) = (a_{11}, a_{21}); \text{ temos } a_{11} = 1 \text{ e } a_{21} = k$$

$$(0,1) = a_{12} (1,0) + a_{22} (0,1) = (a_{12}, 0) + (0, a_{22}) = (a_{12}, a_{22}); \text{ temos } a_{12} = 0 \text{ e } a_{22} = 1$$

Logo, a matriz associada é:

$$\begin{pmatrix} 1 & 0 \\ k & 1 \end{pmatrix}$$

Exemplo

Considere o retângulo ABCD nomeado no sentido horário, de base 2(eixo x) e altura 1(eixo y), tendo o vértice A apoiado na origem do plano cartesiano. Aplicando a transformação de cisalhamento no eixo y, de constante $k=2$, qual é a matriz de vértices após a transformação?

1) Destacam-se as coordenadas do retângulo ABCD:

A(0,0), B(0,1), C(2,1), D(2,0)

2) Depois, obtém-se a matriz dos vértices do retângulo:

$$= \begin{pmatrix} 0 & 0 & 2 & 2 \\ 0 & 1 & 1 & 0 \end{pmatrix}$$

3) Multiplica-se a matriz de transformação com a matriz de vértices:

$$\begin{pmatrix} 1 & 0 \\ 2 & 00^1 \end{pmatrix} \cdot \begin{pmatrix} 0 & 0 & 2 & 2 \\ 0 & 1 & 1 & 0 \end{pmatrix} = \begin{pmatrix} 1 \cdot 0 + 0 \cdot 0 & 1 \cdot 0 + 0 \cdot 1 & 1 \cdot 2 + 0 \cdot 1 & 1 \cdot 2 + 0 \cdot 0 \\ 2 \cdot 0 + 0 \cdot 0 & 2 \cdot 0 + 0 \cdot 1 & 2 \cdot 2 + 0 \cdot 1 & 2 \cdot 2 + 0 \cdot 0 \end{pmatrix}$$

$$= \begin{pmatrix} 0 & 0 & 2 & 2 \\ 0 & 1 & 5 & 4 \end{pmatrix}$$

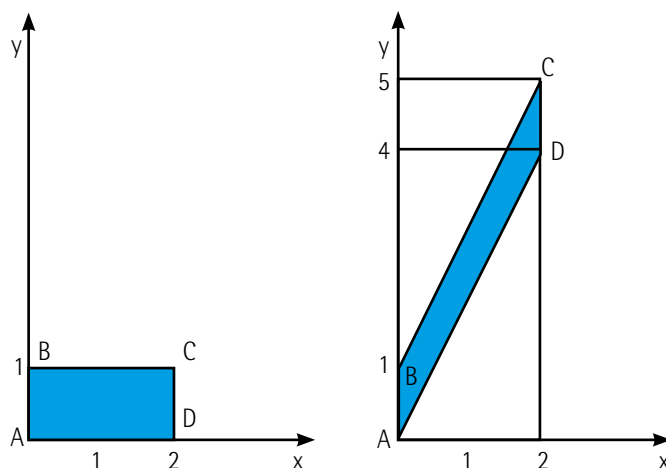


Figura 70

7 CONCATENAÇÃO DE TRANSFORMAÇÕES GEOMÉTRICAS

É uma técnica utilizada para aplicar diversas transformações a um objeto em uma única operação. Isso é feito multiplicando as matrizes de transformação de cada operação em ordem sequencial.

Por exemplo, se quisermos realizar uma translação seguida de uma rotação em um objeto, podemos concatenar as matrizes de transformação das duas operações em uma única matriz de transformação e aplicá-la ao objeto. A matriz resultante será o resultado da multiplicação das matrizes de transformação das duas operações.

A ordem das transformações na concatenação é importante, pois a multiplicação das matrizes não é comutativa. Ou seja, a ordem em que as transformações são aplicadas afeta o resultado final.

A concatenação de transformações geométricas é amplamente usada em aplicações de computação gráfica, como modelagem 3D, animação e jogos.

A concatenação de transformações geométricas em computação gráfica pode ser definida matematicamente como a multiplicação de matrizes de transformação em ordem sequencial.

Suponha que temos uma sequência de transformações geométricas $T_1, T_2, T_3, \dots, T_n$ que queremos aplicar a um objeto. Cada uma dessas transformações pode ser representada por uma matriz de transformação homogênea, denotada por $M_1, M_2, M_3, \dots, M_n$.

A matriz resultante da concatenação de todas as transformações é dada pela multiplicação das matrizes de transformação em ordem sequencial:

$$M = M_n \times M_{n-1} \times \dots \times M_2 \times M_1$$

A ordem das transformações na concatenação é importante, pois a multiplicação das matrizes não é comutativa.

Um exemplo de concatenação de transformações geométricas pode ser a transformação de um objeto em uma cena 3D. Suponha que temos um cubo na origem e queremos movê-lo para uma posição específica na cena, rotacioná-lo e redimensioná-lo. Podemos fazer isso concatenando as matrizes de transformação de translação, rotação e escala em ordem sequencial e aplicando a matriz resultante ao cubo. O resultado será o cubo transformado de acordo com as transformações especificadas.

7.1 Coordenadas homogêneas

Coordenadas homogêneas são uma maneira de representar pontos, vetores e matrizes com escopo de operar esses objetos, bem como favorecer generalizações no espaço 2D ou 3D. No caso do espaço 2D, cada ponto (x,y) é representado por (x,y,w) , sendo $w = 1$. Logo, a representação de um ponto em duas dimensões é dada por $(x,y,1)$.

Em computação gráfica, a representação em coordenadas homogêneas para indicar os pontos no espaço 2D converte as equações das transformações geométricas em multiplicação de matrizes de ordem 3.

Por exemplo, a transformação de dilatação fica:

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} k_x & 0 & 0 \\ 0 & k_y & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

Na equação matricial anterior, é possível perceber que as coordenadas são os vetores representados em colunas de três elementos, e a operação de transformação de dilatação está representada pela matriz de ordem 3.

7.2 Concatenação de transformações

É a aplicação sucessiva de várias transformações geométricas em um objeto para produzir um resultado final desejado. Essas transformações podem incluir translação, rotação, escala ou dilatação e projeção. A concatenação de transformações é feita multiplicando as matrizes de transformação correspondentes e aplicando a matriz resultante no objeto.

Por exemplo, considere um objeto que precisa ser primeiro rotacionado em torno do eixo z, depois escalado e, finalmente, transladado para uma posição final. Sejam R a matriz de rotação, S a matriz de escala e T a matriz de translação correspondente a cada transformação, a matriz de transformação resultante pode ser obtida por:

$$M = T \cdot S \cdot R$$

Em resumo, a concatenação de transformações é um processo importante em computação gráfica para aplicar uma sequência de transformações geométricas em um objeto, a fim de produzir uma transformação composta que possa ser aplicada no objeto de uma só vez.

Vejamos como ficam, na forma matricial, as principais transformações aplicadas de forma sucessivas.

