



Tratamiento de imágenes

Representación y resolución de imágenes digitales

Héctor Alejandro Montes
h.a.montes@fi.uaemex.mx
<http://fi.uaemex.mx/h.a.montes>

Advertencia

No use estas diapositivas como referencia única de estudio durante este curso. La información contenida aquí es una guía para las sesiones de clase y de estudio futuro. Para obtener información más completa, refiérase a la bibliografía listada en la última diapositiva.

Representación de Imágenes Digitales

- Una vez que una imagen ha sido *digitalizada* podemos almacenarla y/o representarla como en una matriz de $M \times N$

$$A = \begin{pmatrix} a_{1,1} & a_{1,2} & L & a_{1,N} \\ a_{2,1} & a_{2,2} & L & a_{2,N} \\ M & M & O & M \\ a_{M,1} & a_{M,2} & L & a_{M,N} \end{pmatrix}$$

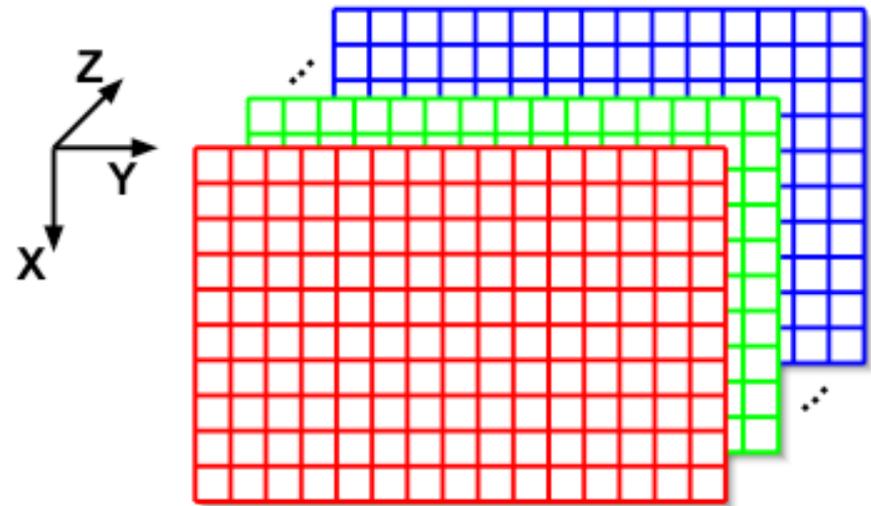
- Cada elemento de la matriz es un *píxel*

Representación de Imágenes Digitales

- La intensidad adquirida por el sensor en las coordenadas $(x,y) = (0,0)$ corresponde al elemento $a_{11} = f(0,0)$
 - En general $a_{M,N} = f(M-1,N-1)$
- Por convención, el eje **X** crece hacia abajo y el eje **Y** crece hacia la derecha

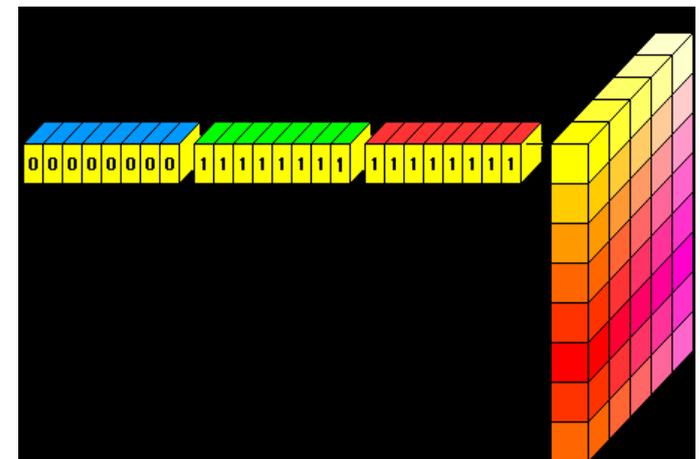
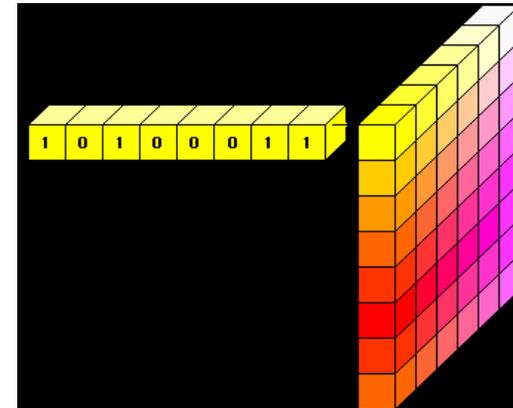
Representación de Imágenes Digitales

