Transformações Geométricas

Jefersson Alex dos Santos

jefersson@dcc.ufmg.br

DCC
DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO

UFMG

[1] Baseado nas aulas do Prof. Alexandre Xavier Falcão

Introdução

UFMG

- Nesta aula vamos considerar **transformações geométricas** ϕ em um **espaço afim** sobre os spels $p \in D_I$ de uma imagem $\hat{I} = (D_I, \vec{I})$.

Introdução

UFMG

- Nesta aula vamos considerar **transformações geométricas** ϕ em um **espaço afim** sobre os spels $p \in D_I$ de uma imagem $\hat{I} = (D_I, \vec{I})$.
- O espaço afim é uma generalização do espaço Euclideano, que inclui pontos, vetores, e certas operações entre eles, tais como adição e multiplicação por um escalar.

Introdução

UFMG

- Nesta aula vamos considerar **transformações geométricas** ϕ em um **espaço afim** sobre os spels $p \in D_I$ de uma imagem $\hat{I} = (D_I, \vec{I})$.
- O espaço afim é uma generalização do espaço Euclideano, que inclui pontos, vetores, e certas operações entre eles, tais como adição e multiplicação por um escalar.
- Uma transformação geométrica mapeia um ponto (ou vetor) em um outro ponto (ou vetor) do espaço afim (e.g., translação, rotação, escalamento, e projeção).

Introdução

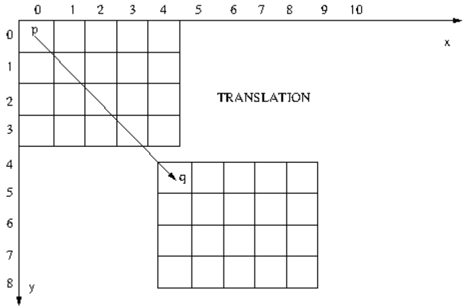
UFMG

- Uma transformação ϕ (e.g., translação) sobre um spel $p \in D_I \subset Z^n \subset \Re^n$ gera um novo ponto $q = \phi(p) \in \Re^n$.

Introdução

UFMG

- Uma transformação ϕ (e.g., translação) sobre um spel $p \in D_I \subset Z^n \subset \Re^n$ gera um novo ponto $q = \phi(p) \in \Re^n$.
- Se aplicarmos ϕ para todo spel em D_I, a imagem $\hat{I} = (D_I, \vec{I})$ será mapeada em um **domínio real** $D_J \subset \Re^n$, gerando $\hat{J} = (D_J, \vec{J})$ tal que para todo $q \in D_J$ existe um $p = \phi^{-1}(q) \in D_I \subset Z^n$ cujo $\vec{I}(p) = \vec{J}(q)$.



Introdução

UFMG

- Para obtermos uma imagem $\hat{J} = (D_J, \vec{J})$ com **domínio inteiro** $D_J \subset Z^n$, devemos aplicar a inversa $\phi^{-1}(q)$ em todo $q \in D_J$, obtendo $p = \phi^{-1}(q) \in \Re^n$.

Introdução

UFMG

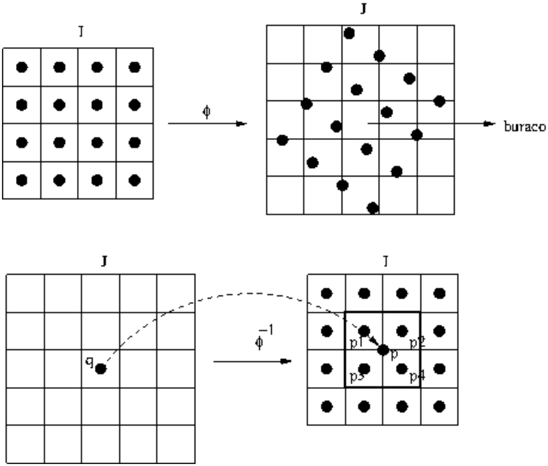
- Para obtermos uma imagem $\hat{J} = (D_J, \vec{J})$ com **domínio inteiro** $D_J \subset Z^n$, devemos aplicar a inversa $\phi^{-1}(q)$ em todo $q \in D_J$, obtendo $p = \phi^{-1}(q) \in \Re^n$.
- Depois o valor $\vec{J}(q) = \vec{I}(p)$ é obtido por **interpolação** dos valores conhecidos $\vec{I}(p_i)$ para $p_i \in D_I \subset Z^n$ em uma dada adjacência $\mathscr{A}(p) \subset D_I$.

