

Transformaciones Geométricas

Rodrigo Javier Herrera García *

Las *transformaciones geométricas* son necesarias en el procesamiento de imágenes debido a las distorsiones que se presentan en el momento de la captura. Dos imágenes tomadas al mismo objeto en diferentes momentos no son exactamente iguales. Las transformaciones geométricas permiten intentar igualarlas. Por ejemplo, las imágenes satelitales de una misma zona de la tierra varían por la posición del satélite puesto que este no tiene la misma trayectoria y la captura de la imagen no se hace desde la misma posición. Para compararlas es necesario modificar alguna. Por otro lado, también se presenta distorsión espacial por la curvatura de la tierra.

Una transformación geométrica se compone de dos pasos: la *transformación de coordenadas* de los píxeles y la modificación del brillo de cada uno de los píxeles conocida como *interpolación del brillo*. El primer paso ubica los píxeles en una nueva posición, la que le corresponde en la imagen transformada. Sin embargo, la nueva posición puede no coincidir con la rejilla determinada para una imagen digital. Entonces, se hace necesario modificar el brillo del píxel para ubicarlo en la nueva posición. Los métodos de obtención del nuevo brillo pueden hacerse teniendo en cuenta sus vecinos dada su alta correlación. Luego los métodos más comunes son los de interpolación. En seguida, algunos de los métodos más comunes de cada uno de los pasos.

1. Transformación de las Coordenadas

Una transformación de coordenadas de los píxeles consiste de un vector de funciones de transformación \mathbf{T} que relaciona la posición (x, y) de un píxel con una nueva posición (x', y') . La **Figura 1** ilustra de manera general el proceso de transformación. El vector \mathbf{T} está definido por dos ecuaciones funcionales que corresponden al cambio de posición horizontal y vertical de los píxeles. Entonces,

$$x' = f_x(x, y) \quad y' = f_y(x, y) \quad (1)$$

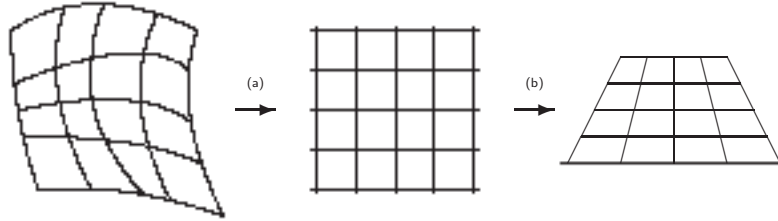


Figura 1: Transformación geométrica de una imagen. Una imagen distorsionada espacialmente, se transforma en una imagen regular (a) o al contrario (b).

Las funciones f_x y f_y son conocidas de antemano o pueden derivarse a partir de las imágenes original y transformada. Para encontrar las ecuaciones se recurre a la correspondencia de algunos píxeles dentro de las dos imágenes. En general, las ecuaciones se aproximan a polinomios de la siguiente manera

$$x' = \sum_{i=0}^m \sum_{j=0}^m a_{ij} x^i y^j \quad y \quad y' = \sum_{i=0}^m \sum_{j=0}^m b_{ij} x^i y^j \quad (2)$$

*Profesor del Proyecto Curricular de Ingeniería Electrónica, Facultad de Ingeniería, Universidad Distrital Francisco José de Caldas, Bogotá, D.C., Colombia.

donde m es el grado de los polinomios. Teniendo en cuenta que esta transformación es lineal con respecto a los coeficientes, entonces, si se conoce la posición (x, y) y (x', y') de suficientes pares de puntos en las dos imágenes, es posible determinar los coeficientes a_{ij} y b_{ij} por medio de la solución de un conjunto de ecuaciones lineales. Sin embargo, entre mayor sea el grado de los polinomios, mayor es el número de pares de posiciones de píxeles que se deben conocer.

Si la transformación de las coordenadas no cambia rápidamente con respecto a la posición, que es lo que sucede en la mayoría de los casos, se utilizan polinomios de bajo grado. La cantidad de puntos que se deben conocer son relativamente pocos. Por ejemplo, para $m = 2$ se necesitan 6 pares de puntos mínimo; para $m = 3$ se requieren 10 pares. Lo importante es que los pares de puntos estén distribuidos en la imagen de manera que puedan expresar la transformación de coordenadas. En general, se debe buscar que estén distribuidos uniformemente en la imagen. De hecho, la transformación de coordenadas se vuelve más sensible a la distribución cuando el grado de los polinomios es mayor.

