¡Pero si estoy dibujando matrices!

La matemática (súper básica) detrás de las imágenes digitales

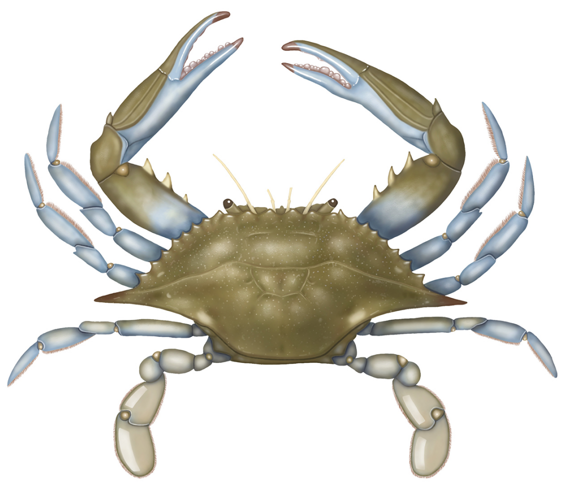
M. E. Bernaschini

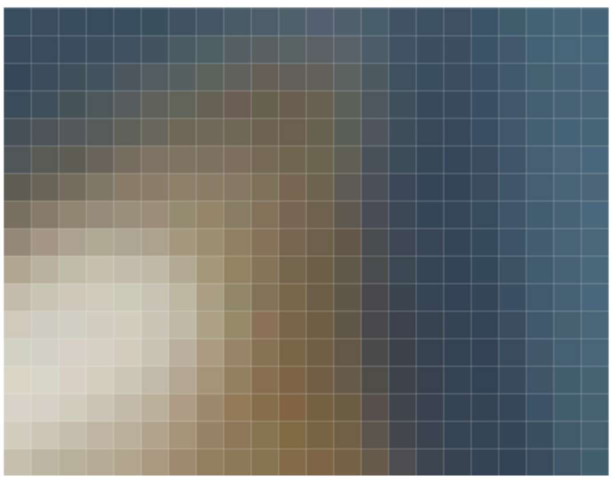
28 de julio de 2020

Hace algunos meses me propuse aprender dibujo digital. Hasta entonces sólo manejaba algunas técnicas tradicionales, como grafito y lápices de colores, así que me anoté en un curso online de ilustración de fauna marina con técnica digital. El curso me pareció una maravilla. Como trabajo final ilustré el cangrejo azul *Callinectes sapidus*, nativo de las costas del océano Atlántico en occidente y especie invasora en el Mediterráneo.

Al principio, dibujar con una tableta gráfica mirando la pantalla y no mi mano me resultaba bastante incómodo, pero sólo me llevó algunos días de práctica para entrenar a mi cerebro coordinar mano-pantalla. Más allá de esa dificultad motora, muchas fueron las ventajas que encontré al dibujar con técnica digital: se aceleran los procesos; me veo más predispuesta a experimentar sin miedo a meter la pata (gracias al Ctrl Z); se puede trabajar en capas, como si se estuvieran superponiendo papeles vegetales en donde se dibujan distintos aspectos de la ilustración en cada hoja. Pero de todas las ventajas, mi favorita es la posibilidad de hacer zoom.

Los detalles finos hacen que una ilustración cobre vida. Ya sea una textura intrincada bien resuelta, reflejos de luz precisos, bordes suaves, etc. En general, esos detalles finos son muy pequeños, y a escala real son difíciles de lograr. Es por ello que, en mi opinión, la posibilidad de hacer zoom (para facilitar el trazado de los detalles) es la ventaja del dibujo digital que ha enriquecido en mayor medida mis ilustraciones. Y es también la que me ha hecho caer en cuenta de que en realidad estoy dibujando matrices.





¿Qué es una matriz? y ¿cómo una imagen digital puede ser representada como una matriz?

Así como la materia está constituida por átomos y los organismos vivos por células, las imágenes digitales están compuestas por píxeles. Un **píxel** [1] es una celdilla cuadrada de un único color y sin un tamaño concreto. Por tanto, una imágen digital es una retícula rectangular formada por estas *unidades de color* que en su conjunto nuestros cerebros interpretan como un rostro, un paisaje o un cangrejo azul.

Ahora bien, si a cada color le asignamos un número esa cuadrícula de píxeles se transforma en nada más y nada menos que en una matriz.