Introdução

UFMG

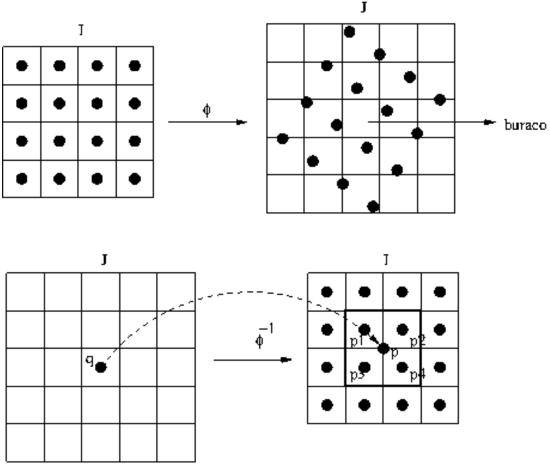
- Para obtermos uma imagem $\hat{J} = (D_J, \vec{J})$ com **domínio inteiro** $D_J \subset Z^n$, devemos aplicar a inversa $\phi^{-1}(q)$ em todo $q \in D_J$, obtendo $p = \phi^{-1}(q) \in \Re^n$.
- Depois o valor $\vec{J}(q) = \vec{I}(p)$ é obtido por **interpolação** dos valores conhecidos $\vec{I}(p_i)$ para $p_i \in D_I \subset Z^n$ em uma dada adjacência $\mathscr{A}(p) \subset D_I$.
- Esta estratégia também evita a formação de “buracos” na imagem transformada, já que seu domínio deve ser inteiro.

Introdução



Introdução

UFMG



A adjacência $\mathscr{A}$ é definida pelo piso e teto das coordenadas reais de $p \in D_I$.

Introdução

UFMG

Nesta aula iremos estudar as seguintes transformações geométricas:

- Translação e escalamento.
- Rotação em torno da origem e eixo principal (x, y, ou z).
- Rotação em torno de ponto arbitrário.
- Reflexão
- Cisalhamento

Coordenadas Homogêneas

UFMG

- Permite que as transformações espaciais sejam realizadas por meio de multiplicação de matrizes
- Facilita a combinação de várias transformações em um resultado composto

Um ponto que tem suas coordenadas expressas por (x,y,z) é descrito por coordenadas homogêneas como (wx,wy,wz,w), sendo w um valor diferente de zero
Exemplo: $(3,2,8,4)$ e $(6,4,16,8)$ são representações diferentes para o mesmo ponto

Transformação para coordenadas cartesianas

(1) Divisão dos primeiros componentes pelo último e (2) remoção do último

Transformações Afins

UFMG

Affine Transform

- Transformações lineares e translações. Outros grupos: Euclideano, Projetivo
- Preservam paralelismo entre retas e planos
- Não preservam comprimentos, distâncias, áreas, volumes, ângulos, ou perpendicularidade

Transformações Afins

UFMG

Affine Transform

- Transformações lineares e translações. Outros grupos: Euclideano, Projetivo
- Preservam paralelismo entre retas e planos
- Não preservam comprimentos, distâncias, áreas, volumes, ângulos, ou perpendicularidade

Podem ser expressas como:

$$
\begin{aligned}
x_q &= ax_p + by_p + cz_p + j \\
y_q &= dx_p + ey_p + fz_p + k \\
z_q &= gx_p + hy_p + iz_p + l
\end{aligned}
$$

UFMG

Transformações Afins

Affine Transform

- Transformações lineares e translações. Outros grupos: Euclideano, Projetivo
- Preservam paralelismo entre retas e planos
- Não preservam comprimentos, distâncias, áreas, volumes, ângulos, ou perpendicularidade

Podem ser expressas como:

$$\begin{aligned} x_q &= ax_p + by_p + cz_p + j \\ y_q &= dx_p + ey_p + fz_p + k \\ z_q &= gx_p + hy_p + iz_p + l \end{aligned}$$

Ou na forma matricial, utilizando **coordenadas homogêneas**:

$$\begin{bmatrix} x_q \\ y_q \\ z_q \\ 1 \end{bmatrix} = \begin{bmatrix} a & b & c & j \\ d & e & f & k \\ g & h & i & l \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_p \\ y_p \\ z_p \\ 1 \end{bmatrix}$$

