

Warping Entre Imagens e Sua Aplicação à Visualização de Superfícies Topográficas¹

F.S. de SOUSA², M. MENEGUETTE Jr.³, Departamento de Matemática, Faculdade de Ciências e Tecnologia, UNESP, Cx.P. 957, 19060-900 Presidente Prudente, SP, Brasil.

Resumo. Quando se trabalha com representação de terrenos no computador, busca-se obter modelos mais realísticos do mesmo. Nesse sentido, este trabalho utiliza uma técnica de distorção de imagens, chamada warping por triangulação, aplicando-a à visualização tridimensional de superfícies topográficas, distorcendo a imagem, geralmente provida de fotos de satélite, em cima de seu respectivo modelo digital de terreno.

1. Introdução

No intuito de implementar uma ferramenta que fosse capaz de distorcer uma imagem digital, geralmente provida de fotos de satélite, sobre o modelo digital do terreno correspondente, foi estudada uma técnica bastante empregada em recursos de *warping* e *morphing*⁴ entre imagens. Antes, porém, é necessária uma certa conceituação teórica sobre o problema e sua descrição, levando-se em conta seus modelos matemáticos, visando uma resolução mais precisa e correta.

2. Modelo matemático de imagem

Para representar e manipular imagens no computador, deve-se definir modelos matemáticos adequados a esses objetivos. Quando se observa uma fotografia ou uma cena real, recebe-se de cada ponto um impulso luminoso, que associa uma informação de cor a esse ponto. Dessa forma, o modelo matemático mais natural para descrever uma imagem é o de uma função definida numa superfície bidimensional e tomando valores em um espaço de cor.

Definição 1 *Uma imagem é uma aplicação $f : U \subset \mathbb{R}^2 \rightarrow \mathcal{C}$, onde \mathcal{C} é um espaço de cor.*

¹ Apoio financeiro da FAPESP

² fsimeoni@icmc.sc.usp.br

³ messias@prudente.unesp.br

⁴ *Morphing (metamorfose)*: termo técnico que designa efeito de transição entre imagens.

A função f é chamada de imagem, o conjunto U de suporte da imagem e o conjunto $f(U) \subset \mathcal{C}$ de conjunto de cores da imagem. Em geral, tem-se $\mathcal{C} = \mathbb{R}^n$, mas os casos mais comuns são quando $n = 3$ e $n = 1$. Quando $n = 3$, obtém-se um espaço de cores tricromático com base de primárias R , G e B , resultando em uma imagem colorida. Quando $n = 1$, dizemos que a imagem é monocromática.

A representação mais comum de uma imagem consiste em tomar um subconjunto discreto $U' \subset U$ do domínio da função imagem. Neste caso, cada ponto $(x_i, y_j) \in U'$ é chamado de elemento da imagem ou *pixel*. O caso mais comum de discretização de uma imagem consiste em tomar o domínio como sendo o retângulo

$$U = [a, b] \times [c, d] = \{(x, y) \in \mathbb{R}^2 : a \leq x \leq b \text{ e } c \leq y \leq d\}$$

e discretizar esse retângulo usando os pontos de um reticulado bidimensional $\Delta = (\Delta_x, \Delta_y)$,

$$\Delta = \{(x_i, y_j) \in U : x_i = i \cdot \Delta_x, y_j = j \cdot \Delta_y, i, j \in \mathbb{Z}, \Delta_x, \Delta_y \in \mathbb{R}\}.$$

Assim, cada pixel (x_i, y_j) da imagem pode ser representado por coordenadas inteiras (i, j) . Portanto, a imagem pode ser representada de forma conveniente no formato matricial. Nessa representação, a imagem está associada a uma matriz A de dimensão $m \times n$ da seguinte forma:

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} = \begin{bmatrix} f(x_1, y_1) & f(x_1, y_2) & \cdots & f(x_1, y_n) \\ f(x_2, y_1) & f(x_2, y_2) & \cdots & f(x_2, y_n) \\ \vdots & \vdots & \ddots & \vdots \\ f(x_m, y_1) & f(x_m, y_2) & \cdots & f(x_m, y_n) \end{bmatrix}$$

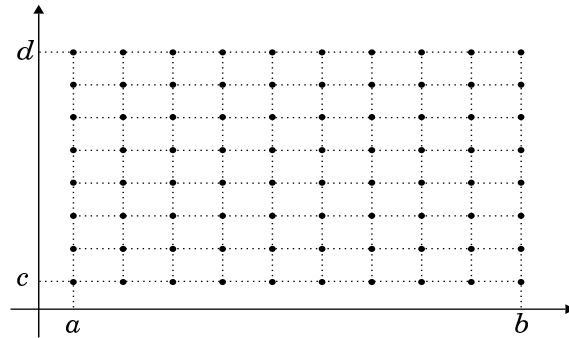


Figura 1: Reticulado uniforme da representação matricial da imagem.

3. Filtro de warping

Como visto anteriormente, a operação (ou filtro) que realiza deformações geométricas em uma imagem digital é comumente conhecido, na literatura de processamento de

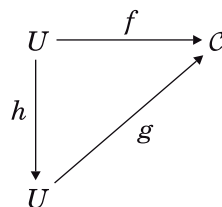
imagens, por *warping*. Seguindo o modelo matemático de imagem dado anteriormente, pode-se definir um filtro de *warping* como sendo simplesmente uma aplicação $h : U \subset \mathbb{R}^2 \rightarrow V \subset \mathbb{R}^2$.