Una **matriz** $n \times m$, A, es un arreglo rectangular de $n \cdot m$ números dispuestos en n filas y m columnas

$$A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nm} \end{pmatrix},$$

cada elementito a_{ij} se llama entrada ij y representa el número ubicado en la fila i y en la columna j. Por ejemplo, una matriz 2×3 podría ser la siguiente:

$$A = \begin{pmatrix} 3 & -1 & 3 \\ -2 & 2 & 7 \end{pmatrix}.$$

Imagina ahora que nos presentan un juego de mesa con fichas de muchas formas y colores, pero sólo nos muestran las fichas y no nos dicen nada acerca de cómo se juega ni cuáles son sus reglas. Evidentemente no vamos a poder jugar. Lo mismo sucede en matemática. Las matrices son como las fichas de un juego, veamos entonces cómo hay que jugar.

A las reglas de este juego matricial las vamos a llamar operaciones.

Operación 1: Dos matrices de igual tamaño se pueden sumar. La suma va a ser una nueva matriz del mismo tamaño en donde cada entrada ij va a ser la suma de las entradas ij de las matrices que se están sumando. Por ejemplo:

$$\begin{pmatrix} 3 & -1 \\ -2 & 2 \\ 0 & 7 \end{pmatrix} + \begin{pmatrix} -1 & 0 \\ 3 & 5 \\ 4 & 6 \end{pmatrix} = \begin{pmatrix} 2 & -1 \\ 1 & 7 \\ 4 & 13 \end{pmatrix}.$$

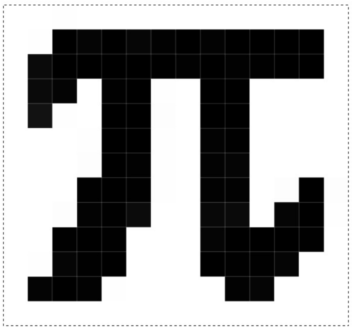
[1] La palabra *píxel* deriva de las palabras inglesas *picture* y *element*.

Operación 2: Dos matrices de tamaños adecuados se pueden multiplicar. Por ejemplo:

$$\begin{pmatrix} 3 & -1 \\ -2 & 0 \end{pmatrix} \cdot \begin{pmatrix} 2 \\ 3 \end{pmatrix} = \begin{pmatrix} 3 \cdot 2 + (-1) \cdot 3 \\ (-2) \cdot 2 + 0 \cdot 3 \end{pmatrix} = \begin{pmatrix} 3 \\ -4 \end{pmatrix}.$$

Además de sumar y multiplicar, con las matrices podemos realizar una gran variedad de operaciones. Todo depende del juego que queramos jugar. Una de las aplicaciones más conocidas es en la resolución de sistemas de ecuaciones lineales (esos en donde nos piden hallar la x y la y). Pero esta vez vamos a jugar al juego de las imágenes digitales.

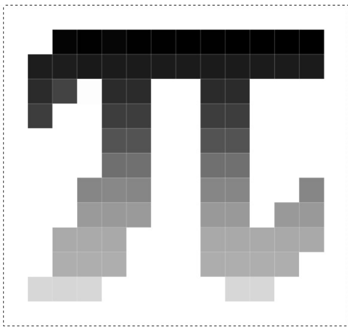
Comenzemos por la situación más sencilla: una imagen en blanco y negro.



En este dibujito se pueden apreciar a simple vista los píxeles que lo componen. Ahora, si al píxel negro le asignamos el número 0 y al píxel blanco el número 1, entonces la matriz asociada quedaría así:

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

Pero claro, no todas las imágenes son en blanco y negro. Podríamos tener una, por ejemplo, en escala de grises.



En este caso, claramente, vamos a necesitar más de dos números para construir su matriz asociada. La cantidad de números necesarios va a ser igual a la cantidad de tonos de grises con los que estemos trabajando, ya que a cada tono le tenemos que asignar un número distinto. Por ejemplo, si la cantidad de tonos son 12, donde el 0 representa al negro y el 11 al blanco, entonces la matriz de la imagen anterior tendría esta pinta:

$$\begin{pmatrix}
11 & 11 & 11 & 11 & 11 & 11 & 11 & 11 & 11 & 11 & 11 & 11 & 11 & 11 \\
11 & 11 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 11 \\
11 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 11 \\
11 & 2 & 2 & 11 & 2 & 2 & 11 & 11 & 2 & 2 & 11 & 11 & 11 & 11 \\
11 & 3 & 11 & 11 & 3 & 3 & 11 & 11 & 3 & 3 & 11 & 11 & 11 & 11 \\
11 & 11 & 11 & 11 & 4 & 4 & 11 & 11 & 4 & 4 & 11 & 11 & 11 & 11 \\
11 & 11 & 11 & 11 & 5 & 5 & 11 & 11 & 5 & 5 & 11 & 11 & 11 & 11 \\
11 & 11 & 11 & 6 & 6 & 6 & 11 & 11 & 6 & 6 & 11 & 11 & 6 & 11 \\
11 & 11 & 11 & 7 & 7 & 7 & 11 & 11 & 7 & 7 & 11 & 7 & 7 & 11 \\
11 & 11 & 8 & 8 & 8 & 11 & 11 & 11 & 8 & 8 & 8 & 8 & 8 & 11 \\
11 & 11 & 9 & 9 & 9 & 11 & 11 & 11 & 9 & 9 & 9 & 9 & 11 & 11 \\
11 & 10 & 10 & 10 & 11 & 11 & 11 & 11 & 11 & 10 & 10 & 11 & 11 & 11 \\
11 & 11 & 11 & 11 & 11 & 11 & 11 & 11 & 11 & 11 & 11 & 11 & 11 & 11
\end{pmatrix}$$

¿Y qué pasa con las imágenes en colores? Bueno, primero deberíamos hablar de los modelos de color. Un **modelo de color** es una representación abstracta de los diferentes colores a través de números. La cantidad de números que define a cada color depende del modelo. Por ejemplo, en el modelo RGB[2] cada color se compone de la combinación (adición) de tres colores base: rojo, verde y azul. Luego, a cada píxel se le asocia un terna

[2] Iniciales de las palabras inglesas *red*, *green* y *blue*

ordenada (r, g, b), donde r especifica la intensidad de rojo, g la intensidad de verde y b la intensidad de azul. La cantidad de intensidades distintas de un sólo color base depende de la **profundidad de color** y se mide en bits[3] por píxel. Una profundid de 24 bits por píxel en el model RGB quiere decir que por cada color base tenemos 256 intensidades posibles, lo que implica que nuestra paleta de colores tiene $256 \cdot 256 \cdot 256 = 16.777.216$ de colores distintos, cantidad suficiente para que el ojo humano perciba una imagen de calidad[4]. Otro modelo muy popular, y que se utiliza en imágenes para impresión, es el model CMYK. En este caso la paleta de colores se consigue combinando los colores cian, magenta, amarillo y negro.

Momento... si en el modelo RGB de 24 bits a cada píxel se lo identifica con una terna ordenada, ¿cómo se construye la matriz asociada a una imagen multicolor? Bueno, el problema es más o menos sencillo, tenemos que transformar las ternas ordenadas en números enteros de forma tal que a cada número entero no negativo menor que $16.777.216$ le corresponda una única terna ordenada (esto en matemática se conoce por el nombre de **función inyectiva**). Varios programas gráficos utilizan la siguiente función con esa propiedad:

$$f(r, g, b) = 256^2 \cdot r + 256 \cdot g + b.$$

Veamos un ejemplo. Supongamos que cada color base tiene un rango de intensidad de 0 a 255. Entonces, el color $(57, 136, 98)$ corresponde al número $256^2 \cdot 57 + 256 \cdot 136 + 98 = 12.648.546$. Ahora, ¿qué color (terna ordenada) se identifica con el número entero $11.975.134$? Por la forma en la que está construida la función f, r resulta ser el cociente de dividir $11.975.134$ por 256^2, g el cociente de dividir el resto de la división anterior por 256 y b el resto de la última división. En otras palabras (o mejor dicho en otros números):

$$11.975.134 = 256^2 \cdot 182 + 47582,$$

$$47582 = 256 \cdot 185 + 222.$$

Por lo tanto, el color que se identifica con el número entero $11.975.134$ es $(182, 185, 222)$.

Dicho esto, ahora sí ya estamos en condiciones de construir la matriz asociada a cualquier imagen multicolor.

Retocando matrices

¿Quién no ha retocado alguna vez una imagen digital? Convertirla en escala de grises, aplicarle un filtro sepia, modificarle el brillo y el contraste, etc. Apuesto un guiso a que todos lo hicimos en algún momento.

[3] Un **bit** (**b**inary un**it**) es la unidad mínima de información que utiliza una computadora, y puede valer 0 o 1.