Una transformación que se utiliza en la práctica es la *transformación bilineal*. Para esta transformación se requiere conocer 4 pares de puntos. La transformación está definida como

$$\begin{aligned} x' &= a_0 + a_1x + a_2y + a_3xy \\ y' &= b_0 + b_1x + b_2y + b_3xy \end{aligned} \quad (3)$$

donde $a_0 = a_{00}$, $a_1 = a_{10}$, $a_2 = a_{01}$, $a_3 = a_{11}$, $b_0 = b_{00}$, $b_1 = b_{10}$, $b_2 = b_{01}$ y $b_3 = b_{11}$, cuando $m = 1$, de acuerdo con la ecuación (2). Una transformación más sencilla es la *transformación afín* donde es necesario conocer sólo tres pares de puntos. La transformación está definida como

$$\begin{aligned} x' &= a_0 + a_1x + a_2y \\ y' &= b_0 + b_1x + b_2y \end{aligned} \quad (4)$$

Dentro de esta transformación se encuentran las transformaciones de coordenadas de píxeles más comunes como son *traslación*, *cambio de escala*, *rotación* y *deslizamiento*. Las definiciones para cada una de ellas son:

traslación: la imagen se traslada de una posición en el plano a otro. Aquí a corresponde al desplazamiento en x y b al desplazamiento en y ,

$$\begin{aligned} x' &= x + a \\ y' &= y + b \end{aligned} \quad (5)$$

cambio de escala: la imagen se hace más grande o más pequeña dependiendo de los factores de cambio. En esta transformación, a y b corresponden a los factores de cambio en los ejes x y y , respectivamente. Las ecuaciones son,

$$\begin{aligned} x' &= ax \\ y' &= by \end{aligned} \quad (6)$$

rotación: la imagen gira un ángulo con respecto al centro de coordenadas. En la rotación, ϕ es el ángulo de rotación a partir del origen,

$$\begin{aligned} x' &= x \cos \phi + y \sin \phi \\ y' &= -x \sin \phi + y \cos \phi \end{aligned} \quad (7)$$

deslizamiento: desplazamiento variable en el eje horizontal que depende linealmente de la posición en el eje vertical. En esta transformación, φ es el ángulo que se forma en la imagen por el deslizamiento en el eje x ,

$$\begin{aligned} x' &= x + y \tan \varphi \\ y' &= y \end{aligned} \quad (8)$$

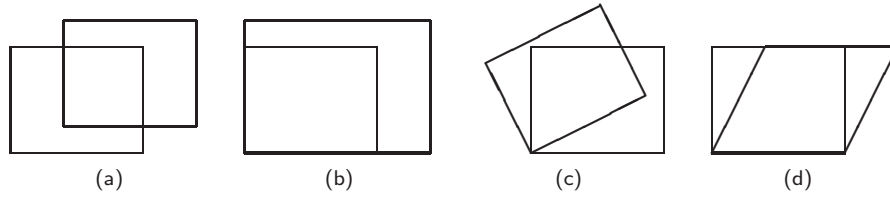


Figura 2: Representación gráfica de las transformaciones de coordenadas (transformaciones afines): (a) traslación, (b) cambio de escala, (c) rotación y (d) deslizamiento.

El deslizamiento es una distorsión común en la captura de imágenes, cuando el detector o el objeto están en movimiento y el registro se hace por filas o columnas, como en las cámaras de televisión. La **Figura 2** muestra gráficamente el proceso de transformación de la traslación, cambio de escala, rotación y deslizamiento.

El *jacobiano* suministra información sobre los cambios en el sistema de coordenadas cuando se aplica una transformación de coordenadas sobre una imagen completa. El jacobiano se define como

$$J = \begin{vmatrix} \frac{\partial x'}{\partial x} & \frac{\partial x'}{\partial y} \\ \frac{\partial y'}{\partial x} & \frac{\partial y'}{\partial y} \end{vmatrix} \quad (9)$$

Si una transformación no tiene inversa, se dice que es *singular* y $J = 0$. Si el área de una imagen no varía con la transformación entonces $J = 1$. Por ejemplo, el jacobiano para la traslación es $J = 1$, para el cambio de escala es $J = ab$, para la rotación es $J = 1$ y para el deslizamiento es $J = 1$. La única transformación que no permitiría hacer mediciones de área directamente sería la de cambio de escala.