Roteiro da Aula

UFMG

1 Translação

2 Escalamento

3 Rotação

4 Reflexão

5 Cisalhamento

6 Interpolação

Translação

UFMG

Seja $q = \phi(p) = (x_q, y_q, z_q)$ o ponto $p = (x_p, y_p, z_p)$ transladado do vetor $\vec{t} = (t_x, t_y, t_z)$.
Temos que:

$$\begin{bmatrix} x_q \\ y_q \\ z_q \end{bmatrix} = \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} + \begin{bmatrix} x_p \\ y_p \\ z_p \end{bmatrix}$$

Translação

UFMG

Se representarmos os pontos p e q em **coordenadas homogêneas**, $p = (x_p, y_p, z_p, 1)$ e $q = (x_q, y_q, z_q, 1)$, a translação passa a ser multiplicativa e pode ser facilmente combinada com as demais transformações geométricas, que são multiplicativas.

Translação

UFMG

Se representarmos os pontos p e q em **coordenadas homogêneas**, $p = (x_p, y_p, z_p, 1)$ e $q = (x_q, y_q, z_q, 1)$, a translação passa a ser multiplicativa e pode ser facilmente combinada com as demais transformações geométricas, que são multiplicativas.

$$\begin{bmatrix} x_q \\ y_q \\ z_q \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_p \\ y_p \\ z_p \\ 1 \end{bmatrix}$$

Para aplicar a inversa, basta transladar de $(-t_x, -t_y, -t_z, 1)$.

Roteiro da Aula

UFMG

1. Translação
2. **Escalamento**
3. Rotação
4. Reflexão
5. Cisalhamento
6. Interpolação

Escalamento

UFMG

Fatores s_x, s_y e s_z podem ser aplicados às coordenadas dos pontos para aumentar/reduzir o tamanho de um objeto (imagem), ou refletí-lo em relação a um dos planos de coordenadas.

$$\begin{bmatrix} x_q \\ y_q \\ z_q \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_p \\ y_p \\ z_p \\ 1 \end{bmatrix}$$

Fatores maiores que 0 e menores que 1 ocasionam redução de tamanho, fatores maiores que 1 ocasionam aumento, e fatores menores que 0 ocasionam reflexão. A inversa $\mathbf{S}^{-1}(s_x, s_y, s_z)$ é $\mathbf{S}(1/s_x, 1/s_y, 1/s_z)$.

Roteiro da Aula

1 Translação

2 Escalamento

3 Rotação

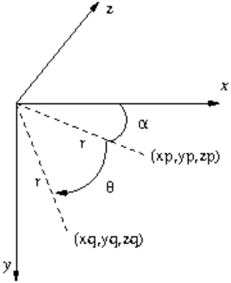
4 Reflexão

5 Cisalhamento

6 Interpolação

Rotação em torno da origem e eixo z

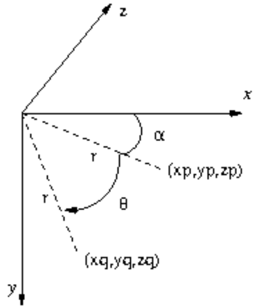
Seja $\vec{V} = (0, 0, 1, 1)$ o vetor que representa o eixo z com origem em $(0, 0, 0, 1)$, a rotação em torno da origem e eixo z



modifica apenas as coordenadas x e y dos pontos, seguindo a regra da mão direita (polegar direito na direção e sentido de $\vec{V}$, e os demais dedos girando para dentro da mão.).

Rotação em torno da origem e eixo z

Esta rotação é representada por uma matriz $\mathbf{R_z}(\theta)$ obtida das relações trigonométricas abaixo.



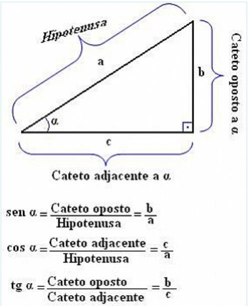
$$x_p = r\cos(\alpha)$$

UFMG

Rotação em torno da origem e eixo z

Esta rotação é representada por uma matriz $\mathbf{R_z}(\theta)$ obtida das relações trigonométricas abaixo.

Relações:

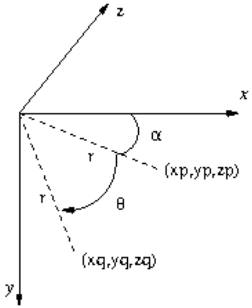


$$
\begin{aligned}
x_p &= r\cos(\alpha) \\
y_p &= r\sin(\alpha)
\end{aligned}
$$

Rotação em torno da origem e eixo z

UFMG

Esta rotação é representada por uma matriz $\mathbf{R_z}(\theta)$ obtida das relações trigonométricas abaixo.