Figura 2: Exemplo de *warping* em uma imagem.

Em geral, são feitas algumas restrições para a função h que define o *warping*. Do ponto de vista matemático, é natural exigir que essa função seja uma bijeção e, além disso, tanto h quanto sua inversa h^{-1} sejam contínuas. Uma função com essas características é chamada de homeomorfismo. Neste caso, a injetividade garante que não há superposição de pontos na deformação e a continuidade garante que não haverá rupturas da imagem. Os casos mais interessantes de *warping* ocorrem quando h é um difeomorfismo, ou seja, h e h^{-1} são de classe C^k .

A imagem filtrada é obtida por $g = f \circ h^{-1} : U \subset \mathbb{R}^2 \rightarrow \mathcal{C}$, ilustrado no diagrama abaixo.



Desse modo, a operação de *warping* faz uma mudança das coordenadas (x, y) dos pixels da imagem, para um outro sistema de coordenadas (u, v) , tal que

$$u = u(x, y), \quad v = v(x, y).$$

Um exemplo simples de *warping* é dado pela rotação da imagem por um ângulo de 90° em torno do pixel localizado na origem. Nesse caso, h é um difeomorfismo dado por

$$h(x, y) = \begin{bmatrix} \cos 90^\circ & -\text{sen } 90^\circ \\ \text{sen } 90^\circ & \cos 90^\circ \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = (-y, x).$$

4. Warping por triangulação

A maneira mais simples de se aplicar um filtro de *warping* é definir um sistema de coordenadas para a imagem através de uma malha de pontos, chamada de máscara.

Definição 2 *Seja $f : U \subset \mathbb{R}^2 \rightarrow \mathcal{C}$ uma imagem. Uma máscara para f é uma matriz M de dimensão $m \times n$ cujas entradas são pontos de U .*

Tendo em vista a máscara M , podemos obter uma triangulação da mesma tomando triângulos da forma $(M_{i,j}, M_{i,j+1}, M_{i+1,j+1})$ e $(M_{i,j}, M_{i+1,j}, M_{i+1,j+1})$, evitando assim uma triangulação de Delaunay, diminuindo a complexidade do problema.

A transformação de *warping* de uma imagem f será obtida a partir de uma transição entre duas máscaras distintas de f , mas com a mesma ordem. Assim, dados M e N duas máscaras distintas de f , A e B triângulos de M e N , respectivamente, sabemos que existe uma única transformação afim $T : U \rightarrow \mathbb{R}^2$ que transforma o triângulo A no triângulo B , ou seja, $T(A) = B$. É possível, então, construir uma função contínua h que associa a cada triângulo da máscara M , o triângulo correspondente (de mesmos índices) na máscara N , bastando definir a restrição da função h naquele triângulo como sendo a transformação afim associada. Matematicamente, a função de warping será dada por:

$$h : U \subset \mathbb{R}^2 \rightarrow \mathbb{R}^2, \text{ com } h|_{A_i} = T_i, \forall A_i \in M \quad (4.1)$$

onde T_i é a única transformação afim entre os triângulos correspondentes A_i e B_i tal que $T_i(A_i) = B_i$.