La **Figura 3** muestra diversas transformaciones de las coordenadas de los píxeles para una imagen. Se busca, en este caso, que la figura plasmada sobre la superficie pueda ser observada de “frente” y no en perspectiva como fue capturada. En primera instancia se realizó una rotación, obteniendo el ángulo a partir de puntos de referencia. El resultado se muestra en la parte (b) de la Figura. Obsérvese que la base del objeto está en paralelo con el eje x . Posteriormente, se hizo un cambio de escala en el eje horizontal (parte (c) de la Figura). Debido a que el objeto tiene una forma aparentemente rectangular (obviamente, la parte derecha del objeto no constituye un lado de un rectángulo), se realizó una transformación de deslizamiento vertical para obtener la forma real del objeto. El lado superior del objeto también debería ser paralelo con el eje x . El resultado se muestra en la parte (d). Los parámetros de este proceso se seleccionaron con la ayuda del objeto real, de donde se obtuvieron las proporciones.

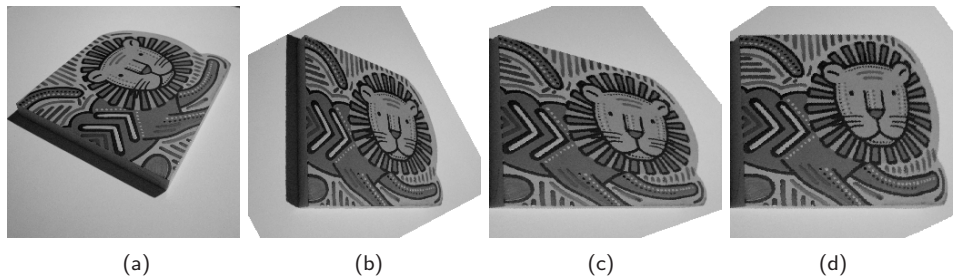


Figura 3: Transformaciones de coordenadas de píxeles sobre una imagen real. (a) Imagen original, (b) resultado de la rotación de la imagen, (c) resultado del cambio de escala en x , (d) resultado del deslizamiento vertical. En todos los casos, las dimensiones de las imágenes cambiaron. Sin embargo, se recortaron al mismo tamaño de la original.

Transformaciones más complejas deben estudiarse particularmente para cada caso. Transformaciones de coordenadas de imágenes sobre superficies curvas pueden implicar un conocimiento previo de las dimensiones de estas curvas (como por ejemplo las imágenes de zonas sobre la superficie de la tierra o de otros planetas). Adicionalmente, los detectores también pueden incluir distorsiones geométricas en las imágenes capturadas que deben tenerse en cuenta para su corrección.

Un ejemplo de transformaciones de coordenadas que dependen de la geometría de la imagen se puede ver en la **Figura 4** donde se busca una representación visual rectangular del iris del ojo que originalmente se tiene en forma radial. Observe que la transformación mapea el centro de la pupila en el eje de las abscisas de la imagen transformada y cada uno de los sectores anulares alrededor del centro de la pupila se mapea en franjas horizontales.