Concatenação de translações (T)

Em duas translações sucessivas aplicadas a uma posição P, a posição final P' será:

$$P' = T(t_{x1}, t_{y1}) \cdot \{T(t_{x2}, t_{y2}) \cdot P\} = \{T(t_{x1}, t_{y1}) \cdot T(t_{x2}, t_{y2})\} \cdot P$$

$$\begin{pmatrix} 1 & 0 & t_{x2} \\ 0 & 1 & t_{y2} \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & t_{x1} \\ 0 & 1 & t_{y1} \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & t_{x2} + t_{x1} \\ 0 & 1 & t_{y2} + t_{y1} \\ 0 & 0 & 1 \end{pmatrix}$$

Concatenação de rotações (R)

Em duas rotações sucessivas aplicadas a uma posição P, a posição final P' será:

$$P' = R(\theta_1) \{R(\theta_2) P\} = \{R(\theta_1) R(\theta_2)\} \cdot P$$

$$\begin{pmatrix} \cos\theta_1 & -\sin\theta_1 & 0 \\ \sin\theta_1 & \cos\theta_1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} \cos\theta_2 & -\sin\theta_2 & 0 \\ \sin\theta_2 & \cos\theta_2 & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} \cos\theta_1 \cdot \cos\theta_2 - \sin\theta_1 \cdot \sin\theta_2 & -\cos\theta_1 \cdot \sin\theta_2 - \sin\theta_1 \cdot \cos\theta_2 & 0 \\ \sin\theta_1 \cdot \cos\theta_2 + \cos\theta_1 \cdot \sin\theta_2 & -\sin\theta_1 \cdot \sin\theta_2 + \cos\theta_1 \cdot \cos\theta_2 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Concatenação de escalas (K)

Em duas transformações de escalas sucessivas aplicadas a uma posição P, a posição final P' será:

$$P' = K(k_{x2}, k_{y2}) \cdot \{K(k_{x1}, k_{y1}) \cdot P\} = \{K(k_{x2}, k_{y2}) \cdot K(k_{x1}, k_{y1})\} \cdot P$$

$$\begin{pmatrix} kx_2 & 0 & 0 \\ 0 & ky_2 & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} kx_1 & 0 & 0 \\ 0 & ky_1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} kx_1 \cdot kx_2 & 0 & 0 \\ 0 & ky_1 \cdot ky_2 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

8 REPRESENTAÇÃO E MODELAGEM DE OBJETOS 2D E 3D

Todas as transformações efetuadas no item anterior são aplicadas às coordenadas dos vértices dos objetos transformando-os e posicionando-os.

Dessa forma, devemos saber representar objetos, imagens e figuras em termos das coordenadas de seus vértices.

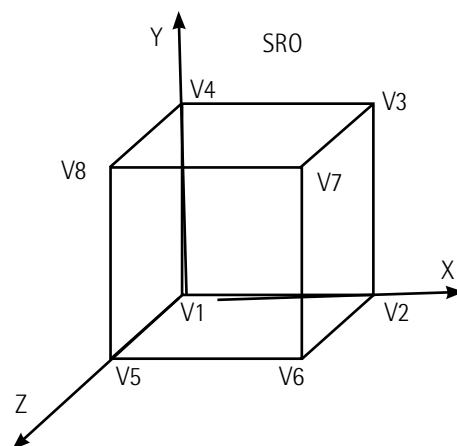


Figura 71

Usando-se um sistema cartesiano xyz, podemos representar adequadamente esses valores e guardá-los em uma estrutura de dados conveniente (lista de vértices, arestas e faces).

Consideraremos um sistema de eixos onde o eixo x é horizontal e positivo para a direita, o eixo y é vertical e positivo para cima e o eixo z é perpendicular a ambos, positivo na direção do produto vetorial $e_3 = e_1 \cdot e_2$, onde e_1 , e_2 e e_3 são os vetores unitários nas direções x, y e z, respectivamente (base). Essa orientação é denominada sistema de mão direita.

Como exemplo, consideremos um cubo de lado 1 (figura anterior) com um dos vértices apoiado na origem do plano cartesiano e será representado pela lista de vértices e de arestas no SRO (ou no SRN) da seguinte forma:

Tabela 3

Vértice	V1	V2	V3	V4	V5	V6	V7	V8
Coordenadas	(0,0,0)	(1,0,0)	(1,1,0)	(0,1,0)	(0,0,1)	(1,0,1)	(1,1,1)	(0,1,1)

Já as arestas serão dadas pelo par (vértice inicial, vértice final) assim:

Tabela 4

Arestas	V1-V2	V2-V3	V3-V4	V4-V1	V1-V5	V2-V6	V3-V7	V4-V8	V5-V6	V6-V7	V7-V8	V8-V5
---------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------

Nesse caso, podemos utilizar a notação de vetores (arrays). Um conjunto de vetores receberá a lista dos vértices armazenados em uma matriz $3 \times N$ (ou $2 \times N$ no caso de 2D), onde N é o número de vértices ordenados de 1 até N colunas e outro receberá a lista de arestas. Assim:

$$\text{Vértices}_{\text{cubo}} = \begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix}$$

Já os vetores de arestas serão armazenados em uma matriz $2 \times M$, independentemente se o objeto é 2D ou 3D. As arestas são ordenadas em colunas de 1 até M. A primeira linha contém o índice do vértice inicial e a segunda o índice do vértice final.

$$\text{Arestas}_{\text{cubo}} = \begin{pmatrix} 1 & 2 & 3 & 4 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 2 & 3 & 4 & 1 & 5 & 6 & 7 & 8 & 6 & 7 & 8 & 5 \end{pmatrix}$$

Essa forma de estruturar os vértices e as arestas de objetos não é única.

Quando desenhamos na tela do computador um objeto modelado por seus vértices $V = \{V1, V2, V3, \dots, Vn\}$, devemos primeiro transformar as coordenadas de cada vértice conforme a

necessidade da representação (isso implica transformações geométricas descritas anteriormente e transformações dos vértices resultantes do SRO à SRU e à SRD, conforme já acentuado) e, em seguida, desenhar as arestas ligando as coordenadas x e y , em pixel, de cada um de seus vértices extremos.

Por exemplo, considere a aresta $A1$ que liga os vértices $V1$ e $V2$. Após efetuarmos uma série de cálculos para posicionar o objeto adequadamente no SRU e passá-lo para o SRD, tomamos apenas os valores transformados das coordenadas X_{pixel} e Y_{pixel} dos vértices $V1$ e $V2$ e os ligamos com um algoritmo de linhas (Bresenham).

Neste tópico, contudo, vamos nos preocupar somente em como podemos criar uma estrutura de vértices e arestas para serem usadas oportunamente.

Exemplo

Considere o objeto a seguir, um pentágono, representado no SRO, ou no SRU, e escreva as matrizes dos vértices e arestas para ele (figura fora de escala e proporções).

