

# Interpolação e Registro de Imagens

Prof. Alexandre Xavier Falcão

Aula 05 - Parte III

## 1 Interpolação

As formas mais comuns de interpolação são por vizinho mais próximo (ordem 0) e linear (ordem 1). No primeiro caso, o brilho do vizinho mais próximo é associado ao ponto onde desejamos saber o brilho. No segundo caso, assumimos que o brilho da imagem varia linearmente ao longo de qualquer eixo principal. No caso 2D, seja  $D_J = MD_I$  uma transformação geométrica que leva pixels  $p$  de  $\hat{I}$  em pixels  $q$  de  $\hat{J}$  (Figure 1). Para evitar buracos, nós estamos interessados em estimar o brilho em pontos  $p$  de  $\hat{I}$  que são mapeados exatamente no centro de cada pixel  $q$  de  $\hat{J}$ . Estes pontos são encontrados com  $p = M^{-1}q$  para todo  $q \in D_J$ , tal que

$$\begin{aligned} J(q) &\leftarrow I(p) \\ I(p) &= \frac{d(p, q_{34})I(q_{12}) + d(p, q_{12})I(q_{34})}{d(q_{12}, q_{34})} \\ I(q_{12}) &= \frac{d(q_{12}, q_2)I(q_1) + d(q_{12}, q_1)I(q_2)}{d(q_1, q_2)} \\ I(q_{34}) &= \frac{d(q_{34}, q_4)I(q_3) + d(q_{34}, q_3)I(q_4)}{d(q_3, q_4)} \end{aligned}$$

onde  $d_{p,q_{12}} = y_p - \lfloor y_p$ ,  $d_{p,q_{34}} = \lceil y_p - y_p$ ,  $d_{q_{12},q_1} = d_{q_{34},q_3} = x_p - \lfloor x_p$ , e  $d_{q_{34},q_2} = d_{q_{34},q_4} = \lceil x_p - x_p$  de acordo com a Figura 1. Em 3D, o cálculo é similar levando em consideração os três eixos.

Imagens 3D são obtidas normalmente com amostragem anisotrópica. Para evitar distorções na visualização, a imagem é interpolada para voxels cúbicos (amostragem isotrópica) de lado igual à menor dimensão. A interpolação linear neste caso pode ser calculada de forma mais eficiente se simularmos um processo de reamostragem. Interpolamos ao longo de  $x$ , linha por linha, gerando uma primeira imagem intermediária, depois esta imagem é interpolada ao longo de  $y$ , coluna por coluna, gerando uma segunda imagem intermediária, e depois a segunda imagem é interpolada ao longo de  $z$ . Ao longo do eixo  $x$ , por exemplo, partimos do primeiro pixel  $q_1$  até o último  $q_n$  com intervalos  $d_x$  iguais à dimensão final desejada.

O pixel  $p$  equivalente ao  $k$ -ésimo pixel da direção  $x$  na imagem interpolada  $\hat{J}$  será obtido por  $p = q_1 + (k - 1)d_x\vec{i}$ . Sejam  $q_k = (\lfloor x_p, y_p, z_p)$  e  $q_{k+1} = (\lceil x_p, y_p, z_p)$ , o brilho  $J(p)$  será dado por

$$J(p) = \frac{d(p, q_k)I(q_k) + d(p, q_{k+1})I(q_{k+1})}{d(q_k, q_{k+1})}$$

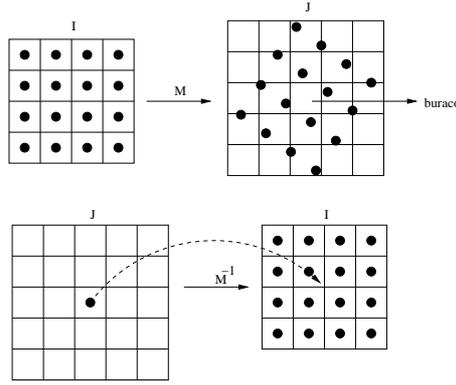


Figura 1: Buracos podem ser evitados por interpolação quando aplicamos  $D_J = M^{-1}D_I$  em vez de  $D_I = MD_J$ .

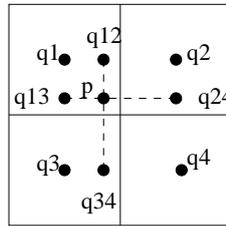


Figura 2: Os pixels  $q_1, q_2, q_3, q_4$  são os 4 pixels vizinhos mais próximos do ponto  $p$  em  $D_I$ .

## 2 Registro de imagens

Sejam  $\hat{I} = (D_I, I)$  e  $\hat{J} = (D_J, J)$  duas imagens que precisam ser representadas em um mesmo sistema de coordenadas. Por exemplo, desejamos encontrar o mapeamento  $M$  tal que  $D_I = MD_J$  e gerar  $\hat{R} = (D_I, J)$  como sendo a imagem  $\hat{J}$  transformada para o sistema de coordenadas de  $\hat{I}$ . Denominamos então  $\hat{I}$  como imagem fixa e  $\hat{J}$  como imagem móvel. Gerar  $\hat{R}$  é registrar as imagens  $\hat{I}$  e  $\hat{J}$ . Esta operação é necessária quando desejamos estudar alterações de uma propriedade física em uma dada região do espaço para diferentes instantes de tempo (registro intra-modalidade) ou combinar imagens de diferentes propriedades em uma mesma região do espaço (registro inter-modalidade). A aquisição separada das imagens  $\hat{I}$  e  $\hat{J}$  faz com que seus domínios não sejam os mesmos, mas devem possuir uma região em comum, onde o registro deve ser exato.