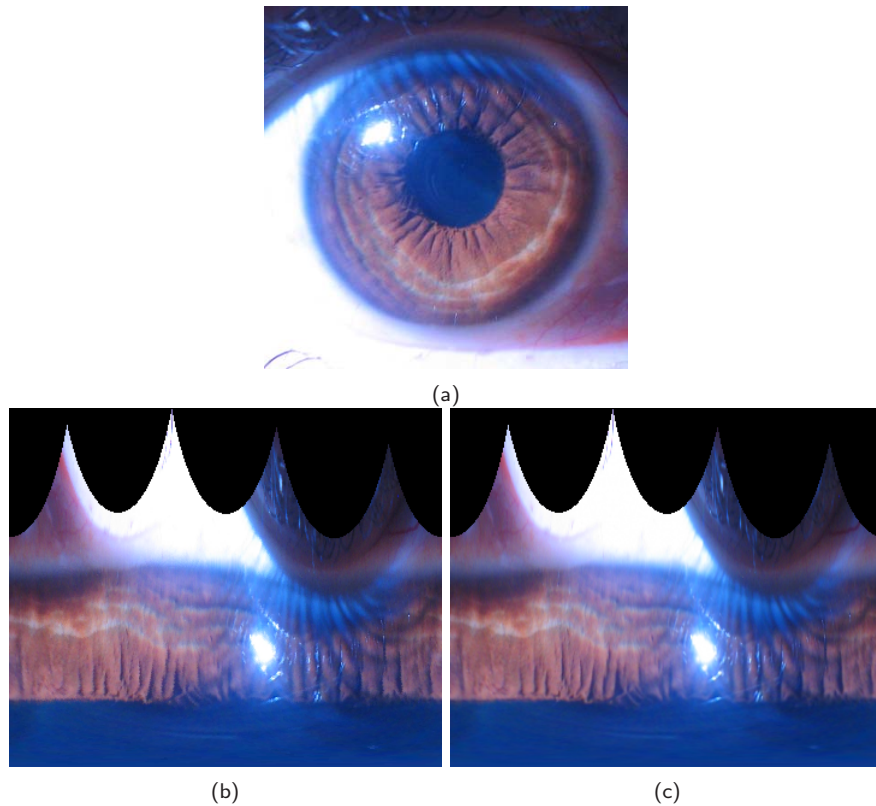


Figura 4: Transformaciones de coordenadas de píxeles sobre una imagen real con geometría propia. (a) Imagen original de un ojo con geometría radial centrada en la pupila; (b) resultado de la transformación de la imagen con interpolación «vecino más cercano», (c) resultado de la misma transformación con interpolación «bilineal». Las imágenes transformadas muestran el ángulo en las abscisas y el radio en las ordenadas. (Véase la sección 2 para la interpolación.)

2. Interpolación del brillo

En una transformación de coordenadas, aplicando la ecuación descrita en (1), se obtienen las nuevas coordenadas (x', y') . La posición de los puntos de la imagen transformada no necesariamente se ajusta con la retícula discreta de la imagen de salida haciendo que los píxeles tengan coordenadas no enteras. Sin embargo es necesario tener una rejilla de coordenadas enteras. Entonces, los valores de brillo se pueden ajustar a las posiciones de la rejilla discreta por medio de la interpolación del brillo de algunos píxeles del vecindario de coordenadas no enteras. Sin embargo, entre más sencilla sea la interpolación, mayor es la pérdida en la precisión fotométrica y hasta en la precisión geométrica misma. No obstante, entre más sencilla sea la interpolación, menor será la carga computacional añadida a la transformación geométrica.

Los tres métodos de interpolación más comunes en las transformaciones geométricas son el *vecino más cercano* o *interpolación de orden cero*, la *bilineal* y la *bicúbica*. El proceso de interpolación consiste en encontrar el nivel de gris de cada uno de los píxeles de la imagen transformada considerando los píxeles superpuestos de la transformación de coordenadas. Esto implica que se debe crear una relación entre los píxeles de la retícula digital de la imagen transformada y los píxeles de la imagen original.

2.1. Interpolación de orden cero

Así como se tuvo un operador de transformación de coordenadas a través de la ecuación (1), también se puede tener un operador inverso de manera que

$$x = f_x^{-1}(x', y') \quad y = f_y^{-1}(x', y') \quad (10)$$

donde f_x^{-1} y f_y^{-1} corresponden a las funciones inversas de la transformación de coordenadas.

La transformación de coordenadas puede producir valores de x' y y' que no son números enteros. Esto hace que las coordenadas no coincidan con la retícula digital de la imagen transformada. Se debe entonces aplicar algún algoritmo de aproximación al número entero más cercano. Usualmente, el algoritmo preferido es el de redondeo (en inglés *round*). Luego, las coordenadas para el píxel serían x'_q y y'_q obtenidas como