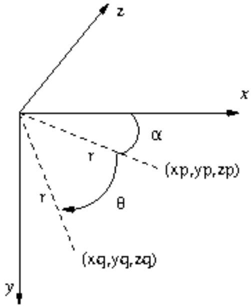
$$\begin{aligned} x_p &= r\cos(\alpha) \\ y_p &= r\sin(\alpha) \\ x_q &= r\cos(\theta+\alpha) \end{aligned}$$

Propriedade:
$\cos(a+b) =$
$\cos(a)\cos(b) - \sin(a)\sin(b)$

Rotação em torno da origem e eixo z

UFMG

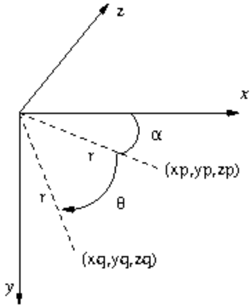
Esta rotação é representada por uma matriz $\mathbf{R_z}(\theta)$ obtida das relações trigonométricas abaixo.



$$
\begin{aligned}
x_p &= r\cos(\alpha) \\
y_p &= r\sin(\alpha) \\
x_q &= r\cos(\theta+\alpha) \\
x_q &= r\cos(\alpha)\cos(\theta) - r\sin(\alpha)\sin(\theta)
\end{aligned}
$$

Rotação em torno da origem e eixo z

Esta rotação é representada por uma matriz $\mathbf{R_z}(\theta)$ obtida das relações trigonométricas abaixo.

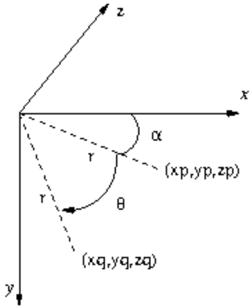


$$
\begin{aligned}
x_p &= r\cos(\alpha) \\
y_p &= r\sin(\alpha) \\
x_q &= r\cos(\theta+\alpha) \\
x_q &= r\cos(\alpha)\cos(\theta) - r\sin(\alpha)\sin(\theta) \\
x_q &= x_p\cos(\theta) - y_p\sin(\theta)
\end{aligned}
$$

Rotação em torno da origem e eixo z

UFMG

Esta rotação é representada por uma matriz $\mathbf{R_z}(\theta)$ obtida das relações trigonométricas abaixo.



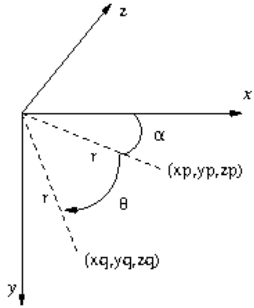
$$
\begin{aligned}
x_p &= r\cos(\alpha) \\
y_p &= r\sin(\alpha) \\
x_q &= r\cos(\theta+\alpha) \\
x_q &= r\cos(\alpha)\cos(\theta) - r\sin(\alpha)\sin(\theta) \\
x_q &= x_p\cos(\theta) - y_p\sin(\theta) \\
y_q &= r\sin(\theta+\alpha)
\end{aligned}
$$

Propriedade:
$\sin(a+b) =$
$\cos(a)\sin(b) + \sin(a)\cos(b)$

UFMG

Rotação em torno da origem e eixo z

Esta rotação é representada por uma matriz $\mathbf{R_z}(\theta)$ obtida das relações trigonométricas abaixo.

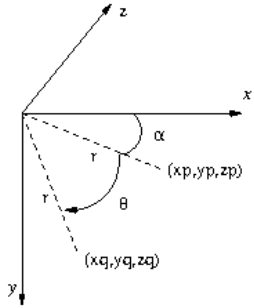


$$
\begin{aligned}
x_p &= r\cos(\alpha) \\
y_p &= r\sin(\alpha) \\
x_q &= r\cos(\theta+\alpha) \\
x_q &= r\cos(\alpha)\cos(\theta) - r\sin(\alpha)\sin(\theta) \\
x_q &= x_p\cos(\theta) - y_p\sin(\theta) \\
y_q &= r\sin(\theta+\alpha) \\
y_q &= r\cos(\alpha)\sin(\theta) + r\sin(\alpha)\cos(\theta)
\end{aligned}
$$

Rotação em torno da origem e eixo z

UFMG

Esta rotação é representada por uma matriz $\mathbf{R_z}(\theta)$ obtida das relações trigonométricas abaixo.