[4] En la profundidad de 24 bits por píxel del modelo RGB, la intesidad de cada color base se identifica con un número menor a 256 (desde 0 hasta 255), y todo número (entero no negativo) menor que 256 se puede almacenar en 8 bits. Por ejemplo:

$$157 = 2^7 \cdot 1 + 2^6 \cdot 0 + 2^5 \cdot 0 + 2^4 \cdot 1 + 2^3 \cdot 1 + 2^2 \cdot 1 + 2^1 \cdot 0 + 2^0 \cdot 1,$$

entonces el 157 se encuentra codificado en los bits 10011101. Luego, por cada píxel hacen falta almacenar 8 bits por intesidad de color, y como son tres colores base, necesitamos $3 \cdot 8 = 24$ bits por píxel.

Ahora, si una imagen digital es una matriz, modificar una imagen implica modificar una matriz. Lo que nos convierte a todos en algebristas inadvertidos.

A grandes rasgos, existen dos tipos de tratamientos para imágenes digitales: aquellos en donde se modifica el color del píxel, y aquellos en donde se modifica la posición del píxel sin cambiar su color. Dentro del primer grupo se encuentran, por ejemplo, el ajuste de brillo y contraste, la convesión a escala de grises, el efecto sepia y el ajuste de color. En el segundo grupo podemos encontar la rotación, la reflexión y la traslación.

Veamos algunos ejemplos del primer grupo:

Conversión a escala de grises

Tomemos un píxel cualquiera de nuestra imagen. En el modelo RGB su color va a estar determinado por una terna ordenada (r, g, b). Para transformar ese color en un tono de gris adecuando lo que debemos hacer es calcular el promedio de las tres componentes, pues los grises son píxeles en donde $r = g = b$. Entonces, si definimos la matriz T de la siguiente manera:

$$T = \begin{pmatrix} \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{pmatrix},$$

la operación de conversión a escala de grises sería multiplicar cada terna ordenada (r, g, b) por la matriz T:

$$T \cdot \begin{pmatrix} r \\ g \\ b \end{pmatrix} = \begin{pmatrix} \frac{r+g+b}{3} \\ \frac{r+g+b}{3} \\ \frac{r+g+b}{3} \end{pmatrix}.$$

Notar que el nuevo color del píxel quedaría así:

$$\left(\frac{r+g+b}{3}, \frac{r+g+b}{3}, \frac{r+g+b}{3} \right),$$

y como todas sus componentes son iguales, sería efectivamente un tono de gris.

Este tipo de operación se llama **transformación lineal**, y es también la que se utiliza para aplicar el filtro sepia (con una matriz T diferente).

Ajuste del canal azul

Supongamos ahora que queremos ajustar el color azul sin modificar el rojo y el verde. Entonces, lo que debemos hacer es sumar a cada terna ordenada (r, g, b) la matriz columna C:

$$C = \begin{pmatrix} 0 \\ 0 \\ f \end{pmatrix},$$

donde f es un número que depende del grado de ajuste que deseemos. Luego, el nuevo color del píxel sería así:

$$\begin{pmatrix} 0 \\ 0 \\ f \end{pmatrix} + \begin{pmatrix} r \\ g \\ b \end{pmatrix} = \begin{pmatrix} r \\ g \\ b+f \end{pmatrix}.$$

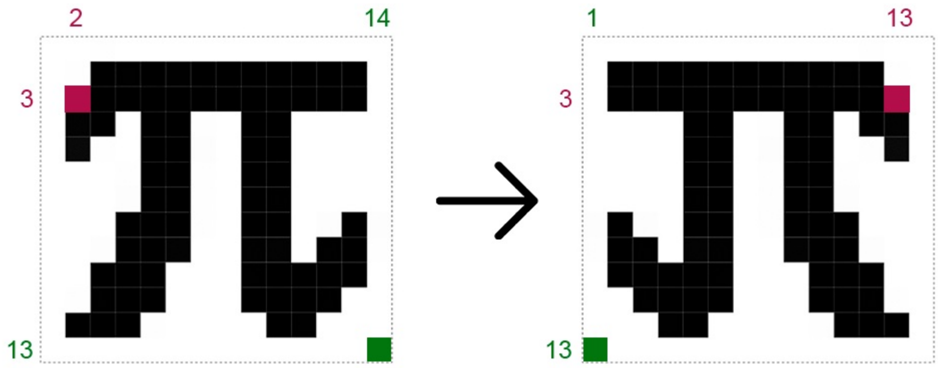
Análogamente se pueden ajustar los canales rojo y verde.