$$(x'_q, y'_q) = (\text{round}(x'), \text{round}(y')) \quad (11)$$

En la *interpolación de orden cero*, la asignación del nivel de gris para un píxel en la retícula digital de la imagen transformada corresponde al nivel de gris de aquel píxel de la imagen original que esté espacialmente más cercano a las coordenadas (x_r, y_r) obtenidas a través de la transformación geométrica inversa de la ecuación (10) aplicada sobre las coordenadas de la retícula digital (x'_q, y'_q) . La **Figura 5** muestra un ejemplo. Se puede observar que después de obtener las coordenadas de un píxel, a través de la transformación inversa, éstas pueden no ser enteras o no coincidir con la retícula digital original. Se debe buscar entonces el nivel de gris que corresponda apropiadamente con la imagen original. Como se observa, el vecino más cercano corresponde al píxel que está a menor distancia espacial¹. Luego, las coordenadas que corresponden a la retícula digital de la imagen original se obtienen como

$$\begin{aligned} (x, y) &= (\text{round}(f_x^{-1}(x'_q, y'_q)), \text{round}(f_y^{-1}(x'_q, y'_q))) \\ &= (\text{round}(x_r), \text{round}(y_r)) \end{aligned} \quad (12)$$

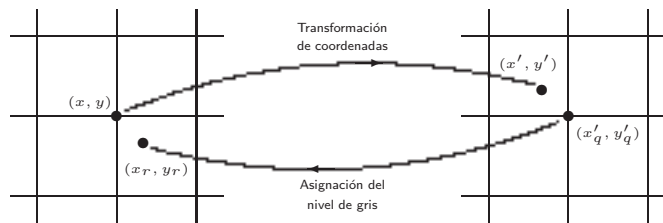


Figura 5: Interpolación de orden cero o vecino más cercano para definir el nivel de gris de un píxel de una imagen transformada geométricamente. (Observe que las coordenadas de los píxeles tanto de la imagen original como de la transformada pueden no coincidir.)

El nivel de gris que le corresponde al píxel (x'_q, y'_q) es el del píxel (x, y) de donde proviene (x', y') . Así se obtiene el nivel de gris para todos los píxeles de la retícula digital de la imagen transformada. Las imágenes de la **Figura 3** fueron obtenidas a través de este método.

2.2. Interpolación bilineal

El error de posición de la interpolación del vecino más cercano es a lo sumo de medio píxel. Este error es perceptible en objetos que contengan líneas rectas que pueden aparecer como escaleras después de la transformación. La *interpolación bilineal* puede reducir este fenómeno y, de hecho, produce mejores resultados en la imagen transformada. Este método de interpolación considera cuatro vecinos del píxel a evaluar y supone que la variación de la función de brillo o intensidad es lineal dentro de este vecindario.

La **Figura 6** muestra el proceso de cálculo del nivel de gris por interpolación bilineal. Un método común de obtención del nivel de gris se hace calculando linealmente a lo largo de cada fila de la imagen

¹La distancia considerada aquí es la distancia euclidiana.

y luego a lo largo de cada columna. El valor estimado del nivel de gris del píxel es entonces,

$$f(x'_q, y'_q) = (1-a)(1-b)f(x, y) + (1-a)bf(x, y+1) + a(1-b)f(x+1, y) + abf(x+1, y+1) \quad (13)$$

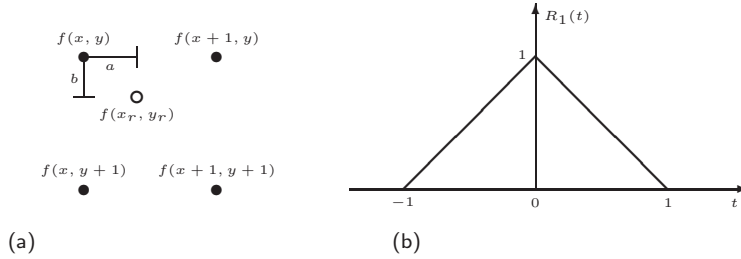


Figura 6: Interpolación bilineal. (a) Ubicación del píxel, al cual se le va a estimar la intensidad, y de sus cuatro vecinos necesarios para el cálculo. (b) Función de interpolación bilineal $R_1(t)$.

