



SEGMENTACIÓN DE IMÁGENES MÉDICAS

Tesina de Grado

DIRECTOR: Claudia Russo

AUTOR: Walter Maximiliano Martínez Krawczuk

Legajo 2748/2

Buenos Aires – Argentina

Email: info@wmmk.com.ar

ÍNDICE

CAPÍTULO PRIMERO: Fundamentos teóricos

- 1 Imágenes digitales
- 2 Captura de imágenes
- 3 Definiciones:
 - 3.1 Muestreo
 - 3.2 Resolución
 - 3.3 Resolución espacial
 - 3.4 Conectividad de píxeles
 - 3.5 Ruido
 - 3.6 Artefactos
 - 3.7 Volumen parcial
 - 3.8 No homogeneidad
 - 3.9 Forma
- 4 Color
 - 4.1 Modelo RGB
 - 4.2 Otros modelos de colores

CAPÍTULO SEGUNDO: Representación manipulación y almacenamiento de imágenes

- 5 Representación de imágenes
 - 5.1 1 bit por píxel (2 tonos)
 - 5.2 8 bits por píxel
 - 5.3 15 bits por píxel
 - 5.4 16 bits por píxel
 - 5.5 24 bits por píxel
 - 5.6 32 bits por píxel
- 6 Manipulación de imágenes digitales
- 7 Medios de almacenamiento
- 8 Formatos de archivo
 - 8.1 BMP
 - 8.2 JPG

- 8.3 TIFF
- 8.4 GIF
- 8.5 PNG
- 9 Tratamiento de imágenes digitales
 - 9.1 Manipulación de colores
 - 9.2 Conversiones entre representaciones
 - 9.3 Escala de grises
 - 9.4 Cálculo del histograma de una imagen
 - 9.5 Ecuación del histograma

CAPÍTULO TERCERO:

- 10 Introducción
 - 10.1 Binarización por detección de umbral
 - 10.2 Selección del umbral óptimo
 - 10.3 Umbrales basados en varias variables
 - 10.4 Crecimiento de Regiones
 - 10.5 Crecimiento de regiones mediante adición de píxeles
 - 10.6 Crecimiento y división
 - 10.7 División y fusión de imágenes
 - 10.8 Extracción de regiones por el color
 - 10.9 Segmentación basada en clustering

CAPÍTULO CUARTO: Imágenes Médicas

- 11 Tomografía Computada (TC)
 - 11.1 Definición
 - 11.2 Aplicaciones
 - 11.3 Obtención de imágenes por TC
 - 11.4 Ejemplos de TC
- 12 Resonancia Magnética (RMI)
 - 12.1 Definición
 - 12.2 Aplicaciones
 - 12.3 Obtención de imágenes por RMI

- 12.4 Ejemplos de RMI
- 12.5 TC vs. RMI
- 13 Formatos estándar de imágenes médicas
 - 13.1 DICOM

CAPÍTULO QUINTO: Segmentación de Imágenes Médicas

- 14 Clasificación de técnicas de segmentación
 - 14.1 Contextuales
 - 14.2 No Contextuales
- 15 Algoritmos y técnicas de segmentación
 - 15.1 Umbralización
 - 15.2 Similitud de Regiones (Semilla)
 - 15.3 Crecimiento de Regiones
 - 15.4 División y mezcla
 - 15.5 Comparación de resultados
- 16 Solución propuesta
 - 16.1 Método de segmentación elegido
 - 16.2 Mejoras realizadas al método de segmentación

CAPÍTULO SEXTO: Conclusiones

- 17 ¿Porqué el método de segmentación propuesto es mejor que el original?
- 18 Comparación de resultados
- 19 Trabajos futuros

ANEXOS:

Anexo A: Clasificación de Imágenes Médicas

Anexo B: Otras definiciones

Anexo C: Figuras y Tablas

BIBLIOGRAFÍA

A mi familia

CAPÍTULO PRIMERO: Fundamentos teóricos

1 Imágenes digitales

Las imágenes digitales son el principal ingrediente de lo que se conoce como visión por computador y representan con algún tipo de codificación, normalmente en una matriz de números de dos dimensiones, una escena del mundo real.

Tras la captura de la imagen tendremos una matriz de valores en dos dimensiones (2D), es decir, una *imagen digital*. Los valores de la matriz 2D representan niveles de intensidad.

2 Captura de imágenes

La obtención de imágenes digitales requiere dos elementos básicos. El primero es un dispositivo físico que es sensible a una determinada onda del espectro de energía electrón electromagnético (tal como rayos X, ultravioleta, visible, infrarrojo, etc.), y que produce una señal eléctrica de salida proporcional al nivel de energía incidente en cualquier instante de tiempo. En la siguiente figura se muestra el rango de frecuencias del espectro electromagnético.