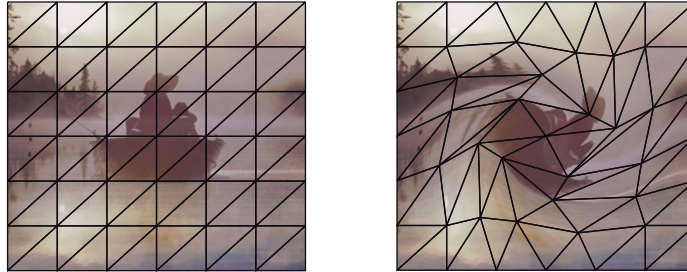
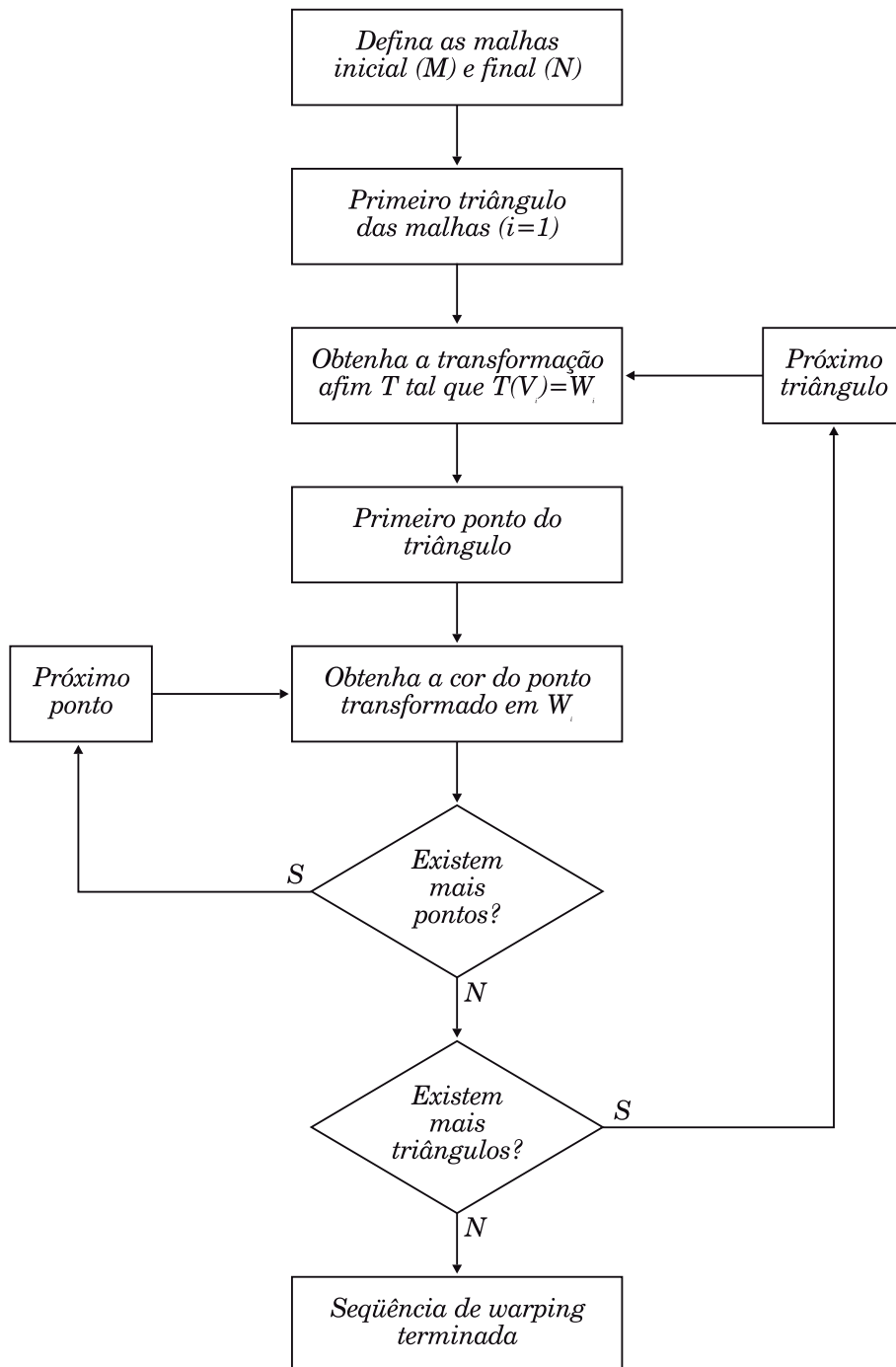


Figura 3: *Warping* de uma imagem com sua máscara.

4.1. Algoritmo para *warping* por triangulação

Com tudo o que foi visto até agora, pode-se definir um algoritmo que, dado uma imagem e suas máscaras inicial e final, encontre a transformação de *warping* h , aplicando-a na mesma. Tal algoritmo pode ser descrito através do fluxograma da Figura 4.

Figura 4: Fluxograma para o algoritmo de *warping* por triangulação

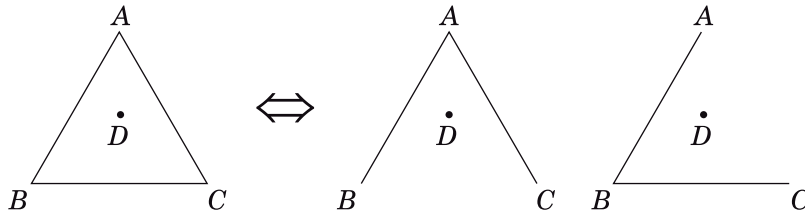
Porém, a implementação desta técnica de *warping* traz alguns problemas como, por exemplo, encontrar os pontos da imagem que pertencem a um triângulo da máscara, ou ainda, a determinação de cada transformação T que mapeia os triângulos correspondentes nas máscaras M e N . Como resultado do estudo desta técnica, apresenta-se, a seguir, possíveis soluções para estes problemas.

4.2. O problema de pontos interiores

O problema da determinação dos pontos interiores de um triângulo pode ser apresentado da seguinte forma: “Dado um triângulo com vértices A , B e C e um ponto D , determinar se D pertence ao interior do triângulo ABC ”.

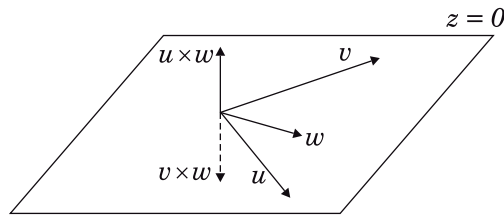
Para a solução deste problema, podemos enunciar um teorema cuja demonstração é trivial e não será feita aqui.

Teorema 1 *Sejam $A = (x_a, y_a)$, $B = (x_b, y_b)$ e $C = (x_c, y_c)$ os vértices de um triângulo e $D = (x_d, y_d)$ um ponto qualquer. O ponto D pertence ao interior do triângulo ABC se, e somente se, pertencer ao interior dos ângulos CAB e ABC .*



Tendo o resultado do Teorema 1, basta então obter um algoritmo que determine se o ponto dado pertence ao interior de um determinado ângulo. Para tanto, sejam $\vec{A} = (x_a, y_a, 0)$, $\vec{B} = (x_b, y_b, 0)$, $\vec{C} = (x_c, y_c, 0)$ e $\vec{D} = (x_d, y_d, 0)$. Definindo os vetores $u = \vec{B} - \vec{A}$, $v = \vec{C} - \vec{A}$ e $w = \vec{D} - \vec{A}$, o ponto D está no interior do ângulo CAB se, e somente se, as bases $\{u, v, u \times w\}$ e $\{u, v, v \times w\}$ tem orientações contrárias⁵. De fato, temos duas possíveis situações para o vetor w :