donde $f(\cdot)$ corresponde a la función de intensidad o nivel de gris de la imagen. Las coordenadas (x, y) provienen del truncamiento² de las coordenadas (x_r, y_r) en la imagen original. Luego, estas coordenadas pueden encontrarse como

$$(x, y) = (\text{trunc}(x_r), \text{trunc}(y_r)) \quad (14a)$$

$$(x+1, y) = (\text{trunc}(x_r) + 1, \text{trunc}(y_r)) \quad (14b)$$

$$(x, y+1) = (\text{trunc}(x_r), \text{trunc}(y_r) + 1) \quad (14c)$$

$$(x+1, y+1) = (\text{trunc}(x_r) + 1, \text{trunc}(y_r) + 1) \quad (14d)$$

Observe que las distancias a y b determinan el grado de ponderación de los niveles de gris de los cuatro vecinos. Obviamente, si a o b son cero o uno, el píxel a estimar coincide con las coordenadas de la imagen en la transformación horizontal o vertical o en ambas.

La interpolación bilineal introduce una mayor carga computacional en el proceso de definición del nivel de gris de cada píxel. Adicionalmente puede suceder que le añada algo de difuminado a la imagen transformada por el hecho de determinar los niveles de intensidad como promedio de un vecindario. Sin embargo, en el sentido pictórico, este método de interpolación es preferido al del vecino más cercano ya que reduce el efecto de forma de escalera en las líneas rectas.

Si se observa la ecuación (13), se puede pensar en aplicar cualquier función de interpolación $R(t)$ que tenga valor de cero por fuera de más o menos la distancia entre dos píxeles. Entonces, se tiene una generalización del proceso de interpolación. Por ejemplo, la función de interpolación bilineal para una señal cualquiera está definida como

$$R_1(t) = \begin{cases} t+1, & -1 \leq t \leq 0 \\ 1-t, & 0 < t \leq 1 \\ 0, & \text{lo demás} \end{cases} \quad (15)$$

cuya gráfica se muestra en la **Figura 6**. De esta manera, la interpolación bilineal puede ser vista como la suma de cuatro funciones de interpolación ponderadas de la siguiente forma

$$f(x'_q, y'_q) = R(-a)R(-b)f(x, y) + R(-a)R(1-b)f(x, y+1) + R(1-a)R(-b)f(x+1, y) + R(1-a)R(1-b)f(x+1, y+1) \quad (16)$$

²La operación de truncamiento (o *trunc*, en inglés) consiste en eliminar la parte fraccionaria de un número real. El número obtenido se aproxima al máximo entero menor.

2.3. Interpolación bicúbica

En el método de *interpolación bicúbica*, el nivel de gris de cada píxel de la imagen transformada se obtiene a partir de un polinomio cúbico. La función de interpolación puede ser la *cúbica* o *B-spline cúbica*. La función de interpolación B-spline cúbica se obtiene a partir de la convolución de cuatro funciones de Heavyside definidas como

$$h(t) = \begin{cases} 1, & -\frac{1}{2} \leq t \leq \frac{1}{2} \\ 0, & \text{en otros casos} \end{cases} \quad (17)$$

Entonces, la función de interpolación B-spline cúbica es

$$R_C(t) = \begin{cases} \frac{1}{6}(t+2)^3, & -2 \leq t < -1 \\ \frac{2}{3} - t^2 - \frac{1}{2}t^3, & -1 \leq t < 0 \\ \frac{2}{3} - t^2 + \frac{1}{2}t^3, & 0 \leq t < 1 \\ \frac{1}{6}(2-t)^3, & 1 \leq t \leq 2 \\ 0, & \text{los demás casos} \end{cases} \quad (18)$$

cuya gráfica se puede ver en la parte (a) de la **Figura 7**.

La función de interpolación bicúbica, también conocida como ‘sombrero mejicano’, se encuentra a partir de la definición generalizada

$$R_C(t) = \begin{cases} A_1|t|^3 + B_1|t|^2 + C_1|t| + D_1, & 0 \leq |t| < 1 \\ A_2|t|^3 + B_2|t|^2 + C_2|t| + D_2, & 1 \leq |t| \leq 2 \\ 0, & \text{los demás casos} \end{cases} \quad (19)$$

donde los factores se determinan a partir de las siguientes condiciones:

1. $R_C(0) = 1$; $R_C(1) = R_C(-1) = R_C(2) = R_C(-2) = 0$ y
2. Las derivadas $R'_C(0) = 0$, $R'_C(1) = -1$, $R'_C(-1) = 1$ y $R'_C(2) = R'_C(-2) = 0$.