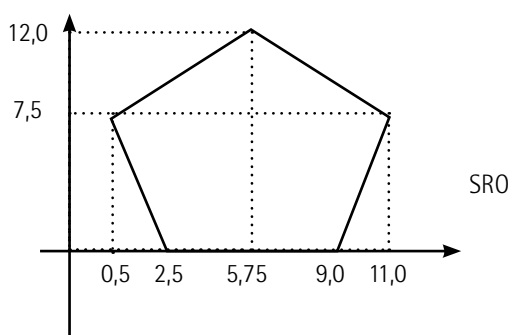


Figura 72

Resolução

Inicialmente, devemos identificar e rotular cada vértice e aresta do objeto, conforme figura a seguir.

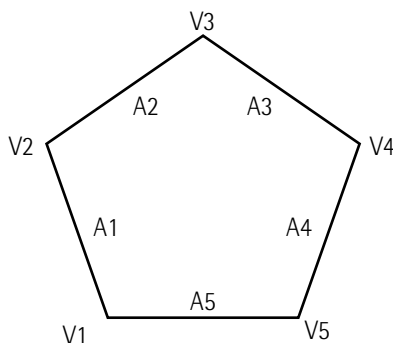


Figura 73

Comparando a posição de cada elemento com os valores representados do objeto no SRO, teremos a matriz dos vértices de tipo 2 x 5 (pois o objeto é 2D e tem 5 vértices):

$$\text{Vértices} = \begin{pmatrix} 2,5 & 0,5 & 5,75 & 11 & 9 \\ 0 & 7,5 & 12 & 7,5 & 0 \end{pmatrix}$$

Onde cada vértice ocupa, em ordem, uma coluna; a primeira linha guarda a coordenada x, e a segunda linha guarda a coordenada y.

No caso do objeto em questão, as arestas ligam os vértices um a um em uma sequência que pode ser armazenada na matriz de tipo 2 x 5 (pois são sempre duas linhas e, no caso, cinco arestas), assim:

$$\text{Arestas} = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 2 & 3 & 4 & 5 & 1 \end{pmatrix}$$

Exemplo

Escreva um programa, ou pseudocódigo, que utilize as matrizes vértices e arestas em um desenho de 2D. Considere que o desenho está centralizado e completamente contido no interior da viewport, para evitar o clipping, cujos limites são (XPMIN, YPMIN) e (YPMAX, YPMAX).

Resolução

Vamos descrever um pseudocódigo para resolver esse problema. Notemos que a matriz dos vértices é um array de tipo real com valores obtidos no SRO. Cada vértice terá que ser mapeado para a viewport e armazenado em um array de tipo inteiro. Por sua vez, a matriz de arestas já é um array de tipo inteiro. Por fim, assume-se que existam funções que determinam os valores máximos e mínimos de x e y dos vértices e uma rotina que calcula e traça linhas denominadas Bresenham (xi, yi, xf, yf). Então:

nVertices, nArestas: inteiro;

SX, SY: real;

XPMIN, YPMIN: inteiro;

XRMIN, YRMIN: real;

Leia (nVertices, nArestas);

Matriz VSRO[2,nVertices]: real;

Matriz VSRD[2,nVertices], A[2,nArestas]: inteiro;

```
Leia (VSRO[1..2,1..nVertices]);

Leia (A[1..2,1..nArestas]);

// Determinando o tamanho da WINDOW para envolver
// completamente o objeto. Ou seja:

XRMIN = MINIMO_DE_VSRO(1,1..nVertices); //apenas 1ª linha
XRMAX = MAXIMO_DE_VSRO(1,1..nVertices);
YRMIN = MINIMO_DE_VSRO(2,1..nVertices); // apenas 2ª linha
YRMAX = MAXIMO_DE_VSRO(2,1..nVertices);

// Define viewport

Leia (XPMIN,YPMIN,YPMAX,YPMAX);

// Calcula fator de escala para o mapeamento window-to-viewport

SX = (XPMAX-XPMIN)/(XRMAX-XRMIN);
SY = (YPMAX-YPMIN)/(YRMAX-YRMIN);

// Converte VSRO para VSRD

Para i = 1 até nVertices faça

VSRD(1,i) = SX*(VSRO(1,i)-XRMIN)+XPMIN;
VSRD(2,i) = -SY*(VSRO(2,i)-YRMIN)+YPMAX;

Fim_Para

// Desenha objeto na viewport usando Bresenham(xi,yi,xf,yf)

Para i = 1 até nArestas faça

Bresenham(VSRD(1,A(1,i)), VSRD(2,A(1,i)), VSRD(1,A(2,i)), VSRD(2,A(2,i)));

Fim_Para

Fim_Programa
```

8.1 Transformações projetivas

Projeções em computação gráfica são usadas para transformar objetos tridimensionais em uma imagem bidimensional, que pode ser exibida em uma tela. Em outras palavras, deve ser criada uma imagem em duas dimensões a partir da cena em três dimensões que contém os objetos.

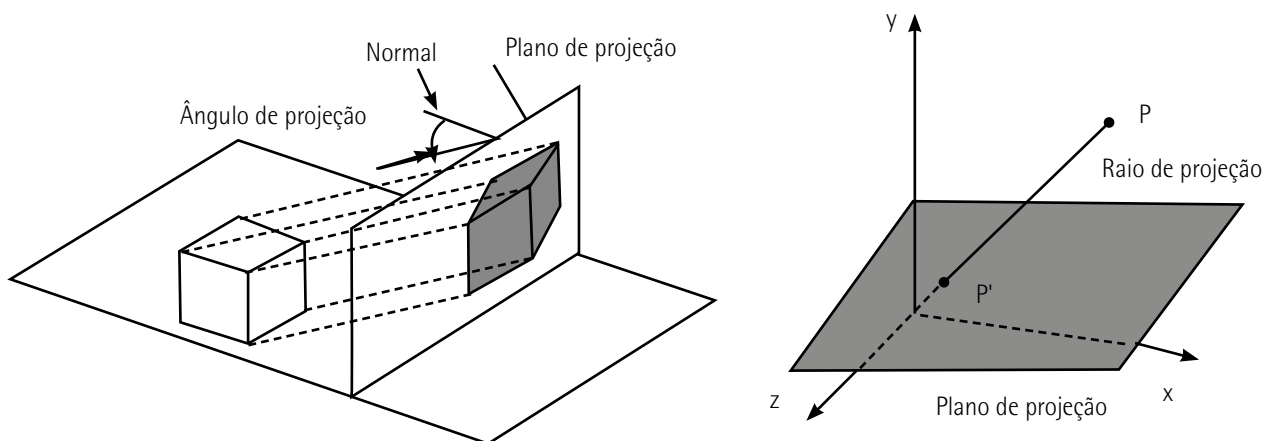


Figura 74

São elementos básicos:

- **Plano de projeção:** superfície onde será projetado o objeto, ou seja, onde ele será representado em 2D.
- **Raios de projeção:** retas que passam pelos pontos do objeto e pelo centro de projeção.
- **Centro de projeção:** ponto fixo de onde os raios de projeção partem.

Existem dois tipos principais de projeções: paralelas ou cilíndricas e projeções em perspectiva ou cônicas.

As paralelas são aquelas em que os raios de luz que atingem o objeto são paralelos entre si.