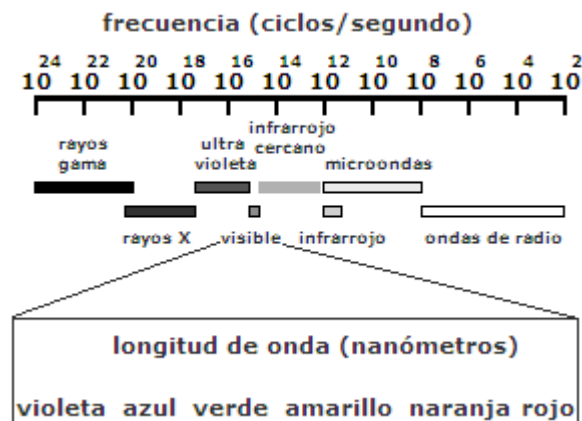


figura 1: espectro electromagnético

El segundo denominado *digitalizador* es un dispositivo para convertir la señal eléctrica continua de salida del dispositivo físico en un conjunto discreto de localizaciones del plano de la imagen y después de la cuantización de dicha muestra.

Esto implica, en primer lugar, determinar el valor de la imagen continua en cada una de las diferentes localizaciones discretas de la imagen (cada valor localizado de forma discreta se

denomina muestra de la imagen), y luego asignar a cada muestra una etiqueta entera discreta, que es representativa del rango en el cual varía la muestra.

Una vez capturada la señal continua, y cuantificada espacialmente y en amplitud, se obtiene una imagen digital, que es como se representa en el computador, es decir, tendremos la matriz (2D) de números enteros. Estos son los valores que se manipulan para extraer información de las imágenes mediante los programas de software.

3 Definiciones

3.1 Muestreo

Es el proceso de medición de los valores de una imagen como una función $f(x,y)$ en un intervalo discreto. Cada muestra corresponde a una pequeña área cuadrada llamada **píxel**. Una imagen digital es un arreglo 2-dimencional de píxeles. Los píxeles son indexados por coordenadas enteras x,y .

3.2 Resolución

Se entiende por resolución al número de puntos (píxeles) que contiene la imagen. Se mide en puntos por pulgada (o ppp) y refleja simplemente el número de puntos utilizados para componer una imagen. Por ejemplo, en una imagen con una resolución de 300 ppp, cada pulgada de la imagen contiene 300 puntos. Dicho de otro modo, cada punto o píxel que compone la imagen ocupa $1/300$ de pulgada. Esto entraña que una imagen de baja resolución (por ejemplo, 50 ppp) puede aparecer moteada, mientras que es probable que una imagen de mayor resolución tenga una apariencia correcta.

Dado que la resolución refleja el número de puntos por pulgada de una imagen, una imagen de alta resolución dará como resultado un archivo de mayor tamaño que la misma imagen a una resolución inferior.

La diferencia entre una imagen en blanco y negro, en escala de grises y en color radica en el número de bits (o valores) asociados a cada punto de la imagen.

Para precisar cuántos colores se mostrarán en una imagen, hay que elevar 2 a una potencia igual al número de bits asociados a la representación interna de la imagen en una computadora, y obtendrá el número de colores posibles en la visualización.

Así, pues:

1. Una imagen en blanco y negro es una imagen de 1 bit. Esto significa que cada punto o píxel de la imagen sólo puede tener dos valores: blanco o negro.
2. Una imagen de escala de grises es de 8 bits, lo que significa que cada punto puede tener un valor entre 256. Por lo tanto, un píxel en una imagen de escala de grises puede ser de color negro, blanco, o de un tono de gris entre 254 valores posibles.
3. Una imagen de 16 colores es una imagen de 4 bits, ya que cada punto puede mostrar un color entre 16 posibles.
4. Una imagen de 256 colores es una imagen de 8 bits, ya que cada punto puede mostrar un color entre 256 posibles.
5. Una imagen de color auténtico o verdadero es una imagen de 24 bits ($3 \times 8 \text{ bits} = 24 \text{ bits}$) compuesta de puntos rojos, verdes y azules, cada uno de los cuales puede mostrar un valor entre 256 posibles. Una imagen digital de color auténtico es lo más parecido a una imagen de calidad fotográfica, ya que puede mostrar un color entre 16,7 millones de tonos posibles.

3.3 Resolución espacial

La resolución espacial de una imagen es el tamaño físico de los píxeles de esa imagen (el área que representa un píxel dentro de la imagen.) Determina qué cantidad de información útil puede extraerse de la imagen.