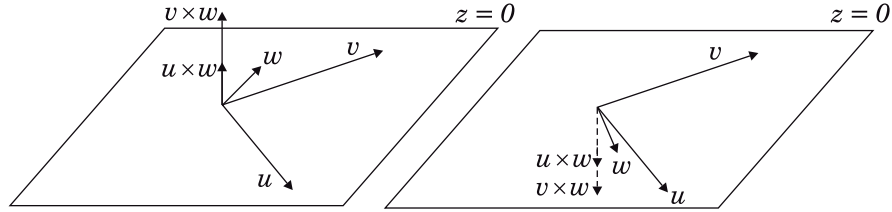
1º Caso: O vetor w pertence ao interior do ângulo formado pelos vetores u e v . Gráficamente, esta situação pode ser ilustrada pelo seguinte gráfico:



É fácil perceber que os vetores $u \times w$ e $v \times w$ terão mesma direção, mas sentidos opostos.

⁵Aqui, $u \times v$ denota o produto vetorial de u por v .

2º Caso: O vetor w não pertence ao interior do ângulo formado pelos vetores u e v . Graficamente, esta situação pode ser visualizada pelos seguintes gráficos:



Nesta situação, os vetores $u \times w$ e $v \times w$ terão não apenas a mesma direção, mas também o mesmo sentido. Agora, como

$$u \times w = \begin{vmatrix} \vec{i} & \vec{j} & \vec{k} \\ x_b - x_a & y_b - y_a & 0 \\ x_d - x_a & y_d - y_a & 0 \end{vmatrix} = (0, 0, u_w)$$

$$v \times w = \begin{vmatrix} \vec{i} & \vec{j} & \vec{k} \\ x_c - x_a & y_c - y_a & 0 \\ x_d - x_a & y_d - y_a & 0 \end{vmatrix} = (0, 0, v_w)$$

onde

$$u_w = (x_b - x_a)(y_d - y_a) - (y_b - y_a)(x_d - x_a)$$

$$v_w = (x_c - x_a)(y_d - y_a) - (y_c - y_a)(x_d - x_a)$$

segue que o ponto D está no interior do ângulo formado por u e v se, e somente se, os números u_w e v_w têm sinais opostos, ou ainda, se $u_w \cdot v_w \leq 0$.

Como foi visto na seção 2., uma imagem f pode ser representada por uma matriz de dimensão $m \times n$, e assim, as coordenadas dos pixels da imagem podem ser representados por números inteiros. Na verdade, quando se implementa uma ferramenta de processamento de imagens, cujo objetivo é mostrar o resultado obtido na tela do computador, usa-se coordenadas inteiras, pois os pixels do monitor são referenciados assim.

Desta forma, para verificar se o próximo ponto, digamos $(x_{d+1}, y_d) = (x_d + 1, y_d)$, observe que

$$\begin{aligned} \overline{u_w} &= (x_b - x_a)(y_d - y_a) - (y_b - y_a)(x_d + 1 - x_a) \\ &= (x_b - x_a)(y_d - y_a) - (y_b - y_a)(x_d - x_a) - (y_b - y_a) \\ &= u_w - (y_b - y_a). \end{aligned} \tag{4.2}$$

Analogamente,

$$\begin{aligned} \overline{v_w} &= (x_c - x_a)(y_d - y_a) - (y_c - y_a)(x_d + 1 - x_a) \\ &= (x_c - x_a)(y_d - y_a) - (y_c - y_a)(x_d - x_a) - (y_c - y_a) \\ &= v_w - (y_c - y_a). \end{aligned} \tag{4.3}$$

Assim, como os fatores $A_1 = (y_b - y_a)$ e $A_2 = (y_c - y_a)$ dependem apenas do triângulo dado e são constantes, o cálculo do produto vetorial é realizado apenas no início, bastando somar estas constantes para os demais pontos da linha. De modo análogo ao que foi feito em (4.2) e (4.3), para verificar se o ponto $(x_d, y_{d+1}) = (x_d, y_d + 1)$ pertence ao ângulo, basta notar que

$$\begin{aligned}\widetilde{u}_w &= u_w + (x_b - x_a), \\ \widetilde{v}_w &= v_w + (x_c - x_a).\end{aligned}$$

Desta maneira, uma vez obtidos os valores iniciais de u_w e v_w , basta acrescentar valores inteiros aos mesmos para obter os valores de u_w e v_w para o próximo ponto. Isto torna o algoritmo extremamente rápido e eficiente, melhorando sobremaneira a eficiência do processo de *warping*.

4.3. Determinação das transformações

Como visto em (4.1), o filtro de warping por triangulação pode ser dado por uma função h tal que a sua restrição a um triângulo da malha M mapeia-o em seu triângulo correspondente na malha N , através de uma transformação afim.

O problema proposto aqui é o seguinte: “Dados dois triângulos $V, W \subset \mathbb{R}^2$, encontrar a única transformação afim $T : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ tal que $T(V) = W$.”