- Cuando el sistema de adquisición tiene un sólo filtro óptico, la imagen es *monocromática*
 - Aunque el filtro puede no ser estrictamente monocromático, es decir de una sólo longitud de onda.
- Si el sistema de adquisición de imágenes tuviese más de un filtro, la matriz **a** se vuelve multi-dimensional



Representación de Imágenes Digitales

- Si R es el número de filtros del sistema de adquisición, el sistema tiene:
 - $R = 1$ filtro \Rightarrow *Monocromática*
 - $R = 3$ filtros \Rightarrow *Color*
 - $R > 3$ filtros \Rightarrow *Multiespectral*
 - $R \gg 3$ filtros \Rightarrow *Hiperespectral*



Representación de Imágenes Digitales

- Si el número de niveles de cuantificación (*niveles de grises*) del filtro i -ésimo R_i es $L_i = 2^{k_i}$, entonces, para almacenar una imagen necesitamos:

$$b = \sum_{i=1}^R M \cdot N \cdot k_i$$

- Donde b es el número de *bits* requerido para almacenar una imagen digitalizada
- Si todos los filtros R_i usan un cuantificador con el mismo número de niveles, entonces $k_i = k_j \mid i \neq j$, por tanto:

$$b = M \cdot N \cdot R \cdot k$$

Representación de Imágenes Digitales

- El color es mas costoso en términos de espacio.
- El tiempo de proceso y de transmisión requeridos es mayor.
- **Ejemplo:** La transmisión de datos usualmente se realiza en paquetes de un bit de inicio, un byte (8 bits) de información y un bit de paro.
 - ¿Cuántos minutos tomaría transmitir una imagen de 512×512 con 256 niveles de gris a 9,600 *bits/s*?
 - ¿Cuál deberá ser la capacidad en bits/s de un canal de transmisión digital si las imágenes del inciso anterior (25 imágenes por segundo) van a ser transferidas?

Representación de Imágenes Digitales

- Color usualmente se entiende como 3 *filtros* o *bandas*
 - Aplica también a imágenes *multiespectrales* e *hiperespectrales*
- El **color** es **útil** en algunas aplicaciones (medicina, agricultura, entre otras)
 - En algunas aplicaciones es innecesario usar color

Rango Dinámico

- Es la relación entre el valor más alto y el más bajo que un fotoreceptor puede adquirir y se expresa en decibeles (dB)

$$RD (dB) = 10 \log_{10} \left(\frac{V_{MAX}}{V_{MIN}} \right)$$

- El *ojo humano* es capaz de ver objetos cuando la iluminación es escasamente la de la luna y cuando es un día soleado
 - Esto equivale a un **RD=90 dB**
- Actualmente *no existe* ningún dispositivo de adquisición de imágenes con este mismo rango dinámico

Rango Dinámico

- Si se **rebasa el máximo** de intensidad que el fotorreceptor admite, se dice que se ha **saturado**
 - Usualmente se produce un color **blanco**
- Cuando **no se alcanza el mínimo**, el fotorreceptor **no emite respuesta**
 - Usualmente se produce un color **negro**

Rango Dinámico - HDRI

- **High Dynamic Range Imaging** (HDRI): conjunto de métodos que busca ampliar el rango dinámico de un dispositivo
 - Con HDRI se pueden representar con mayor precisión el rango de niveles de intensidades de una escena real
 - Desde luz solar directa hasta luz estelar tenue
 - Inicialmente desarrollada para juegos
 - Actualmente se usa también en fotografía, cine, entre otras
 - Complejidad alta, especialmente en la adquisición