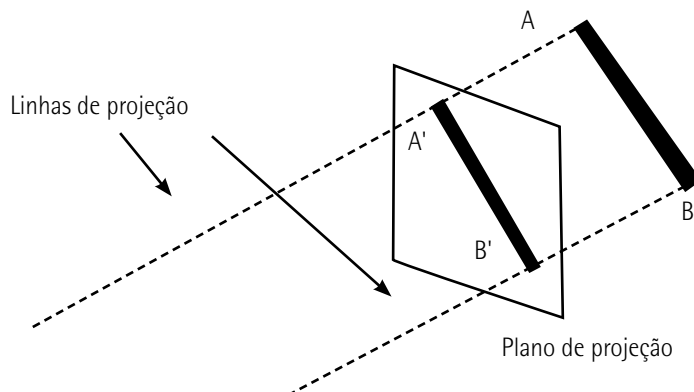


Figura 75 – Projeção cilíndrica



Observação

Com origem no latim, a palavra perspectiva significa "ver através de". Trata-se da representação bidimensional de algo tridimensional (CRUZ, 2019).

Isso significa que as linhas paralelas no objeto projetado permanecem paralelas na imagem projetada. Existem dois tipos de projeções paralelas: ortográficas e oblíquas.

Na projeção ortográfica, os raios de luz atingem o objeto de forma perpendicular ao plano de projeção. Como resultado, a projeção é feita sem distorção de perspectiva, ou seja, todas as partes do objeto são projetadas em tamanho igual. A equação para a projeção ortográfica é simplesmente uma projeção em dois planos, x e y .

Já na projeção oblíqua, os raios de luz atingem o objeto em um ângulo diferente de 90 graus em relação ao plano de projeção. Isso resulta em uma distorção de perspectiva, em que as partes do objeto mais próximas do observador são projetadas em tamanho maior do que as partes mais distantes. A equação para a projeção oblíqua envolve um ângulo de inclinação em relação ao plano de projeção.

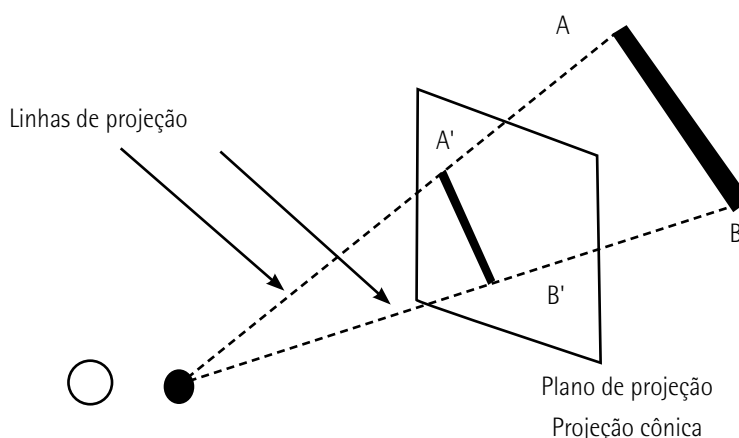


Figura 76

Por outro lado, na projeção em perspectiva, os raios de luz que atingem o objeto se encontram em um ponto específico chamado ponto de fuga. Isso resulta em uma distorção de perspectiva em que as partes do objeto mais próximas do observador são projetadas em tamanho maior do que as partes mais distantes, mas de uma forma mais realista do que na projeção oblíqua. A equação para a projeção em perspectiva envolve a distância do objeto ao ponto de fuga.

Como dito anteriormente, as transformações projetivas são usadas para que objetos tridimensionais possam ser visualizados na tela. Logo, constituem uma etapa importante no pipeline de renderização. Elas envolvem a multiplicação da matriz de transformação do objeto pela matriz de projeção para obter a matriz de transformação final.

A figura a seguir mostra um esquema das subdivisões das transformações projetivas.

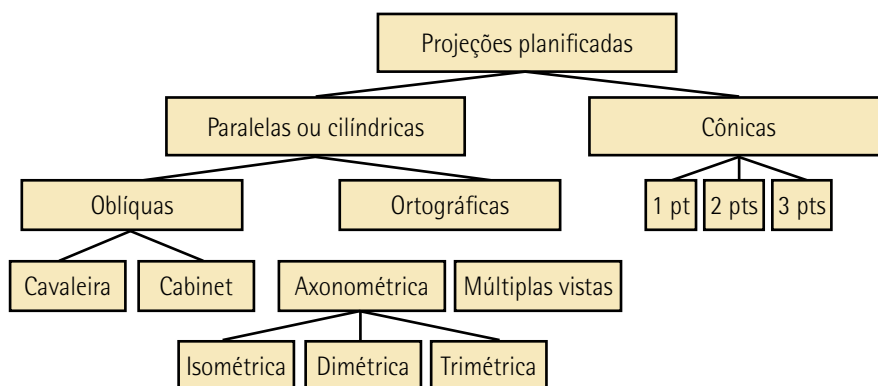


Figura 77

8.2 Projeções paralelas (ortográficas e oblíquas) e em perspectiva

A projeção cilíndrica, também chamada de projeção paralela, é o tipo de projeção cujos raios projetantes que incidem no objeto e no plano de projeção são todos paralelos entre si, como as geratrizes do cilindro. A projeção cilíndrica pode ser ortogonal (ortográfica) ou oblíqua.

Na projeção cilíndrica oblíqua, os projetantes partem do infinito e têm direção oblíqua em relação ao plano de projeção, isto é, formam ângulos diferentes de 90° . Assim,

- o centro de projeção é localizado no infinito;
- todas as linhas de projeção são paralelas entre si;
- são tradicionalmente usadas em engenharia e desenhos técnicos;
- em alguns casos, preservam as dimensões do objeto;
- não produzem imagem realista.