A transformação  $M$  pode ser rígida (envolver apenas translação e/ou rotação) ou deformável (envolver variação de escala e/ou outras deformações). Ela pode ser encontrada interativamente, a partir de pontos de correspondências marcados em  $\hat{I}$  e  $\hat{J}$ , ou automaticamente, a partir de *matching* entre características de pontos (linhas, regiões) extraídas das imagens. Sejam  $S_I$  e  $S_J$  pontos extraídos de  $\hat{I}$  e  $\hat{J}$ , respectivamente, a operação de *matching* envolve a busca do mapeamento  $M$  que maximiza ou minimiza uma **função critério** baseada em características de  $S_I$  e  $MS_J$ . Existem várias técnicas de busca para  $M$  (e.g., descendente de gradiente, programação dinâmica, busca exaustiva) e funções critério (e.g., informação mútua,

correlação, erro quadrático médio), as quais podem levar em conta as coordenadas dos pontos e/ou outras características desses pontos na imagem. Na próxima aula, vamos estudar técnicas de extração de características, um algoritmo de busca, e funções critério para registro automático de imagens. Nesta aula, vamos ver apenas o caso mais simples de registro interativo.

## 2.1 *Matching* direto entre pontos marcados nas imagens

Seja  $M$  a transformação 3D abaixo que leva pontos  $p = (x, y, z, 1)$  de  $D_J$  em pontos  $q = (x', y', z', 1)$  de  $D_I$ .

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} a & b & c & d \\ e & f & g & h \\ i & j & k & l \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Na primeira linha, por exemplo, temos que  $x' = ax + by + cz + d$ . Para encontrar as incógnitas  $a, b, c, d$  nós precisamos de 4 equações como esta. Ou seja, 4 correspondências entre  $q_i = (x'_i, y'_i, z'_i, 1)$  e  $p_i = (x_i, y_i, z_i, 1)$  para  $i = 1, 2, 3, 4$ . Estas correspondências devem ser não-colineares e podem ser usadas para encontrar as demais incógnitas.

A solução é obtida pela regra de Cramer, a qual calcula os determinantes das matrizes abaixo para encontrar cada incógnita. As demais incógnitas são encontradas de forma similar.

$$a = \frac{\begin{vmatrix} x'_1 & y_1 & z_1 & 1 \\ x'_2 & y_2 & z_2 & 1 \\ x'_3 & y_3 & z_3 & 1 \\ x'_4 & y_4 & z_4 & 1 \end{vmatrix}}{\begin{vmatrix} x_1 & y_1 & z_1 & 1 \\ x_2 & y_2 & z_2 & 1 \\ x_3 & y_3 & z_3 & 1 \\ x_4 & y_4 & z_4 & 1 \end{vmatrix}}$$

$$b = \frac{\begin{vmatrix} x_1 & x'_1 & z_1 & 1 \\ x_2 & x'_2 & z_2 & 1 \\ x_3 & x'_3 & z_3 & 1 \\ x_4 & x'_4 & z_4 & 1 \end{vmatrix}}{\begin{vmatrix} x_1 & y_1 & z_1 & 1 \\ x_2 & y_2 & z_2 & 1 \\ x_3 & y_3 & z_3 & 1 \\ x_4 & y_4 & z_4 & 1 \end{vmatrix}}$$

$$c = \frac{\begin{vmatrix} x_1 & y_1 & x'_1 & 1 \\ x_2 & y_2 & x'_2 & 1 \\ x_3 & y_3 & x'_3 & 1 \\ x_4 & y_4 & x'_4 & 1 \end{vmatrix}}{\begin{vmatrix} x_1 & y_1 & z_1 & 1 \\ x_2 & y_2 & z_2 & 1 \\ x_3 & y_3 & z_3 & 1 \\ x_4 & y_4 & z_4 & 1 \end{vmatrix}}$$

$$d = \frac{\begin{vmatrix} x_1 & y_1 & z_1 & x'_1 \\ x_2 & y_2 & z_2 & x'_2 \\ x_3 & y_3 & z_3 & x'_3 \\ x_4 & y_4 & z_4 & x'_4 \end{vmatrix}}{\begin{vmatrix} x_1 & y_1 & z_1 & 1 \\ x_2 & y_2 & z_2 & 1 \\ x_3 & y_3 & z_3 & 1 \\ x_4 & y_4 & z_4 & 1 \end{vmatrix}}$$

Após encontrar a transformação, nós podemos aplicar sua inversa  $M^{-1}$ , para evitar buracos, a todos os spels da aproximação  $\hat{R}$  e obter seus valores por interpolação linear entre os 6-adjacentes mais próximos em  $\hat{J}$ .

Quando o número de pontos correspondentes é maior que 4, a busca por  $M$  deve ser baseada em função critério.