Rango Dinámico - HDRI

- Es común lograr HDRI usando **varias imágenes** con **diferente nivel de exposición**



Resolución

- Existen varias **definiciones**:
 - Distancia que cubre un pixel
 - Distancia que puede detectar un sensor
 - Tamaño mínimo de un elemento para que pueda ser discriminado
 - Tamaño que un píxel representa en el mundo real
- En general **describe el detalle contenido en una imagen**

Resolución

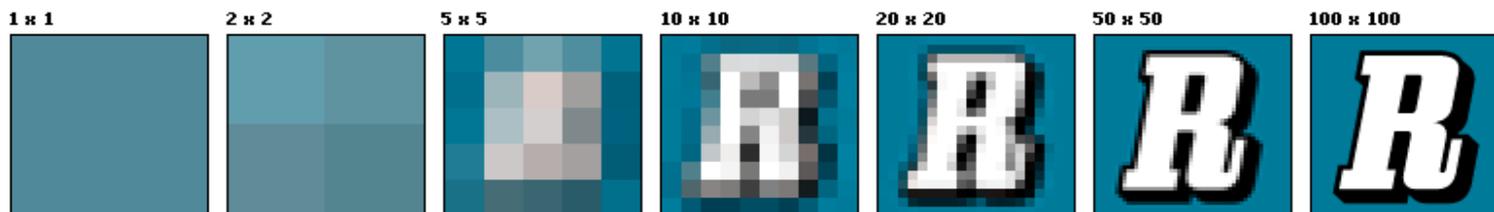
- Indica la relación entre la **distancia física** y el **tamaño del pixel**
- Afecta la **calidad** de la imagen porque determina qué objetos serán correctamente captados o desplegados
- Cuantifica **qué tan cerca** pueden estar los píxeles uno del otro y ser resultados de manera visible
 - Tamaños físicos: líneas por *mm* o por *pulgada*
 - Tamaño de la imagen: alto y ancho

Resolución

- Existen varias formas de expresar la resolución en una imagen digital:
 - Resolución de píxeles
 - Resolución espacial
 - Cuadros distinguibles
 - Resolución espectral
 - Resolución temporal
 - Resolución radiométrica

Resolución de píxeles

- Se mide por el **conteo de píxeles**
 - Aunque estándares internacionales, de Japón y EE.UU. dicen que no debe ser así
 - Una imagen de $N \times M$ puede tener menos líneas que las de una TV
- La **convención** es usar 2 números enteros positivos: **N** para columnas (ancho) y **M** para renglones (alto)
- Otra **convención** es describirla como el **número total de píxeles**, típicamente en megapíxeles: $N \times M / 1,000,000$
- Otra **convención** es describirla en **píxeles por unidad** de longitud o por unidad de área (ppi, dpi, ppsi)
 - Esto se conoce como **densidad de píxeles**