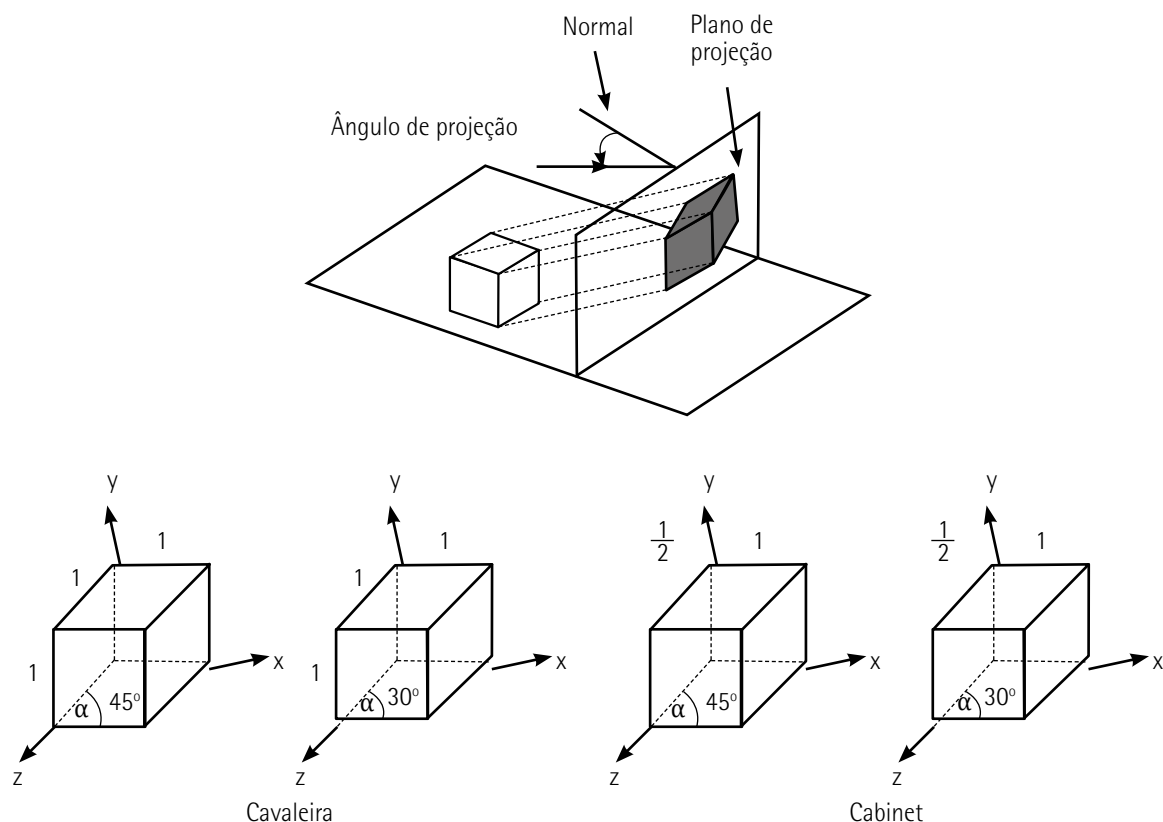
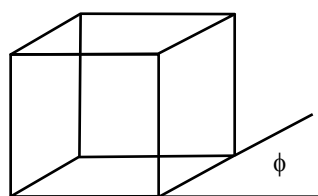


Figura 78 – Projeções cilíndricas (paralelas) oblíquas

Para manter uma noção de profundidade, adiciona-se um ângulo que relaciona a distância do plano de projeção com uma transformação linear, cuja matriz homogênea é dada por:



$$M_p = \begin{pmatrix} 1 & 0 & \lambda \cos \phi & 0 \\ 0 & 1 & \lambda \sin \phi & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Figura 79

Onde ϕ é o ângulo da projeção e λ é um fator que estabelece a relação entre a profundidade em 3D com a elevação vertical da projeção. Assumindo $\lambda=0$, temos a projeção ortográfica, onde as profundidades não são representadas. Com $\lambda = 1.0$, a projeção recebe o nome de projeção cavaleira, sendo muito utilizada em desenho técnico, e para $\lambda = 0.5$ obtém-se a projeção cabinet, onde os comprimentos ao longo da profundidade são representados com a metade do seu valor.

Na projeção cilíndrica ortogonal, os projetantes partem do infinito e têm direção ortogonal em relação ao plano de projeção, isto é, formam com o plano um ângulo de 90° .

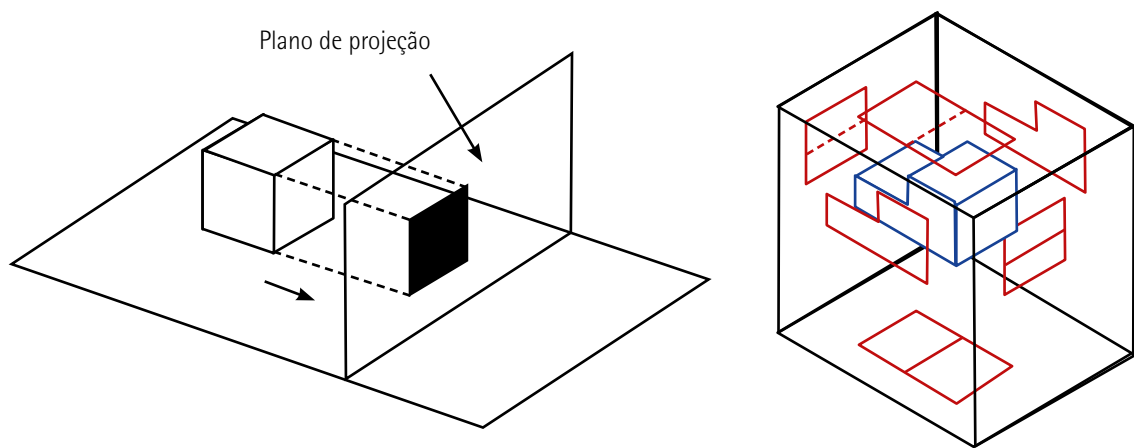
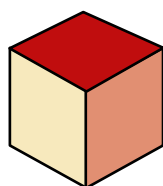
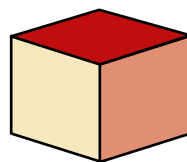


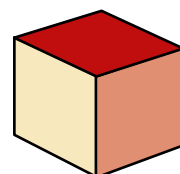
Figura 80 – Projeções cilíndricas (paralelas) ortogonais de múltiplas vistas



Isométrica



Dimétrica



Trimétrica

Figura 81 – Projeções cilíndricas (paralelas) ortogonais axonométricas

8.3 Projeção cônica

A projeção cônica, ou perspectiva, é o tipo de projeção cujos raios que incidem no objeto e no plano de projeção são todos concorrentes no ponto O (vértice do cone), como as geratrizes do cone. O centro de projeção, nesse caso, está a uma distância finita do objeto e os projetantes são convergentes.

- todos os raios de projeção partem do centro de projeção e interceptam o plano de projeção com diferentes ângulos;
- representam a cena vista de um ponto de observação a uma distância finita;
- os raios projetores não podem ser paralelos.
- baseiam-se no número de pontos de fuga da imagem projetada;
- são mais realistas na representação de objetos;
- não reproduzem as verdadeiras medidas do objeto.

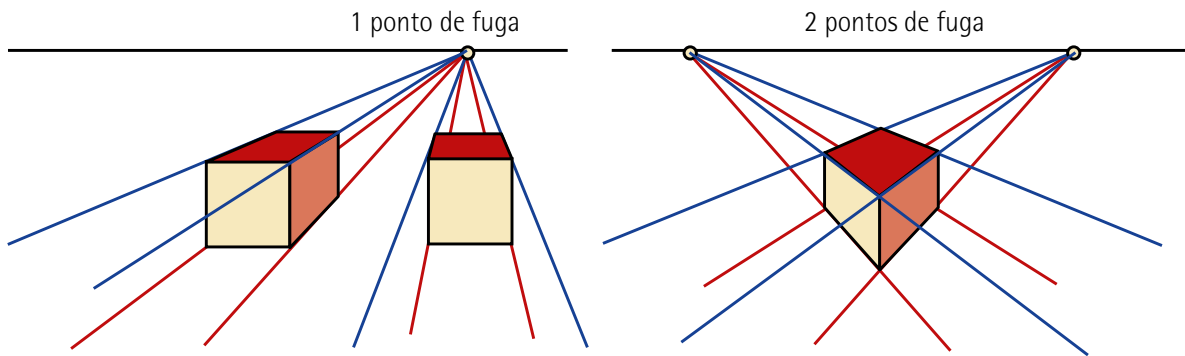


Figura 82 – Projeções cônicas com um e dois pontos de fuga



Saiba mais

Os pontos de fuga dão mais realidade ao desenho. Para saber mais, acesse:

CRUZ, T. Ponto de fuga: quantos utilizar em um desenho. *Viva Decora*, 4 fev. 2019. Disponível em: <https://bit.ly/2K1QvwU>. Acesso em: 30 mar. 2023.