De uma forma mais específica, sendo p_1, p_2 e p_3 vértices de V e q_1, q_2 e q_3 vértices de W , tais que

$$\begin{aligned}p_1 &= (v_1, w_1), & q_1 &= (x_1, y_1), \\ p_2 &= (v_2, w_2), & q_2 &= (x_2, y_2), \\ p_3 &= (v_3, w_3), & q_3 &= (x_3, y_3),\end{aligned}\tag{4.4}$$

encontrar a transformação T tal que $T(V) = W$ é equivalente a encontrar uma transformação T tal que

$$\begin{aligned}T(p_1) &= q_1, \\ T(p_2) &= q_2, \\ T(p_3) &= q_3.\end{aligned}\tag{4.5}$$

Antes de determinar quem é T , é necessário ter em mãos alguns resultados que são enunciados a seguir:

Teorema 2 *Sejam p_1, p_2 e p_3 pontos não colineares de \mathbb{R}^2 . Então, cada ponto p do plano pode ser escrito de modo único na forma*

$$p = \lambda_1 p_1 + \lambda_2 p_2 + \lambda_3 p_3,$$

onde λ_1, λ_2 e λ_3 são números reais satisfazendo $\lambda_1 + \lambda_2 + \lambda_3 = 1$. Os coeficientes λ_1, λ_2 e λ_3 são denominados coordenadas baricêntricas de p em relação a p_1, p_2 e p_3 .

Demonstração: Basta observar que, dados p , p_1 , p_2 e p_3 , sendo $p = (x, y)$ e $p_i = (x_i, y_i)$, $i = 1, 2, 3$, os termos λ_1 , λ_2 e λ_3 , tais que $\sum_{i=1}^3 \lambda_i = 1$, são soluções do seguinte sistema de três equações a três incógnitas:

$$\begin{cases} \lambda_1 x_1 + \lambda_2 x_2 + \lambda_3 x_3 = x \\ \lambda_1 y_1 + \lambda_2 y_2 + \lambda_3 y_3 = y \\ \lambda_1 + \lambda_2 + \lambda_3 = 1 \end{cases} \quad (4.6)$$

onde o determinante do sistema é dado por

$$D = \begin{vmatrix} x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \\ 1 & 1 & 1 \end{vmatrix} = \begin{vmatrix} x_1 & x_2 & x_3 & x_1 \\ y_1 & y_2 & y_3 & y_1 \end{vmatrix} \quad (4.7)$$

que representa o dobro da área do triângulo de vértices p_1 , p_2 e p_3 , já que estes não são colineares. Portanto, $D \neq 0$ e o sistema possui solução única para cada p , o que demonstra o teorema. ■

Generalizando este teorema, obtém-se o seguinte resultado:

Corolário 1 *Sejam u_0, u_1, \dots, u_n , $n + 1$ pontos do espaço \mathbb{R}^n , de modo que o conjunto $\{u_1 - u_0, u_2 - u_0, \dots, u_n - u_0\}$ forma uma base deste espaço. Então, qualquer ponto $u \in \mathbb{R}^n$ pode ser escrito como uma combinação de suas coordenadas bari-cêntricas da seguinte forma*

$$u = \sum_{i=0}^n \lambda_i u_i, \text{ com } \lambda_i \in \mathbb{R}, \sum_{i=0}^n \lambda_i = 1.$$

Definição 3 *Uma transformação $T : \mathbb{R}^n \rightarrow \mathbb{R}^n$ é dita afim se*

$$T((1-t)u + tv) = (1-t)T(u) + tT(v), \forall u, v \in \mathbb{R}^n, \forall t \in \mathbb{R}.$$

Teorema 3 *Sejam $u = \sum_{i=0}^n \lambda_i u_i$ um vetor do \mathbb{R}^n escrito como uma combinação baricêntrica dos vetores $u_0, u_1, u_2, \dots, u_n$, e $T : \mathbb{R}^n \rightarrow \mathbb{R}^n$ uma transformação afim. Então*

$$T(u) = T\left(\sum_{i=0}^n \lambda_i u_i\right) = \sum_{i=0}^n \lambda_i T(u_i).$$

Demonstração: Será feita por indução finita. Para o caso $n = 1$, temos $\lambda_0 + \lambda_1 = 1 \Rightarrow \lambda_1 = 1 - \lambda_0$. Daí, pela definição de transformação afim, vem

$$\begin{aligned} T(\lambda_0 u + \lambda_1 v) &= T(\lambda_0 u + (1 - \lambda_0) v) \\ &= \lambda_0 T(u) + (1 - \lambda_0) T(v) \\ &= \lambda_0 T(u) + \lambda_1 T(v). \end{aligned} \quad (4.8)$$

Agora, suponha que o teorema seja válido para $n - 1$ pontos, isto é,

$$T\left(\sum_{i=0}^{n-1} \lambda_i u_i\right) = \sum_{i=0}^{n-1} \lambda_i T(u_i). \quad (4.9)$$

Sejam $\lambda_0, \lambda_1, \dots, \lambda_n \in \mathbb{R}$ tais que $\sum_{i=0}^n \lambda_i = 1$ e $u = \sum_{i=0}^n \lambda_i u_i \in \mathbb{R}^n$. Assim,

$$\lambda_0 + \lambda_1 + \dots + \lambda_n = 1 \Rightarrow \lambda_0 + \lambda_1 + \dots + \lambda_{n-1} = 1 - \lambda_n.$$