3.4 Conectividad de píxeles

Se dice que un píxel p , de coordenadas (x,y) tiene cuatro píxeles que establecen con él una relación de vecindad horizontal o vertical, ellos son:

Horizontal: $(x-1,y)$ y $(x+1,y)$

Vertical: $(x,y-1)$ y $(x,y+1)$

Estos cuatro píxeles definen lo que se conoce como entorno de vecindad-4 y se denota $E_4(p)$.

Obviamente existen las excepciones dadas cuando el píxel (x,y) es un punto del borde de la imagen, en cuyo caso algunos de los vecinos definidos anteriormente no existen.

Los cuatro vecinos diagonales de p tienen coordenadas:

$$(x-1,y-1), (x+1,y-1), (x-1,y+1) \text{ y } (x+1,y+1)$$

Y se denotan $E_D(p)$. Estos píxeles junto con los de $E_4(p)$ forman los vecinos y se denotan $E_8(p)$. Como antes, existen las excepciones dadas cuando el píxel (x,y) es un punto del borde de la imagen, en cuyo caso algunos de sus vecinos no existen.

$(x-1,y-1)$	$(x,y-1)$	$(x+1,y-1)$
$(x-1,y)$	(x,y)	$(x+1,y)$
$(x-1,y+1)$	$(x,y+1)$	$(x+1,y+1)$

figura 1: 8 vecinos

Propiedades derivadas de los conceptos de vecindad de píxeles:

a) Conectividad:

Sea V el conjunto de valores de intensidad de los píxeles que se permiten estén adyacentes, por ejemplo, si sólo se desea que exista conectividad entre los píxeles con intensidades de 80, 81 y 83 entonces $V=\{80,81,83\}$. Consideremos tres tipos básicos de conectividad:

Conectividad-4 (también llamado 4-vecinos): Dos píxeles p y q con valores en V se dice que están 4-conectados si q está en el conjunto $E_4(p)$.

Conectividad-8 (también llamado 8-vecinos): Dos píxeles p y q con valores de V están 8-conectados si q está en el conjunto $E_8(p)$.

Conectividad-m (mixta): Dos píxeles p y q con valores en V están m -conectados si q está en el conjunto $E_4(p)$ o q está en $E_D(p)$ y $E_4(p) \cap E_4(q) = \emptyset$.

Un píxel p es contiguo a otro píxel q si están conectados. Se puede definir la adyacencia-4, 8 o m , dependiendo del tipo de conectividad especificada. Dos subconjuntos imagen $S1$ y $S2$ son contiguos si algún píxel de $S1$ es contiguo a algún píxel de $S2$.

b) Camino:

Un camino desde el píxel p con coordenadas (x,y) hasta el píxel q con coordenadas (s,t) es una secuencia de varios píxeles con coordenadas:

$$(x_0,y_0), (x_1,y_1), \dots, (x_n,y_n)$$

donde $(x_0,y_0) = (x,y)$ y $(x_n,y_n) = (s,t)$, (x_i,y_i) son adyacentes a (x_{i-1}, y_{i-1}) con $1 \leq i \leq n$ y n es la longitud del camino. Se pueden definir caminos-4, 8 o m dependiendo del tipo de adyacencia empleada.

Camino simple: es un camino sin píxeles repetidos.

Camino cerrado: es un camino simple en el cual el primer píxel es vecino del último.

c) Región:

Está constituida por un conjunto de píxeles, en el cual hay un camino entre cualquier par de píxeles y todos los píxeles de este camino pertenecen al conjunto.

En caso de haber un camino entre dos píxeles de la imagen, esos píxeles se denominan contiguos. Alternativamente, podemos decir que una región es el conjunto de píxeles donde cada par de píxeles en el conjunto es contiguo. La relación ser contiguo es reflexiva, simétrica y transitiva y por lo tanto definimos una descomposición del conjunto (la imagen) en clases de equivalencia, que serán regiones.

Supongamos que R_j son regiones disjuntas en la imagen que fueron creadas con la relación ser contiguo, y además supongamos, para evitar casos especiales, que esas regiones no tocan los límites de la imagen. Sea la región R la unión de todas las regiones R_j y el conjunto R' el complemento de la región R con respecto a la imagen, el subconjunto de R' que es contiguo con los límites de la imagen se denomina el *fondo* y el resto del conjunto

complementario R' se denomina *huecos*. Si no hay huecos en una región decimos que es *simplemente contigua* y por el contrario, si los hay, decimos que es *múltiplemente contigua*.

d) Frontera:

La frontera interna de una región R es el conjunto de píxeles dentro de la región que tiene uno o más vecinos fuera de R . Esta es una definición intuitiva al considerar la frontera de una región como un conjunto de puntos en su límite.