Definimos os seguintes valores para construir a matriz de transformação dessa projeção:

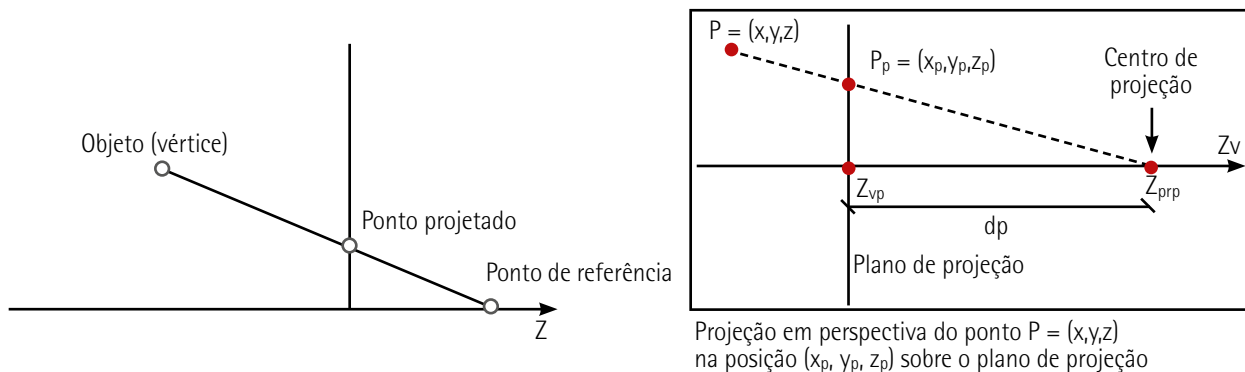


Figura 83

A equação matricial de transformação para a projeção com perspectiva é dada por:

$$\begin{bmatrix} x_h \\ y_h \\ z_h \\ h \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{-Z_{vp}}{d_p} & 0 \\ 0 & 0 & -\frac{1}{d_p} & \frac{Z_{prp}}{d_p} \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Onde:

$$h = \frac{z_{prp} - z}{d_p} \quad x_p = \frac{x_h}{h}, y_p = \frac{y_h}{h}$$

Observe a seguir duas situações:

Exemplo 1

Avalie as afirmações a seguir e assinale a única alternativa verdadeira.

- I – As projeções oblíquas cabinets seguem as mesmas regras das cavaleiras, apenas as dimensões ao longo do eixo z, perpendiculares à tela, é que sofrem um fator de escala de $\frac{1}{2}$.
- II – As projeções ortogonais isométricas preservam a mesma angulação entre os eixos coordenados.
- III – As projeções cilíndricas são de três tipos, segundo a quantidade de pontos de fuga.

É correto apenas o que se destaca em:

- A) I
- B) I e II.
- C) II.
- D) I e III.
- E) III.

Resolução

As afirmações I e II são verdadeiras. A III é falsa, pois são as projeções cônicas que são classificadas segundo o número de pontos de fuga, e não as cilíndricas.

Conclui-se que a alternativa correta é a B.

Considere as figuras, projeções de um dado de jogo comum, e avalie as afirmativas seguintes:

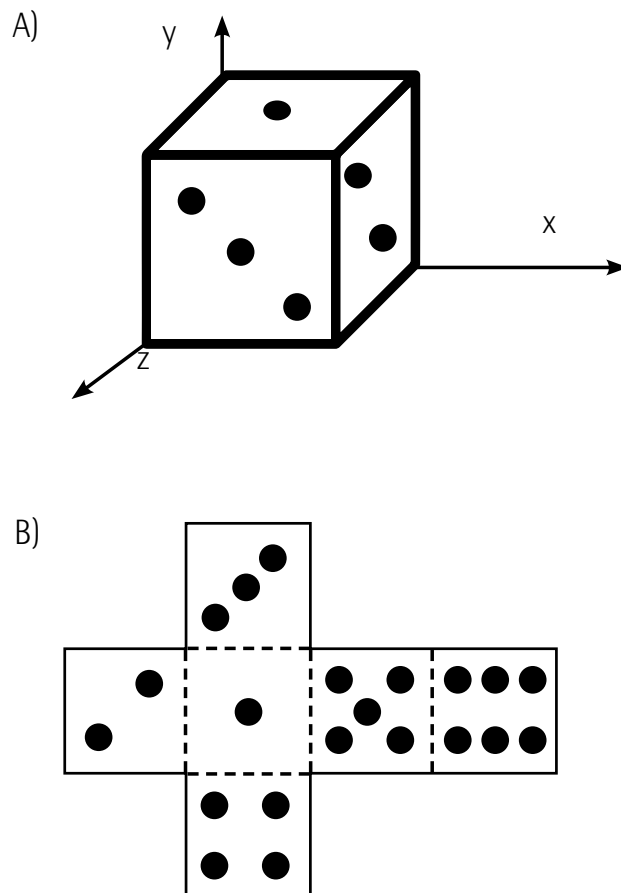


Figura 84

- I – Em um dado comum, segundo a figura, a soma dos pontos das faces opostas é igual a 7.
- II – A projeção (A) corresponde, aproximadamente, a uma projeção ortogonal cabinet de 45° .
- III – A planificação representada em (B) equivale a projeções cônicas em múltiplas faces ortogonais.
- IV – Partindo da figura (A), uma rotação de 90° em x, seguida de outra de 90° em z, expõe as faces de 1, 3 e 5.

É correto apenas o que se destaca em:

- A) Todas estão corretas.
- B) I.
- C) I, II e III.
- D) I e II.
- E) Nenhuma das afirmativas está correta.

Analisemos a questão.

A afirmativa I é verdadeira, pois a soma dos pontos das faces opostas em um dado comum é igual a 7, como pode ser comprovado pela projeção na parte B da figura.

Embora a figura A seja, de fato, uma projeção cabinet, pois a dimensão em z está reduzida pela metade, a afirmativa II é falsa, pois a projeção cabinet não é ortogonal (ou ortográfica), mas oblíqua.

Podemos considerar que a planificação em B seja equivalente a uma projeção em múltiplas faces, embora isso não seja completamente verdade, mas a afirmativa III é falsa, porque projeções em múltiplas faces são projeções ortogonais, e não cônicas.

Por fim, a afirmativa IV é falsa, porque a cadeia de transformações sugerida expõe as faces 1, 2 e 3, como pode ser visto na figura a seguir.