Se $\lambda_n = 0$ então o teorema está provado, pois recai no caso $n - 1$. É claro que $\lambda_n \neq 1$, pois se fosse, teríamos $\lambda_1 = \lambda_2 = \dots = \lambda_{n-1} = 0$ e, neste caso, eles não formariam uma combinação baricêntrica. Caso contrário,

$$\lambda_0 + \lambda_1 + \dots + \lambda_{n-1} = 1 - \lambda_n \Rightarrow \frac{\lambda_0 + \lambda_1 + \dots + \lambda_{n-1}}{1 - \lambda_n} = \sum_{i=0}^{n-1} \frac{\lambda_i}{1 - \lambda_n} = 1. \quad (4.10)$$

Daí,

$$\begin{aligned} T\left(\sum_{i=0}^n \lambda_i u_i\right) &= T\left((1 - \lambda_n) \left(\frac{\lambda_0 u_0 + \lambda_1 u_1 + \dots + \lambda_{n-1} u_{n-1}}{1 - \lambda_n}\right) + \lambda_n u_n\right) \\ &\stackrel{(4.8)}{=} (1 - \lambda_n) T\left(\frac{\lambda_0 u_0 + \lambda_1 u_1 + \dots + \lambda_{n-1} u_{n-1}}{1 - \lambda_n}\right) + \lambda_n T(u_n) \\ &\stackrel{(4.10)}{=} (1 - \lambda_n) T\left(\sum_{i=0}^{n-1} \frac{\lambda_i}{1 - \lambda_n} u_i\right) + \lambda_n T(u_n) \\ &\stackrel{(4.9)}{=} (1 - \lambda_n) \sum_{i=0}^{n-1} \frac{\lambda_i}{1 - \lambda_n} T(u_i) + \lambda_n T(u_n) \\ &= \lambda_0 T(u_0) + \lambda_1 T(u_1) + \dots + \lambda_{n-1} T(u_{n-1}) + \lambda_n T(u_n) \\ &= \sum_{i=0}^n \lambda_i T(u_i), \end{aligned}$$

como queríamos demonstrar. Logo, o teorema é válido $\forall n \in \mathbb{N}$. ■

Agora, seja (v, w) um vetor do plano. Podemos escrevê-lo como combinação baricêntrica dos vetores p_1, p_2 e p_3 dados em (4.4). Mais precisamente,

$$(v, w) = \lambda_1(v_1, w_1) + \lambda_2(v_2, w_2) + \lambda_3(v_3, w_3), \text{ com } \sum_{i=1}^3 \lambda_i = 1, \quad (4.11)$$

de onde se obtém um sistema análogo ao dado em (4.6), com representação matricial

$$\begin{bmatrix} v_1 & v_2 & v_3 \\ w_1 & w_2 & w_3 \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \end{bmatrix} = \begin{bmatrix} v \\ w \\ 1 \end{bmatrix}, \quad (4.12)$$

cuja solução é dada por

$$\begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \end{bmatrix} = \frac{1}{k} \cdot \begin{bmatrix} (v - v_2)(w - w_3) - (v - v_3)(w - w_2) \\ (v - v_3)(w - w_1) - (v - v_1)(w - w_3) \\ (v - v_1)(w - w_2) - (v - v_2)(w - w_1) \end{bmatrix}, \quad (4.13)$$

onde $k = (v_1 - v_2)(w_1 - w_3) - (v_1 - v_3)(w_1 - w_2)$.

Assim, usando o fato de T ser uma transformação afim, pode-se aplicar os resultados obtidos em (4.5) e no Teorema 3, donde

$$\begin{aligned} T(v, w) &= T(\lambda_1(v_1, w_1) + \lambda_2(v_2, w_2) + \lambda_3(v_3, w_3)) \\ &= \lambda_1 T(v_1, w_1) + \lambda_2 T(v_2, w_2) + \lambda_3 T(v_3, w_3) \\ &= \lambda_1(x_1, y_1) + \lambda_2(x_2, y_2) + \lambda_3(x_3, y_3) \\ &= (\lambda_1 x_1 + \lambda_2 x_2 + \lambda_3 x_3, \lambda_1 y_1 + \lambda_2 y_2 + \lambda_3 y_3), \end{aligned}$$

onde os valores λ_1 , λ_2 e λ_3 são dados por (4.13).

Finalmente, para se obter uma aplicação afim que transforma o triângulo V no triângulo W , primeiro calcula-se as coordenadas baricêntricas de um ponto (v, w) em relação aos vértices de V e, depois, para encontrar $T(v, w)$, basta escrever os vértices de W como uma combinação das mesmas coordenadas baricêntricas encontradas.

Pode-se observar ainda que, para obter uma forma genérica para T , basta fazer

$$\begin{aligned} \lambda_1(v, w) &= \frac{1}{k} ((v - v_2)(w - w_3) - (v - v_3)(w - w_2)), \\ \lambda_2(v, w) &= \frac{1}{k} ((v - v_3)(w - w_1) - (v - v_1)(w - w_3)), \\ \lambda_3(v, w) &= \frac{1}{k} ((v - v_1)(w - w_2) - (v - v_2)(w - w_1)), \end{aligned}$$

e a transformação afim $T : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ é dada por

$$T(v, w) = \lambda_1(v, w)(x_1, y_1) + \lambda_2(v, w)(x_2, y_2) + \lambda_3(v, w)(x_3, y_3). \quad (4.14)$$

4.4. Interpolação dos pontos transformados

Um dos problemas gerados pela deformação de imagens é a criação de “buracos”, devido ao aumento da área da figura transformada. Este buracos nada mais são do que pontos na imagem final g a qual não foram atribuídos valores pela transformação de *warping* h .