El ajuste de brillo también se obtiene sumando a cada terna ordenada una matriz columna, pero una con todas sus entradas iguales. En cambio, el contraste se logra combinando de cierta manera los dos tipos de operaciones anteriores (la transformación lineal y la suma de una matriz columna).

Veamos ahora un ejemplo del segundo grupo.

Reflexión

Seguramente habrás notado que cuando te sacas una selfie la imagen sale reflejada, como si te estuvieras mirando en un espejo. Entonces, ¿qué retoque (operación) podemos aplicarle a esa imagen (matriz) para darla vuelta? Bueno, cada píxel de la imagen ocupa una posición dentro de la matriz, esa posición está dada por la columna en la que está, que la denotamos con la letra i, y la fila a la cual pertenece, denotada por la letra j. Si la matriz tiene dimensiones $n \times m$ (n filas y m columnas), entonces al píxel en la posición (i, j) lo tenemos que llevar a la posición $(i, m + 1 - j)$. Por ejemplo, si nuestra matriz tiene 13 filas y 14 columnas, al píxel de la posición $(3, 2)$ lo tenemos que trasladar a la posición $(3, 13)$, y al píxel de la posición $(13, 14)$ a la posición $(13, 1)$.



Pero, ¿cómo podemos expresar esta operación de forma matricial? Para responder a esta pregunta primero vamos a representar la posición (i, j) como una matriz columna de tres componentes:

$$\begin{pmatrix} i \\ j \\ 1 \end{pmatrix}.$$

Entonces, la operación que hace la reflexión va a estar dada por la siguiente transformación lineal:

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & -1 & m+1 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} i \\ j \\ 1 \end{pmatrix} = \begin{pmatrix} i \\ m+1-j \\ 1 \end{pmatrix}.$$

La rotación también se logra aplicando una transformación lineal, y la traslación sumando una matriz columna a la posición.

Por supuesto, existen muchas más operaciones con matrices que podemos aplicar para retocar imágenes. ¿No te dan ganas de inventar la tuya y ver qué pasa? Por mi parte voy a seguir experimentando con mi cangrejo azul.

Nota

En este artículo hemos hablado de imágenes digitales formadas por píxeles (llamadas **mapa de bits** o **bitmaps**), pero existe otro tipo y muy distinto: las imágenes vectoriales. Una **imagen vectorial** es una imagen digital que está constituida por objetos (contornos y rellenos) definidos a través de relaciones matemáticas. La ventaja de este tipo de formato es que permite redimensionar la imagen sin sufrir pérdida de calidad. Se utiliza popularmente para la creación de ilustraciones, gráficos, logos y símbolos.

La ciencia del procesamiento de imágenes

Compresión de imágenes, eliminación de ruido, detección de bordes, mejora de la calidad, clasificación... La lista sigue y es muy extensa. Todo se puede abordar desde la matemática, la estadística y el machine learning. El procesamiento de imágenes es un área de las ciencias de la computación que, desde sus orígenes en la década del 60, sigue siendo muy activa y cada vez más interdisciplinaria. Sus aplicaciones alcanzan tanto a otras ramas científicas como a la industria.

Pero no todo es color esperanza. Históricamente, la computación es un área integrada mayoritariamente por hombres. Y mucho tienen que ver los estereotipos de género. En la década del 70 un científico de una universidad norteamericana usó una fotografía (recortada) de la revista Pleyboy de una modelo (Lena) para probar sus algoritmos de procesamiento de imágenes. Desde entonces la misma imagen fue (y es) utilizada por la comunidad de investigadores. Hoy en día se promueve su desuso, ya que la propia imagen, extendida en tantísimas publicaciones científicas, refuerza el estereotipo de género (varón heterosexual) dentro del ámbito de la computación. [5]

En la Argentina existen varias iniciativas que intentan potenciar la diversidad de género en áreas de la ciencia y la tecnología. Sus objetivos principales son motivar, formar y acompañar a niñas, mujeres y personas

[5] Enlace a la nota

de género no binario a través de charlas, encuentros, talleres y eventos diversos. Para mayor información visitar sus sitios web: Mujeres en Tecnología, Chicas en Tecnología y Red Argentina de Género, Ciencia y Tecnología.

Referencias

[1] GIMP, aplicaciones didácticas. La imagen digital. Ministerio de educación de España.

[2] El álgebra lineal y el procesamiento digital de imágenes. Anibal Rodriguez.

[3] Curso de introducción al procesamiento de imágenes radiológicas en ámbito clínico. P. Pérez y M. Valente.