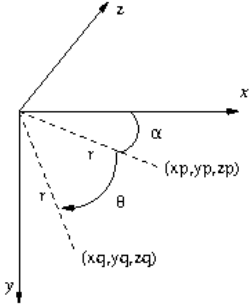


$$
\begin{aligned}
x_p &= r\cos(\alpha) \\
y_p &= r\sin(\alpha) \\
x_q &= r\cos(\theta+\alpha) \\
x_q &= r\cos(\alpha)\cos(\theta) - r\sin(\alpha)\sin(\theta) \\
x_q &= x_p\cos(\theta) - y_p\sin(\theta) \\
y_q &= r\sin(\theta+\alpha) \\
y_q &= r\cos(\alpha)\sin(\theta) + r\sin(\alpha)\cos(\theta) \\
y_q &= x_p\sin(\theta) + y_p\cos(\theta)
\end{aligned}
$$

Rotação em torno da origem e eixo z

UFMG

Esta rotação é representada por uma matriz $\mathbf{R_z}(\theta)$ obtida das relações trigonométricas abaixo.



$$
\begin{aligned}
x_p &= r\cos(\alpha) \\
y_p &= r\sin(\alpha) \\
x_q &= r\cos(\theta+\alpha) \\
x_q &= r\cos(\alpha)\cos(\theta) - r\sin(\alpha)\sin(\theta) \\
x_q &= x_p\cos(\theta) - y_p\sin(\theta) \\
y_q &= r\sin(\theta+\alpha) \\
y_q &= r\cos(\alpha)\sin(\theta) + r\sin(\alpha)\cos(\theta) \\
y_q &= x_p\sin(\theta) + y_p\cos(\theta) \\
z_q &= z_p
\end{aligned}
$$

Rotação em torno da origem e eixo z

$$\begin{bmatrix} x_q \\ y_q \\ z_q \\ 1 \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 & 0 \\ \sin(\theta) & \cos(\theta) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_p \\ y_p \\ z_p \\ 1 \end{bmatrix}$$

Rotações em torno da origem e eixos x e y

UFMG

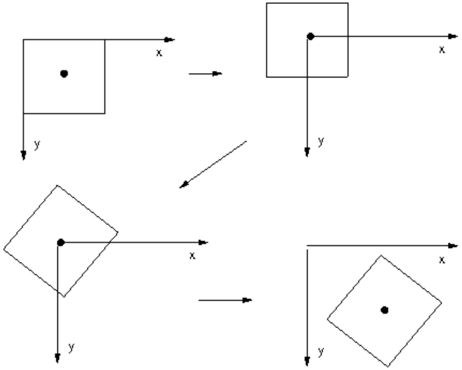
As rotações em torno da origem e eixos x e y são obtidas de forma similar.

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

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

Rotação em torno de um ponto arbitrário

O objeto (imagem) deve ser transladado para que seu centro geométrico fique na origem do sistema de coordenadas. Após aplicar a rotação, transladamos de volta o objeto (imagem) evitando cortes de cena.



Rotação em torno de um ponto arbitrário

UFMG

Por exemplo, a rotação $\mathbf{R}_x(\theta)$ de um ângulo θ em torno do vetor $\vec{V} = (1,0,0,1)$ e do centro geométrico $(x_c, y_c, z_c, 1)$ de um objeto (imagem) é dada por

$$\begin{bmatrix} 1 & 0 & 0 & d/2 \\ 0 & 1 & 0 & d/2 \\ 0 & 0 & 1 & d/2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\theta) & -\sin(\theta) & 0 \\ 0 & \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & -x_c \\ 0 & 1 & 0 & -y_c \\ 0 & 0 & 1 & -z_c \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

onde d é a diagonal do objeto (imagem).

Roteiro da Aula

UFMG

1 Translação

2 Escalamento

3 Rotação

4 Reflexão

5 Cisalhamento

6 Interpolação

Reflexão

UFMG

Apenas valores da coordenada do eixo refletido sofrem alteração (inversão de sinal)

$$\mathbf{E_{yz}}(\theta) = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{E_{xz}}(\theta) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{E_{xy}}(\theta) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Roteiro da Aula

UFMG

1 Translação

2 Escalamento

3 Rotação

4 Reflexão

5 Cisalhamento

6 Interpolação

Cisalhamento

Shear

UFMG

- Altera as coordenadas dos pontos de acordo com uma função de direção
- Causa uma deformação dos objetos presentes na imagem
- Altera os pontos em direção paralela a um plano