A Figura 5 mostra com clareza o efeito causado por este problema. Para resolvê-lo, existem várias técnicas de interpolação que podem ser aplicadas, porém, as mais utilizadas são replicação de pixels (também chamado de filtro *box*) e interpolação bilinear, devido ótima relação entre custo computacional e qualidade dos resultados.

4.4.1. Técnica de replicação de pixels

Esta técnica consiste em, durante o processo de aplicação do *warping*, fazer com que os pixels não definidos na imagem resultante sejam preenchidos com o valor

do pixel definido mais próximo. Usualmente, este processo se resume em apenas repetir na linha e na coluna seguinte da imagem resultante, o valor do pixel atual. Por exemplo, considere um fragmento da imagem original f dado por

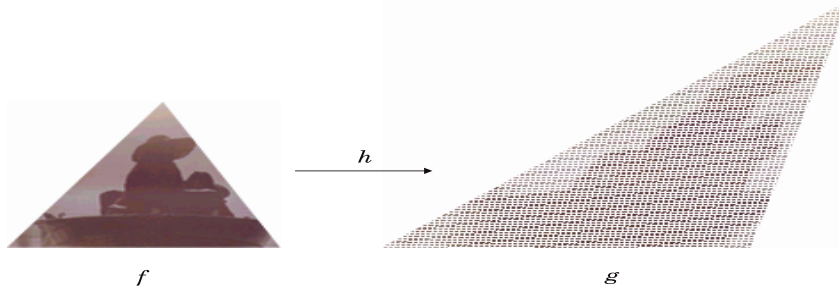


Figura 5: Transformação de *warping* h aplicado a uma imagem f e a imagem resultante g .

$$\begin{array}{ccccc} & & \vdots & & \vdots \\ \cdots & f(i, j) & & f(i, j + 1) & \cdots \\ \cdots & f(i + 1, j) & & f(i + 1, j + 1) & \cdots \\ & & \vdots & & \vdots \end{array}$$

Após a aplicação da transformação h , obtém-se, por exemplo, a imagem g dada por

$$\begin{array}{ccccc} & & \vdots & & \vdots \\ \cdots & g(i, j) & 0 & g(i, j + 1) & \cdots \\ & 0 & 0 & 0 & \\ \cdots & g(i + 1, j) & 0 & g(i + 1, j + 1) & \cdots \\ & & \vdots & & \vdots \end{array}$$

Daí, aplicando-se o filtro descrito, vem

$$\begin{array}{ccccc} & & \vdots & & \vdots \\ \cdots & g(i, j) & g(i, j) & g(i, j + 1) & \cdots \\ & g(i, j) & g(i, j) & g(i, j + 1) & \\ \cdots & g(i + 1, j) & g(i + 1, j) & g(i + 1, j + 1) & \cdots \\ & & \vdots & & \vdots \end{array}$$

Esse filtro, apesar de ser o mais rápido, é o que possui piores resultados. Como aumentam-se os detalhes da imagem, os defeitos inerentes desta reconstrução ficam mais visíveis, deixando na imagem um efeito “quadriculado”. No entanto, esse recurso é bem popular, devido à sua facilidade de implementação e eficiência computacional.

4.4.2. Interpolação bilinear

O método da interpolação bilinear consiste em obter a cor de um pixel não definido através do cálculo de médias, aplicado entre as linhas e colunas da imagem. Para simplificar, considere que os pixel da imagem g são dados por $g(i, j)$. Assim, no mesmo exemplo acima, sendo a imagem após a transformação de *warping* dada por

$$\begin{array}{ccccccc}
 & & & \vdots & & & \\
 & & & g(i, j) & 0 & g(i, j + 1) & \\
 \dots & & & 0 & 0 & 0 & \dots \\
 & & & g(i + 1, j) & 0 & g(i + 1, j + 1) & \\
 & & & \vdots & & &
 \end{array}$$

aplicando-se a interpolação bilinear, obtemos

$$\begin{array}{ccccccc}
 & & & \vdots & & & \\
 \dots & & \frac{g(i, j)}{\frac{g(i, j) + g(i + 1, j)}{2}} & \frac{\frac{g(i, j) + g(i, j + 1)}{2}}{\frac{g(i, j) + g(i, j + 1) + g(i + 1, j) + g(i + 1, j + 1)}{4}} & \frac{g(i, j + 1)}{\frac{g(i, j + 1) + g(i + 1, j + 1)}{2}} & \dots & \\
 & & g(i + 1, j) & \frac{g(i + 1, j) + g(i + 1, j + 1)}{2} & g(i + 1, j + 1) & & \\
 & & & \vdots & & &
 \end{array}$$

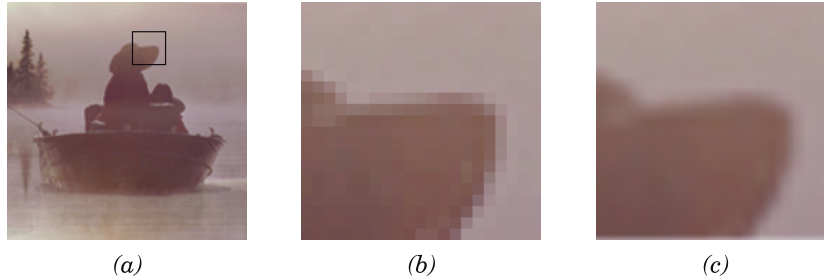


Figura 6: Transformação de *zoom* da imagem (a) por (b) replicação de pixels e (c) interpolação bilinear.