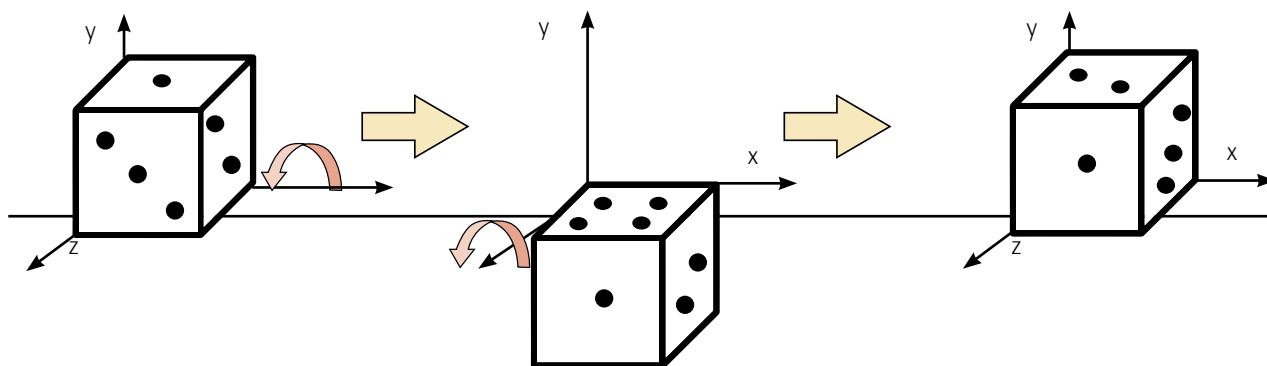


Figura 85

Assim, a alternativa correta é a B.

8.4 Transformações em três dimensões

Referem-se à aplicação de operações geométricas em objetos tridimensionais em um espaço 3D, resultando em uma nova configuração do objeto. Essas transformações podem incluir operações como rotação, translação e escala.

Em computação gráfica, as transformações em 3D são comumente representadas por meio de matrizes de transformação 4x4, que permitem a manipulação de pontos em coordenadas homogêneas. As coordenadas homogêneas adicionam uma dimensão extra ao espaço tridimensional, permitindo que as transformações sejam representadas de maneira uniforme para todos os tipos de transformações.

As transformações em 3D podem ser divididas em dois tipos principais: geométricas e de visualização. As primeiras alteram a geometria dos objetos tridimensionais, enquanto as de visualização alteram a perspectiva do observador em relação ao objeto.

Exemplos de transformações em 3D incluem:

- **Rotação:** girar um objeto em torno de um ou mais eixos.
- **Translação:** mover um objeto de um ponto para outro no espaço 3D.
- **Escala:** aumentar ou diminuir o tamanho de um objeto em relação a um ponto de referência.
- **Cisalhamento:** distorcer a forma de um objeto de acordo com um eixo específico.
- **Projeção:** representar um objeto tridimensional em uma imagem 2D, como em uma câmera ou tela de computador.

As transformações em três dimensões em computação gráfica são operações matemáticas que alteram a posição, a orientação e o tamanho de objetos 3D. As transformações básicas são rotação, translação e escala, mas outras também podem ser usadas, como cisalhamento e projeções. Elas são representadas por matrizes de transformação homogênea 4x4.

A matriz 4x4 contém informações de rotação, translação e escala em três dimensões, bem como informações de projeção e outras transformações. Ela é representada da seguinte maneira:

$$\begin{bmatrix} R_{11} & R_{12} & R_{13} & T_x \\ R_{21} & R_{22} & R_{23} & T_y \\ R_{31} & R_{32} & R_{33} & T_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Figura 86

Onde R_{ij} é a rotação na direção i em relação à direção j , T_x , T_y e T_z são os componentes de translação em x , y e z , respectivamente.

Destacaremos a seguir exemplos de transformações em três dimensões:

Na rotação em torno do eixo X , a matriz de rotação pode ser representada como

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(a) & -\sin(a) & 0 \\ 0 & \sin(a) & \cos(a) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Figura 87

Onde a é o ângulo de rotação em radianos.

Em translação em x, y e z, a matriz de translação pode ser representada como

$$\begin{bmatrix} 1 & 0 & 0 & T_x \\ 0 & 1 & 0 & T_y \\ 0 & 0 & 1 & T_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Figura 88

Onde T_x , T_y e T_z são os componentes de translação em x, y e z, respectivamente.

Na escala em x, y e z, a matriz de escala pode ser representada como

$$\begin{bmatrix} S_x & 0 & 0 & 0 \\ 0 & S_y & 0 & 0 \\ 0 & 0 & S_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Figura 89

Onde S_x , S_y e S_z são os fatores de escala em x, y e z, respectivamente.

Em cisalhamento em x e y, a matriz de cisalhamento pode ser representada como

$$\begin{bmatrix} 1 & Sh_x & 0 & 0 \\ Sh_y & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Figura 90

Onde Sh_x e Sh_y são os fatores de cisalhamento em x e y, respectivamente.



Resumo

Ao estudar a síntese de cores, vimos que entre os sistemas de cores os mais usados são o RGB e CMY ou CMYK.

O RGB imita a visão humana e é um sistema aditivo, porque as cores são criadas pela adição de luz de diferentes cores primárias em um fundo escuro ou preto.

O sistema CMY ou CMYK, usado nas impressoras, é um sistema de cor subtrativo, pois representa as cores através da subtração de cores de luz branca ou de uma fonte de luz.

Depois, destacamos as transformações geométricas em duas e três dimensões, que são operações matemáticas aplicadas sobre objetos geométricos em um espaço bidimensional ou tridimensional, a fim de alterar sua posição, tamanho, orientação ou forma.

Em seguida, analisamos a concatenação de transformações geométricas. Trata-se de uma técnica utilizada para aplicar diversas transformações a um objeto em uma única operação. Isso é feito multiplicando as matrizes de transformação de cada operação em ordem sequencial.

Por fim, estudamos a representação e modelagem de objetos 2D e 3D.



Exercícios

Questão 1. (IDIB 2019, adaptada) Sobre os sistemas de cores RGB e CMYK, julgue as afirmativas a seguir.

I – Ambos são o oposto físico/matemático um do outro.

II – O sistema RGB é formado por apenas três cores: o vermelho, o verde e o azul.

III – O sistema CMYK é usado na fotografia, no cinema, no vídeo, na televisão e na tela dos computadores.

IV – A letra K, do CMYK, tanto significa preto (black) como chave (key), uma vez que se trata da cor usada para interferir nos detalhes na impressão.

É correto o que se afirma em:

A) I, apenas.

B) II e III, apenas.

C) III e IV, apenas.

D) I, II e IV, apenas.

E) I, II, III e IV.

Resposta correta: alternativa D.

Análise das afirmativas

I – Afirmativa correta.

Justificativa: a síntese de cores é o processo de combinar diferentes cores para criar uma imagem. A síntese de cores aditiva é utilizada em dispositivos que emitem luz, como monitores, e tem como cores primárias o vermelho, o verde e o azul. Já a síntese de cores subtrativa é usada em dispositivos que refletem luz, como as tintas da impressora, e tem como cores primárias o ciano, o magenta e o amarelo.