Com respeito ao plano yz, a transformação pode ser expressa como:

$$\begin{aligned} x_q &= x_p \\ y_q &= y_p + c_y x_p \\ z_q &= z_p + c_z x_p \end{aligned}$$

Cisalhamento

Shear

UFMG

Forma matricial usando coordenadas homogêneas:

$$\mathbf{C_{yz}}(\theta) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ c_y & 1 & 0 & 0 \\ c_z & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{C_{xz}}(\theta) = \begin{bmatrix} 1 & c_x & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & c_z & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{C_{xy}}(\theta) = \begin{bmatrix} 1 & 0 & c_x & 0 \\ 0 & 1 & c_y & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Roteiro da Aula

UFMG

1 Translação

2 Escalamento

3 Rotação

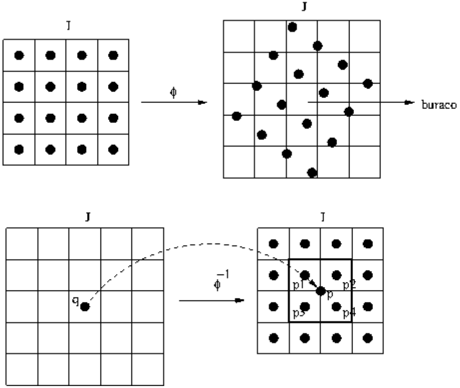
4 Reflexão

5 Cisalhamento

6 Interpolação

Interpolação

Ao aplicar a transformação inversa para obter a transformação direta, a interpolação dos valores da imagem original é adotada em uma adjacência $\mathscr{A}$.



A adjacência $\mathscr{A}$ é definida pelas coordenadas inteiras mais próximas, abaixo e acima, das coordenadas reais (x_p, y_p, z_p) de $p \in D_I$.

Interpolação pelo Vizinho Mais Próximo

UFMG

O valor da intensidade a ser atribuído ao spel p terá o mesmo valor do spel que estiver mais próximo na imagem original.

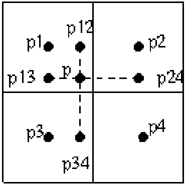
10	4	22
2	18	7
9	14	25

10	10	4	4	22	22
10	10	4	4	22	22
2	2	18	18	7	7
2	2	18	18	7	7
9	9	14	14	25	25
9	9	14	14	25	25

Interpolação Bilinear

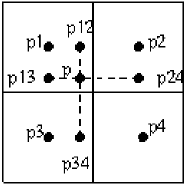
UFMG

- Assume que os valores dos spels variam linearmente em todas as direções
- Utiliza média ponderada de distâncias



Interpolação Bilinear

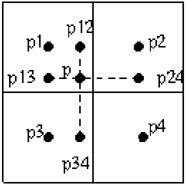
- Assume que os valores dos spels variam linearmente em todas as direções
- Utiliza média ponderada de distâncias



$$\vec{I}(p_{12}) = (x_p - x_{p_1})\vec{I}(p_2) + (x_{p_2} - x_p)\vec{I}(p_1)$$
$$\vec{I}(p_{34}) = (x_p - x_{p_3})\vec{I}(p_4) + (x_{p_4} - x_p)\vec{I}(p_3)$$

Interpolação Bilinear

- Assume que os valores dos spels variam linearmente em todas as direções
- Utiliza média ponderada de distâncias

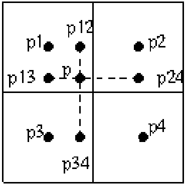


$$
\begin{aligned}
\vec{I}(p_{12}) &= (x_p - x_{p_1})\vec{I}(p_2) + (x_{p_2} - x_p)\vec{I}(p_1) \\
\vec{I}(p_{34}) &= (x_p - x_{p_3})\vec{I}(p_4) + (x_{p_4} - x_p)\vec{I}(p_3) \\
\vec{I}(p) &= (y_p - y_{p_{12}})\vec{I}(p_{34}) + (y_{p_{34}} - y_p)\vec{I}(p_{12})
\end{aligned}
$$

Interpolação Bilinear

UFMG

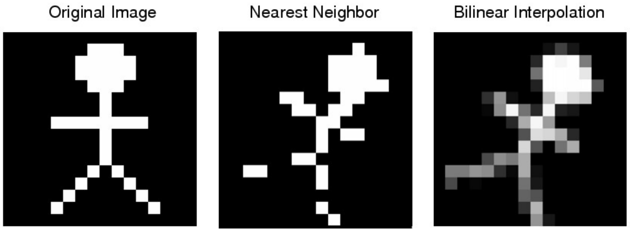
- Assume que os valores dos spels variam linearmente em todas as direções
- Utiliza média ponderada de distâncias