A interpolação bilinear apresenta bons resultados e, ainda, é bastante simples na sua implementação, o que o torna um dos métodos mais utilizados na prática. A Figura 6 traz um exemplo onde a aplicação de *warping* é simplesmente um *zoom* com escala 2, onde pode-se constatar na prática a diferença entre estes dois métodos.

5. Aplicação de *warping* à visualização de superfícies topográficas

Como aplicação de todo este estudo sobre técnicas de *warping*, foi implementada uma ferramenta no módulo de visualização de superfícies topográficas MicroDTM já desenvolvido (ver em [7] e [8]), que é capaz de distorcer uma dada imagem em cima de um modelo digital de terreno criado neste programa.

Aproveitando o fato de que o MicroDTM trabalha com uma estrutura de dados organizada na forma de malha triangular, este processo torna-se muito simples. Em primeiro lugar, tem-se uma malha regular tridimensional como modelo digital do terreno, onde cada vértice tem coordenadas (x, y, z) . Após a aplicação das rotações e da projeção escolhida, obtém-se uma malha bidimensional, com o mesmo número de triângulos, só que totalmente transformada, o que dá o efeito tridimensional ao terreno.

De posse do número de triângulos da malha e da imagem digital, constrói-se uma malha regular bidimensional em cima nas dimensões da imagem e, cada triângulo desta malha corresponde a somente um triângulo da malha transformada.

Desta forma, basta aplicar a técnica de *warping* por triangulação em cima destas duas malhas, obtendo a imagem distorcida em cima do modelo do terreno, como era esperado. A plotagem da imagem é feita através de uma adaptação no algoritmo do pintor, que determina a ordem de apresentação dos triângulos e em seguida os preenche com a parte da imagem que lhe corresponde. Todo esse processo pode ser visualizado nas figuras seguintes.

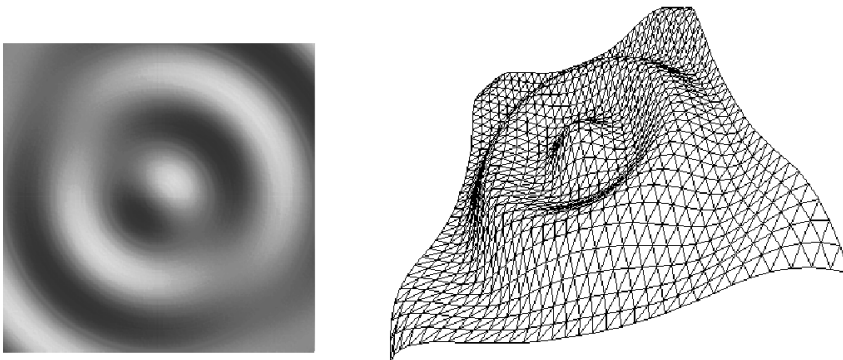


Figura 7: Textura a ser aplicada no gráfico da função $\cos \sqrt{x^2 + y^2}$, obtida no MicroDTM.

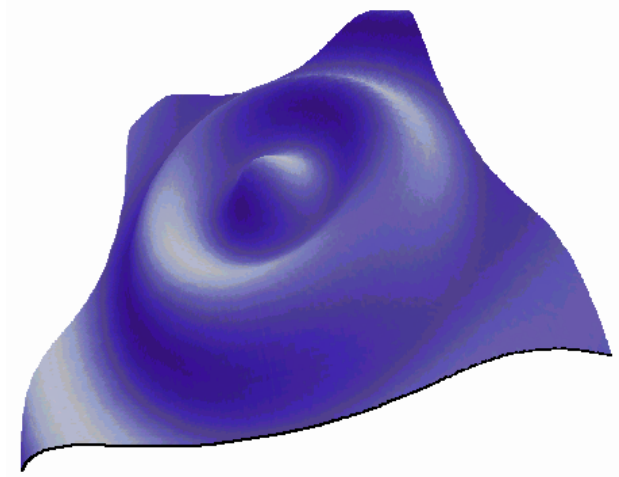


Figura 8: Resultado da distorção, sem a grade da malha triangular.

Referências

- [1] A. O. Artero e M. Meneguette, Transformações Espaciais e Técnicas de Interpolação Aplicadas em Imagens, *Colloquium*, 1 (3) (1998), 43-51.
- [2] E. Doescher e M. Meneguette, Deformações Geométricas em Imagens: metamorfose, registro e texturização, Monografia - FCT/UNESP, 1996.
- [3] L. H. Figueiredo e P. C. P. Carvalho, Introdução à Geometria Computacional, 18º Colóquio Brasileiro de Matemática, IMPA, 1991.
- [4] J. Gomes e L. Velho, “Computação Gráfica: Imagem”, IMPA, 1994.
- [5] J. Gomes e L. Velho, “Computação Gráfica”, Volume 1, IMPA, 1998.
- [6] E. L. Lima, “Curso de Análise”, Volume 2, IMPA, 1981.
- [7] F. S. Sousa e M. Meneguette Jr., Apresentação computacional de objetos 3D em vistas paralela e perspectiva, com ênfase a superfícies topográficas e isolinhas, XXI CNMAC, setembro, 1998.
- [8] F. S. Sousa e M. Meneguette Jr., MicroDTM: Um módulo para a visualização de superfícies topográficas, I Fórum de Ciências da FCT, novembro, 1998.