Luego, la función de interpolación bicúbica queda

$$R_C(t) = \begin{cases} 1 - 2|t|^2 + |t|^3, & 0 \leq |t| < 1 \\ 4 - 8|t| + 5|t|^2 - |t|^3, & 1 \leq |t| \leq 2 \\ 0, & \text{los demás casos} \end{cases} \quad (20)$$

cuya gráfica se puede ver en la parte (b) de la **Figura 7**.

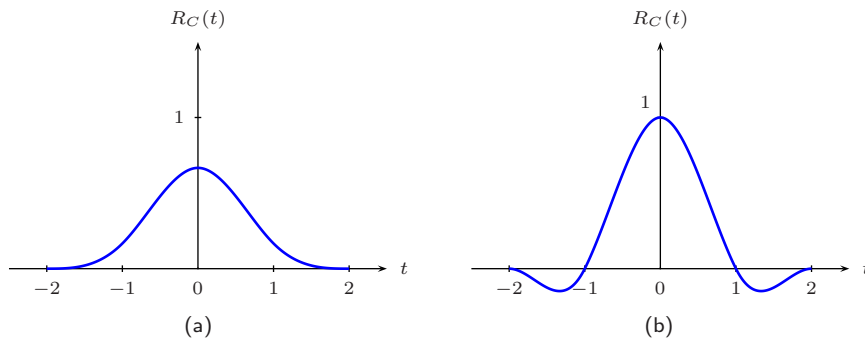


Figura 7: Funciones de interpolación B-spline cúbica en (a) y bicúbica o «sombrero mejicano» en (b).

En general, la función de interpolación está limitada a un vecindario de 4×4 píxeles por la complejidad computacional que implica tener vecindarios de mayor tamaño. En la **Figura 8** se muestra un vecindario

para la interpolación bicúbica en donde el píxel $f(x, y)$ es el vecino más cercano al píxel que se va a interpolar. El nivel de gris del píxel interpolado entonces se obtiene de la forma

$$f(x'_q, y'_q) = \sum_{i=-1}^2 \sum_{j=-1}^2 R_C(i-a)R_C(j-b)f(x+i, y+j) \quad (21)$$

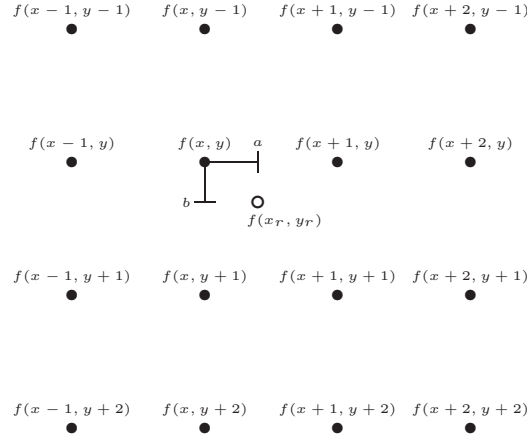


Figura 8: Píxeles involucrados en la interpolación bicúbica o B-spline cúbica en un vecindario de 4×4 .

teniendo en cuenta que las intensidades f de la ecuación tienen equivalentes con los píxeles de la imagen original a través de la transformación de coordenadas inversas. Observe que la misma ecuación es aplicable a la función de interpolación B-spline cúbica.

Con la interpolación bicúbica tampoco se obtiene el efecto de escalera en las líneas rectas o en los bordes definidos de los objetos como en el caso del vecino más cercano. También reduce el difuminado que se presenta en la interpolación bilineal. Es un método de interpolación que permite activar de manera más apropiada el acercamiento de imágenes en puntos específicos por su similitud de forma con la función sinc^3 que es la que permite reconstrucción perfecta de una señal muestreada. En general, la interpolación bicúbica preserva bastante bien los detalles de una imagen.