Os sistemas de cores, em computação gráfica, são modelos matemáticos responsáveis por criar a aparência visual de objetos e as cenas em uma imagem digital.

O modelo de cores aditivo mais comum é o RGB (Red, Green, Blue), cujas cores são representadas por um conjunto de três valores, que indicam as intensidades dos componentes de luz vermelha, verde e azul em determinado pixel de uma imagem.

O modelo de cores subtrativo mais comum é o CMYK (Cyan, Magenta, Yellow, Key/Black), cujas cores primárias podem ser combinadas para formar a cor preta. No entanto, como a combinação de cores subtrativas nem sempre é perfeita, uma cor preta adicional é inserida para garantir uma cor preta mais sólida.

Os modelos RGB e CMYK são considerados modelos opostos, já que o primeiro trabalha de forma aditiva, a partir de cores primárias, com dispositivos emissores de luz, e o segundo trabalha de forma subtrativa, a partir de cores primárias e da cor preta, com elementos refletivos de luz.

II – Afirmativa correta.

Justificativa: o sistema RGB trabalha com composições das cores vermelho, verde e azul, que representam as cores primárias para dispositivos emissores de luz.

III – Afirmativa incorreta.

Justificativa: o sistema CMYK é utilizado em sistemas que trabalham a partir de partículas pigmentadas, como as tintas da impressora, que refletem a luz incidente.

IV – Afirmativa correta.

Justificativa: a letra K, da sigla CMYK, significa tanto preto (black) quanto chave (key), sendo que se trata da cor utilizada para interferir nos detalhes na impressão. Um pigmento preto é adicionado aos pigmentos das cores primárias porque seria difícil imprimir detalhes muito finos em preto, por meio da superposição de três tintas. A combinação de cores de forma subtrativa nem sempre é perfeita, portanto, um pigmento preto adicional garante uma cor preta mais sólida. Além disso, a tinta preta é mais barata do que o preço da quantidade de tinta necessária para gerar preto por meio da combinação das outras três cores. Por esses motivos, a cor preta é adicionada separadamente aos cartuchos, toners ou tanques das impressoras, sendo considerada a cor chave no modelo CMYK.

Questão 2. (UECE-CEV 2018, adaptada) Cada tipo de perspectiva existente mostra o objeto de um modo diferente. Na representação de um cubo, observe os tipos de perspectivas indicados a seguir.

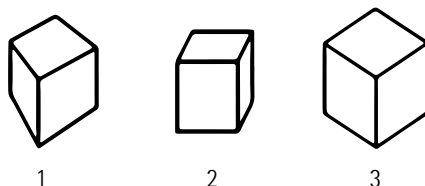


Figura 91

Assinale a alternativa que apresenta a correta relação entre a figura e o tipo de perspectiva utilizado.

A) 1. cavaleira; 2. cônica; 3. isométrica.

B) 1. cônica; 2. isométrica; 3. cabinet.

C) 1. cônica; 2. cabinet; 3. isométrica.

D) 1. isométrica; 2. cavaleira; 3. cônica.

E) 1. cabinet; 2. isométrica; 3. cônica.

Resposta correta: alternativa C.

Análise da questão

No cubo representado pela perspectiva 1, vemos o tipo de projeção cujas arestas do objeto convergem em direção a um ponto de fuga. Nesse caso, as arestas do objeto não são representadas de forma paralela. Esse tipo de projeção assemelha-se bastante à forma como enxergamos objetos tridimensionais ao nosso redor e é denominado projeção **cônica**.

No cubo representado pela perspectiva 2, vemos a face frontal do objeto em destaque, sendo que suas arestas são representadas nas dimensões reais, em ângulos de 0 e 90°. Já as arestas laterais são reduzidas à metade de seu tamanho original, sendo representadas de forma angulada. Essa perspectiva é denominada **cabinet**, no contexto da computação gráfica. Na área de desenho técnico projetivo, a perspectiva cabinet pode ser considerada um subtipo da perspectiva cavaleira.

No cubo representado pela perspectiva 3, vemos as arestas do objeto paralelas entre si, com destaques parecidos para a vista frontal e para a vista lateral do objeto. Esse tipo de perspectiva mantém as mesmas proporções de comprimento, largura e altura do objeto indicado. Temos, nesse caso, uma perspectiva isométrica, que é um caso particular de projeção cilíndrica ortogonal, em que os eixos tridimensionais se encontram a 120° entre si.

REFERÊNCIAS

Textuais

ANNIBAL, H. J. *Computação gráfica: fundamentos da informática*. São Paulo: LTC, 2006.

AZEVEDO, E.; CONCI, A. *Computação gráfica: geração de imagem*. Rio de Janeiro: Campus, 2003a.

AZEVEDO, E.; CONCI, A. *Computação gráfica: teoria e prática*. v. 2. Rio de Janeiro: Campus, 2003b.

CALLIOLI, C. A. *Álgebra linear e aplicações*. 6. ed. Santos: Atual, 2009.

CRUZ, T. Ponto de fuga: quantos utilizar em um desenho. *Viva Decora*, 4 fev. 2019. Disponível em: <https://bit.ly/2K1QvwU>. Acesso em: 30 mar. 2023.

DELGADO, J.; FRENSEL, K.; CRISSA, L. *Geometria analítica e cálculo vetorial I*. Rio de Janeiro: Universidade Federal Fluminense, 2012.

FRIENDLY, M.; DENIS, D. J. *Marcos na história da visualização de dados*. Tradução: Mário Kanno. Toronto: Universidade de York, [s.d.].

INSUA, H. G. *Álgebra linear*. São Paulo: Sol, 2022.

MACHADO, F. C. *Primeiro projeto de análise numérica II*. Curvas de Bézier e desenho de fontes tipográficas. Campinas: Unicamp, 2013. Disponível em: <https://bit.ly/3GhE8eA>. Acesso em: 30 mar. 2023.

SANTOS, R. J. *Um curso de geometria analítica e álgebra linear*. Minas Gerais: UFMG, 2001.

SZESZ JÚNIOR, A. et al. *A matemática e sua aplicabilidade na engenharia de computação*. Universidade Estadual de Ponta Grossa, [s.d.]. Disponível em: <https://bit.ly/40vvRfb>. Acesso em: 28 mar. 2023.

WJEC FURTHER MATHEMATICS. *Polar coordinates*. [s.d.]. Disponível em: <https://bit.ly/3G4iaLU>. Acesso em: 30 mar. 2023.



Handwriting practice lines consisting of 30 horizontal lines. Each line is preceded by a small blue dot on the left margin, serving as a starting point for letter formation. The lines are evenly spaced and extend across the width of the page.



Handwriting practice lines consisting of 30 horizontal blue lines. The first line is a solid blue line, and the subsequent 29 lines are pairs of dashed blue lines, providing a guide for letter height and placement.



Informações:
www.sepi.unip.br ou 0800 010 9000