Características

- Reduz problemas causados pela interpolação pelo vizinho mais próximo
- Causa borramento devido a sua natureza de suavização

Comparação



UFMG

Comparação

(c)using nearest interpolation
(256x256)

(b)using nearest interpolation
(128x128)

(a)The original image
(64x64)

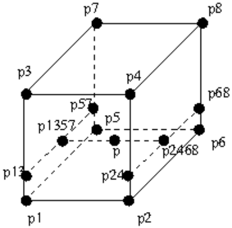
(e)using bilinear interpolation
(256x256)

(d)using bilinear interpolation
(128x128)

UFMG

Interpolação Bilinear

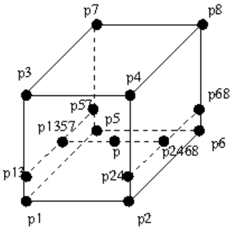
No caso 3D, temos



$$\begin{aligned}
\vec{I}(p_{24}) &= (y_p - y_{p_4})\vec{I}(p_2) + (y_{p_2} - y_p)\vec{I}(p_4) \\
\vec{I}(p_{68}) &= (y_p - y_{p_8})\vec{I}(p_6) + (y_{p_6} - y_p)\vec{I}(p_8) \\
\vec{I}(p_{13}) &= (y_p - y_{p_3})\vec{I}(p_1) + (y_{p_1} - y_p)\vec{I}(p_3) \\
\vec{I}(p_{57}) &= (y_p - y_{p_7})\vec{I}(p_5) + (y_{p_5} - y_p)\vec{I}(p_7)
\end{aligned}$$

Interpolação Bilinear

No caso 3D, temos

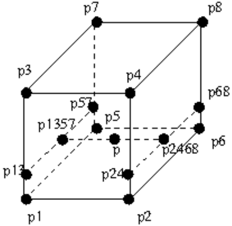


$$
\begin{aligned}
\vec{I}(p_{2468}) &= (z_p - z_{p_{24}})\vec{I}(p_{68}) + (z_{p_{68}} - z_p)\vec{I}(p_{24}) \\
\vec{I}(p_{1357}) &= (z_p - z_{p_{13}})\vec{I}(p_{57}) + (z_{p_{57}} - z_p)\vec{I}(p_{13})
\end{aligned}
$$

Interpolação Bilinear

UFMG

No caso 3D, temos



$$
\begin{aligned}
\vec{I}(p_{2468}) &= (z_p - z_{p_{24}})\vec{I}(p_{68}) + (z_{p_{68}} - z_p)\vec{I}(p_{24}) \\
\vec{I}(p_{1357}) &= (z_p - z_{p_{13}})\vec{I}(p_{57}) + (z_{p_{57}} - z_p)\vec{I}(p_{13}) \\
\vec{I}(p) &= (x_p - x_{p_{1357}})\vec{I}(p_{2468}) + (x_{p_{2468}} - x_p)\vec{I}(p_{1357})
\end{aligned}
$$

Interpolação Bilinear

UFMG

Se uma imagem $\hat{I} = (D_I, \vec{I})$ tem voxels de dimensões $(d_{x_1}, d_{y_1}, d_{z_1})$ e desejamos gerar por interpolação uma imagem $\hat{J} = (D_I, \vec{J})$ com voxels $(d_{x_2}, d_{y_2}, d_{z_2})$, então é mais rápido

- interpolar primeiro $\hat{I} = (D_I, \vec{I})$ ao longo de x, gerando $\hat{I}_1 = (D_I, \vec{I}_1)$ com tamanho de voxel $(d_{x_2}, d_{y_1}, d_{z_1})$,

Interpolação Bilinear

U F $\mathcal{m}$ G

Se uma imagem $\hat{I} = (D_I, \vec{I})$ tem voxels de dimensões $(d_{x_1}, d_{y_1}, d_{z_1})$ e desejamos gerar por interpolação uma imagem $\hat{J} = (D_I, \vec{J})$ com voxels $(d_{x_2}, d_{y_2}, d_{z_2})$, então é mais rápido