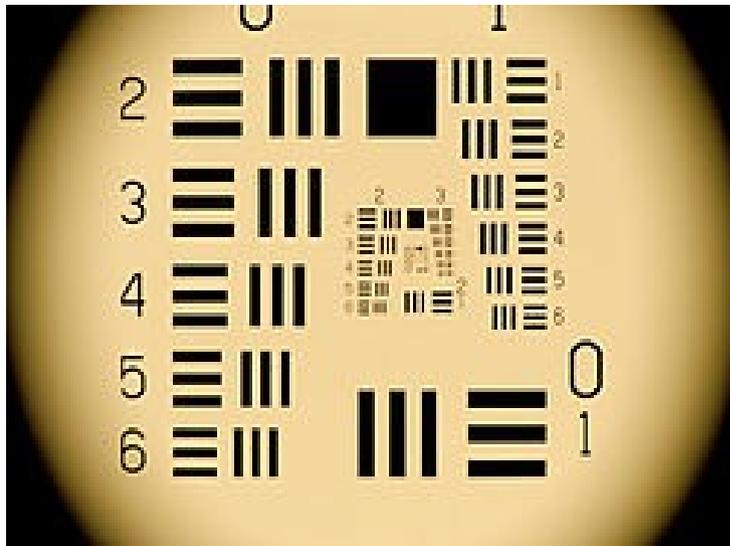
Densidad de píxeles

- Es la medida de la **resolución de dispositivos**: monitores, proyectores, scanners y sensores de cámaras digitales
- **PPI** puede describir también la resolución en píxeles de una imagen **al imprimirse** dentro de un espacio específico
 - Una imagen de 100x100 píxeles que se imprime en 1 pulg² tiene una resolución de 100 ppi
 - Es común 300 y 600 ppi
- Ninguna de estas resoluciones son **resoluciones verdaderas** (excepto la de sensores), pero se usan como tales

Resolución espacial

- Es la medida de qué tan **cerca** se pueden resolver las líneas de una imagen
- Depende de las propiedades del sistema que creó la imagen y no sólo los ppi
- Para usos prácticos, la claridad de una imagen se decide por la resolución espacial y no por el número de pixeles: **número de pixeles por unidad de longitud**
- Usualmente en monitores es de 72 a 100 líneas por pulgada
- En los **scanners**, es más común utilizar la **resolución óptica** para distinguir la resolución espacial del número de ppi

Resolución espacial

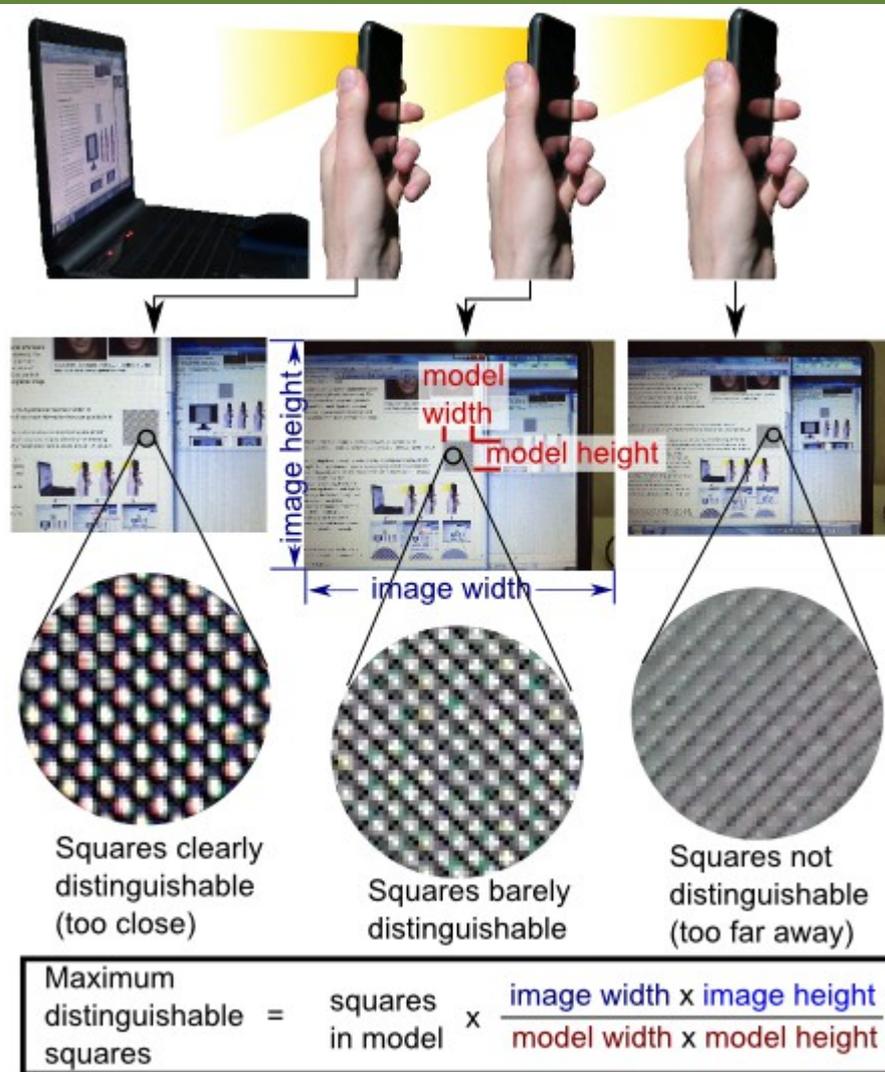


- Patrón utilizado por la USAF para determinar la resolución espacial

- La imagen de la izquierda tiene un mayor conteo de píxeles, pero es peor en resolución espacial