Como una muestra de las características visuales de los diferentes métodos de interpolación, en la **Figura 9** se muestra una imagen y dos ampliaciones, de una parte de ella, con dos métodos de interpolación. El primer método es el de interpolación de orden cero o vecino más cercano y el segundo es el método bilineal. Observe que las imágenes son en color. Los métodos se aplicaron a cada componente de color. Se puede observar que en la imagen interpolada con el vecino más cercano, los objetos muestran el efecto de «pixelado» como si la resolución en píxeles fuera baja. Sin embargo, los tamaños de las imágenes son iguales (384×384 es este caso). En realidad, el método de interpolación produce conjuntos de píxeles de la imagen interpolada con los mismos valores de intensidad o color de un solo píxel de la imagen original. Esto es debido al hecho que las coordenadas de varios píxeles de la imagen interpolada están cerca del mismo píxel de la imagen original cuando se halla su transformación inversa. El método de interpolación bilineal «corrige» este efecto porque la intensidad o el color de los píxeles se hace de manera ponderada y depende de la posición con respecto a sus vecinos cercanos. Por esto, la imagen parece más natural aunque un poco difuminada.

Las transformaciones geométricas son procesos que modifican la geometría de la imagen y están dentro de las que modifican el brillo de los píxeles a partir de otros, como en el caso de la interpolación del vecino más cercano, o a partir de vecindarios locales pequeños, como en el caso de la interpolación bilineal, bicúbica o B-spline cúbica. Debido a la carga computacional que implica, no es común utilizar vecindarios más grandes que 4×4 píxeles. Sin embargo, como se ha visto, se pueden utilizar diversos métodos de interpolación con diferentes resoluciones. La ecuación (21) puede modificarse para que el vecindario sea mayor junto con la función de interpolación más apropiada para la aplicación.

³La función sinc está definida como $\text{sinc}(x) = \frac{\sin(\pi x)}{\pi x}$.

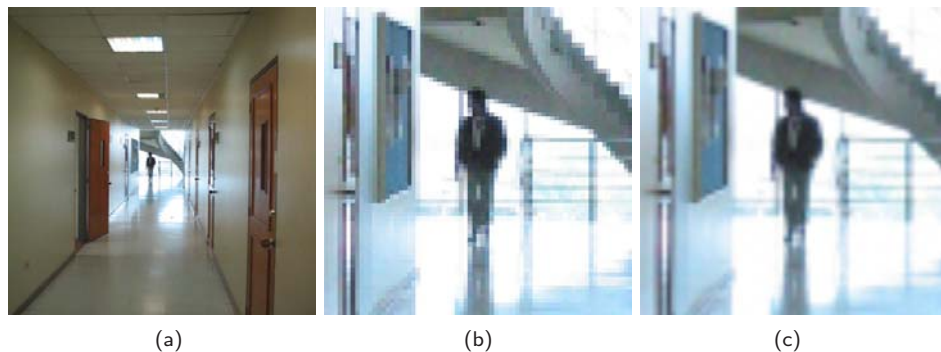


Figura 9: Ejemplo de interpolación en una transformación geométrica. A la imagen en (a) se le hace una ampliación (*zoom*) de 5.4 veces para obtener una imagen del mismo tamaño. En (b) se tiene el resultado de la ampliación con interpolación de orden cero o vecino más cercano. En (c) se tiene la misma ampliación pero el método de interpolación es el bilineal.

Bibliografía

- [1] Gonzalo Pajares Martinsanz y Jesús M. De La Cruz García, *Visión por Computador: Imágenes Digitales y Aplicaciones*, Alfaomega Grupo Editor, S.A. de C.V., México, D.F., 2002.
- [2] John C. Russ, *The Image Processing Handbook*, Fourth Edition, CRC Press LLC, Boca Ratón, Florida, USA, 2002.
- [3] Marcos Faúndez Zanuy, *Tratamiento Digital de Voz e Imagen y Aplicación a la Multimedia*, Alfaomega Grupo Editor, S.A. de C.V., México, D.F., 2001.
- [4] Rafael C. González and Richard E. Woods, *Digital Image Processing*, Second Edition, Prentice-Hall, Inc., New Jersey, USA, 2002.
- [5] Stanley I. Grossman, *Álgebra Lineal*, Quinta edición, McGraw-Hill, México, 1996.
- [6] Steven W. Smith, *The Scientist and Engineers Guide to Digital Signal Processing*, Second Edition, California Technical Publishing, San Diego, California, 1999.
- [7] William K. Pratt, *Digital Image Processing*, Second Edition, John Wiley & Sons, Inc., New York, USA, 1991.