- interpolar primeiro $\hat{I} = (D_I, \vec{I})$ ao longo de x, gerando $\hat{I}_1 = (D_I, \vec{I}_1)$ com tamanho de voxel $(d_{x_2}, d_{y_1}, d_{z_1})$,
- depois interpolar $\hat{I}_1 = (D_I, \vec{I}_1)$ ao longo de y, gerando $\hat{I}_2 = (D_I, \vec{I}_2)$ com tamanho de voxel $(d_{x_2}, d_{y_2}, d_{z_1})$, e

Interpolação Bilinear

UFMG

Se uma imagem $\hat{I} = (D_I, \vec{I})$ tem voxels de dimensões $(d_{x_1}, d_{y_1}, d_{z_1})$ e desejamos gerar por interpolação uma imagem $\hat{J} = (D_I, \vec{J})$ com voxels $(d_{x_2}, d_{y_2}, d_{z_2})$, então é mais rápido

- interpolar primeiro $\hat{I} = (D_I, \vec{I})$ ao longo de x, gerando $\hat{I}_1 = (D_I, \vec{I}_1)$ com tamanho de voxel $(d_{x_2}, d_{y_1}, d_{z_1})$,
- depois interpolar $\hat{I}_1 = (D_I, \vec{I}_1)$ ao longo de y, gerando $\hat{I}_2 = (D_I, \vec{I}_2)$ com tamanho de voxel $(d_{x_2}, d_{y_2}, d_{z_1})$, e
- por fim interpolar $\hat{I}_2 = (D_I, \vec{I}_2)$ ao longo de z, gerando $\hat{J} = (D_I, \vec{J})$ com tamanho de voxel $(d_{x_2}, d_{y_2}, d_{z_2})$.

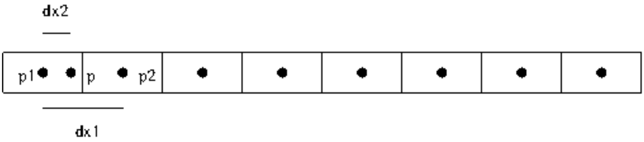
Interpolação Bilinear

UFMG

Por exemplo, a interpolação bilinear ao longo de x em uma dada linha y_p e fatia z_p é realizada para todo p de p_1 até o último voxel da linha, com incrementos d_{x_2}.

$$\begin{aligned} y_p &= y_{p_1} = y_{p_2} \\ z_p &= z_{p_1} = z_{p_2} \\ x_p &= x_{p_1} + d_{x_2} \\ \vec{I}(p) &= (x_p - x_{p_1})\vec{I}(p_2) + (x_{p_2} - x_p)\vec{I}(p_1) \end{aligned}$$

onde p_2 é o próximo voxel na linha após p.



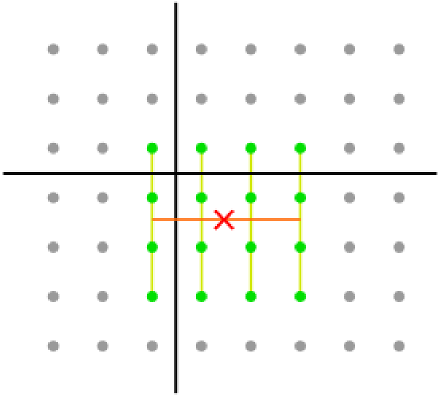
Interpolação Bilinear

Exemplo de escalamento e rotação em torno do centro da imagem e eixo z.

Interpolação Bicúbica

UFMG

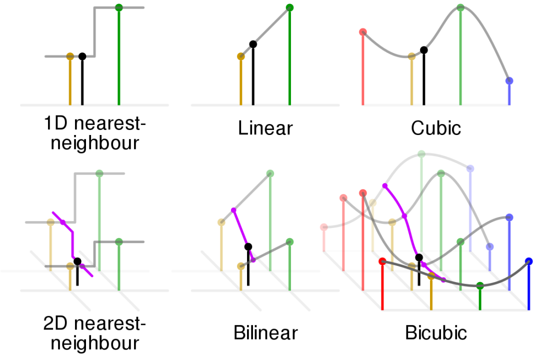
- Utiliza vizinhança 4×4 ao redor do spel p
- Emprega funções como a B-Spline cúbica



Interpolação Bicúbica

UFMG

- Utiliza vizinhança 4×4 ao redor do spel p
- Emprega funções como a B-Spline cúbica



Interpolação Bicúbica

UFMG

- Utiliza vizinhança 4×4 ao redor do spel p
- Emprega funções como a B-Spline cúbica

Características

- Não sofre do problema de bordas serrilhadas
- Não causa borramento
- Preserva detalhes finos na imagem