Cuadros distinguibles

- Integra **resolución espacial** y el **área de una imagen**. Es el número máximo hipotético de *cuadros distinguibles* que pueden caber en una imagen
- Escena de **3264x2448** pixels (8 megapixel), área modelo cubre **322x322** pixels donde los cuadros apenas se distinguen, resulta un estimado de **770,000** cuadros distinguibles



Resolución espectral

- Las imágenes en *color* distinguen luz en diferentes espectros
- Las imágenes multi-espectrales resuelven diferencias más finas de espectro de las necesarias para reproducir color
 - Esto es la intensidad de cada banda que se ha adquirido y creado

Resolución temporal

- Las cámaras de **cine** y las de **alta velocidad** resuelven eventos a diferentes tiempos (velocidades)
- Para cine es usual de **15 a 30 cuadros** por segundo (frames/s)
- Cámaras de alta velocidad lo hacen de **100 a 1000** frames/s o mayores

Resolución radiométrica

- Determina qué tan finamente **un sistema puede representar** las diferencias de intensidad
- Usualmente expresada como el **número de niveles** o un **número de bits**
 - 8 bits o 256 niveles
- Entre más alta sea, mejor se pueden representar las diferencias de intensidad
- En la práctica, la resolución radiométrica se ve limitada por el nivel de **ruido** y no por el número de bits utilizados

Modificar la resolución: Zoom

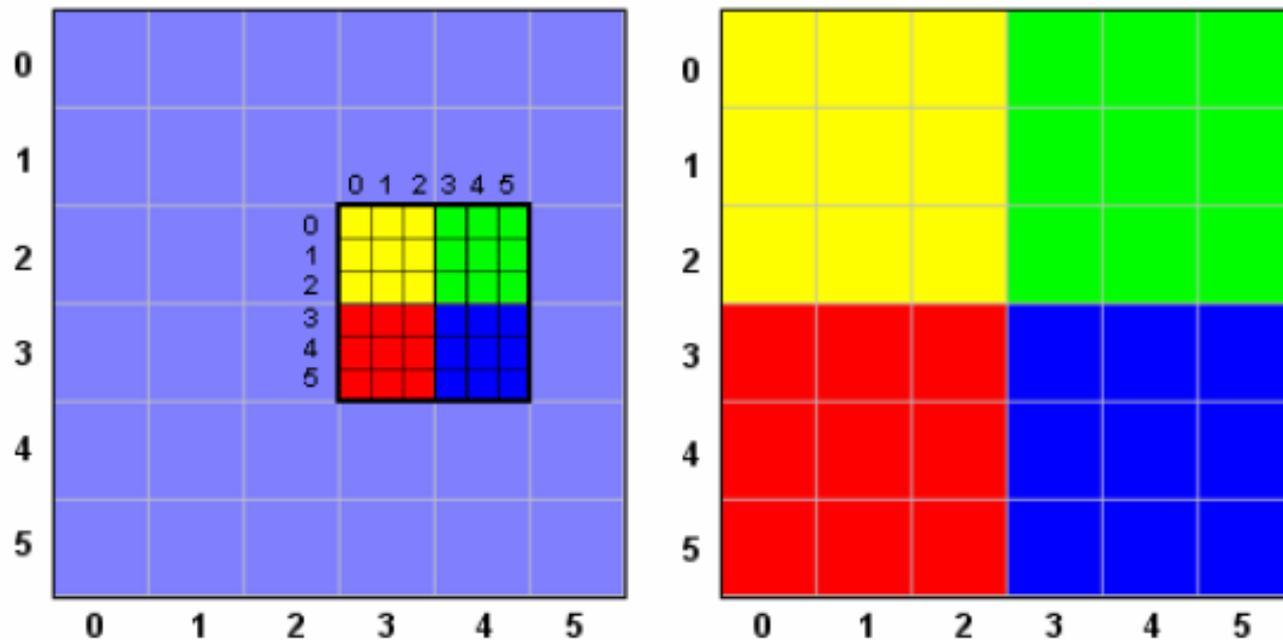
- Conceptualmente la operación de **zoom** es muy simple:



- La parte seleccionada de la imagen de la izquierda es la ROI
- En la operación **zoom**, la ROI se amplía para llenar el área de trabajo completa (*buffer*)

Algoritmo

- Considere la ROI de 6x6 de la imagen de la izquierda
- Haremos *zoom* en la region entre (3,2) y (5,4) para producir la imagen de 6x6 de la derecha



Algoritmo

- Calcular las proporciones x_ratio y y_ratio
 - Dos columnas en la imagen origen deben extenderse para caber en seis columnas en la imagen destino
 - Esto produce una proporción de 1/3 para X y para Y

$$x_{ratio} = y_{ratio} = \frac{2 \text{ pixeles del origen}}{6 \text{ pixeles del destino}} = \frac{1}{3}$$

- Calcular las coordenadas del pixel origen que corresponden a $destino(0,0)$. El valor del pixel en (x,y) se obtiene copiando directamente desde (x',y') :

$$x' = 0 * 0.333 + 3 = 3$$

$$y' = 0 * 0.333 + 2 = 2$$

Algoritmo (2)

- Asignar a *destino(0,0)* el contenido de *origen(x',y')*
 - El paso anterior produjo $x'=3$ y $y'=2$, por lo que:
destino(0,0) = origen(3,2) = Amarillo
- Repetir los pasos 2 y 3 para todos los (x,y) en el *destino*. En general, utilizamos la ecuación:

$$\begin{aligned} \textit{destino}(x, y) &= \textit{origen}(x', y') \textit{ donde} \\ x' &= \textit{floor}(x * x_{ratio} + x_{offset}) \\ y' &= \textit{floor}(y * y_{ratio} + y_{offset}) \end{aligned}$$

Resultados

- Valores calculados para el destino entre (0,0) y (3,3):

$x' = 0 * .333 + 3 = 3$ $y' = 0 * .333 + 2 = 2$	$x' = 1 * .333 + 3 = 3.333$ $y' = 0 * .333 + 2 = 2$	$x' = 2 * .333 + 3 = 3.333$ $y' = 0 * .333 + 2 = 2$
$x' = 0 * .333 + 3 = 3$ $y' = 1 * .333 + 2 = 2.333$	$x' = 1 * .333 + 3 = 3.333$ $y' = 1 * .333 + 2 = 2.333$	$x' = 2 * .333 + 3 = 3.666$ $y' = 1 * .333 + 2 = 2.333$
$x' = 0 * .333 + 3 = 3$ $y' = 2 * .333 + 2 = 2.666$	$x' = 1 * .333 + 3 = 3.333$ $y' = 2 * .333 + 2 = 2.666$	$x' = 3 * .333 + 3 = 3.999$ $y' = 2 * .333 + 2 = 2.666$
$x' = 0 * .333 + 3 = 3$ $y' = 3 * .333 + 2 = 2.999$	$x' = 1 * .333 + 3 = 3.333$ $y' = 3 * .333 + 2 = 2.999$	$x' = 3 * .333 + 3 = 3.999$ $y' = 3 * .333 + 2 = 2.999$

Tarea

- Escriba un programa que tome una imagen de entrada y realice la operación **zoom** para un área arbitrariamente seleccionada y para toda el “área visible”.
- Determine su propia área visible.
- Utilice una imagen de al menos 640x480.
- Implemente utilizando ImageMagick u OpenCV.
- Los procedimientos implementados formarán parte de su Proyecto Final.