La *frontera externa* de una región R es la frontera del fondo, o sea, el complemento de la *frontera interna* de la región.

e) Borde:

Es una propiedad local de un píxel y su vecindad inmediata. Para detectar un borde, se recurre a conceptos de variación de la intensidad del píxel, que se cuantifica en forma de un vector gradiente con magnitud y dirección. Existe una diferencia radical entre frontera y borde, la frontera es un concepto global relacionado con la región. El borde, en cambio, expresa propiedades locales de una función de imagen. La frontera y los bordes están interrelacionados.

3.5 Ruido

La imagen puede ser de baja calidad debido a la presencia de ruido o a una mala adquisición de la misma.

3.6 Artefactos

Por problemas en la adquisición podemos encontrar ruido en la imagen, que distorsiona las características de las distintas regiones.

3.7 Volumen parcial

Al ser las imágenes representaciones 2D de objetos 3D podemos encontrar en la imagen información que no es propiamente de ese plano y que limita la definición de las regiones.

3.8 No homogeneidad

Los objetos no poseen siempre una iluminación uniforme, sino que muchas veces es sólo una textura característica lo que los diferencia de otros.

3.9 Forma

A *grosso modo*, podemos clasificar las formas de los objetos en objetos “*papa*”, con una forma más o menos regular y convexa, objetos con forma de “*coliflor*”, con bordes no definidos y ciertas ramificaciones, y objetos “*spaghetti*”, finos y muy ramificados. Podemos tener dificultades para segmentar estos últimos en caso de que se encuentren entrelazados con otros objetos.

4 Color

El uso del color en el procesamiento de imágenes se debe a dos importantes factores. En primer lugar, en análisis de imágenes, el color es un potente descriptor aunque a menudo simplifica la identificación y extracción de objetos de una escena. En segundo lugar, el ojo humano puede distinguir una amplia gama de colores comparado con los niveles de gris.

En el tratamiento de imágenes, su procesamiento en color se divide en dos áreas primordiales: color propiamente dicho y *pseudocolor*. En la primera categoría se procesan las imágenes obtenidas con un sensor de color. Estas imágenes constan básicamente de tres bandas, a saber rojo, verde y azul. En la segunda, las imágenes monocromas, esto es, imágenes de grises, son coloreadas por asignación de un color a determinado nivel de intensidad de gris. El resultado en este caso es una imagen en las tres bandas de color mencionadas anteriormente.

Fundamentos del color

Resulta muy conocido el hecho de que la luz solar que atraviesa un prisma de cristal se descompone en los siete colores básicos del espectro visible. Los colores que las personas perciben en un objeto están determinados por la naturaleza de la luz reflejada por el mismo. Un cuerpo que refleja la luz de todas las longitudes de onda se muestra como blanco al observador. Sin embargo, un cuerpo que favorece la reflectancia en un rango limitado de longitudes de onda en el espectro visible exhibe un determinado color. Por ejemplo, los objetos verdes reflejan la luz con longitudes de onda en el rango 200 a 270 nm y absorben gran cantidad de energía en otras longitudes de onda.

La conformidad de la luz es vital para la ciencia del color. El ojo humano percibe una inmensa gama de colores como mixturas de los tres primarios: rojo (R), verde (G), y azul (B).

Con el fin de llevar a cabo su estandarización, le CIE (Comisión Internacional de l'Eclairage) asignó en 1931 los siguientes valores de longitud de onda a los colores primarios: azul = 435,8 nm, verde = 546,1 nm y rojo = 700 nm. La mezcla de los colores primarios produce los secundarios. Magenta (rojo y azul) cyan (verde y azul) y amarillo rojo y verde). La combinación de los tres colores primarios o secundarios en proporciones iguales produce color blanco. Conviene diferenciar entre los colores primarios de la luz y de los pigmentos o colorantes. Un color primario se define como aquel que absorbe un color primario de la luz y refleja o transmite los otros dos. Por lo tanto, los colores primarios de los pigmentos son amarillo, magenta y cyan, y los secundarios son rojo verde y azul. Una combinación apropiada de los colores primarios de los pigmentos produce color negro.

El brillo, el matiz y la saturación son -generalmente- características utilizadas para distinguir colores entre sí. El brillo incorpora la noción cromática de intensidad. El matiz es un atributo asociado con la longitud de onda dominante en la mezcla de longitudes de onda de la luz. Viene a ser como el color percibido por el observador, cuando se dice de un objeto que es rojo, anaranjado o amarillo, realmente se está especificando su matiz. La saturación se refiere a la pureza relativa o a la cantidad de luz blanca mezclada con un matiz. El grado de saturación es inversamente proporcional a la cantidad de luz blanca añadida.

Cuando el matiz y la saturación se toman en forma conjunta se denominan cromaticidad, por lo tanto un color se puede caracterizar por su brillo y cromaticidad. Las cantidades de rojo, verde y azul requeridas para formar un color particular se denominan colores triestímulos, denominados X, Y y Z respectivamente.

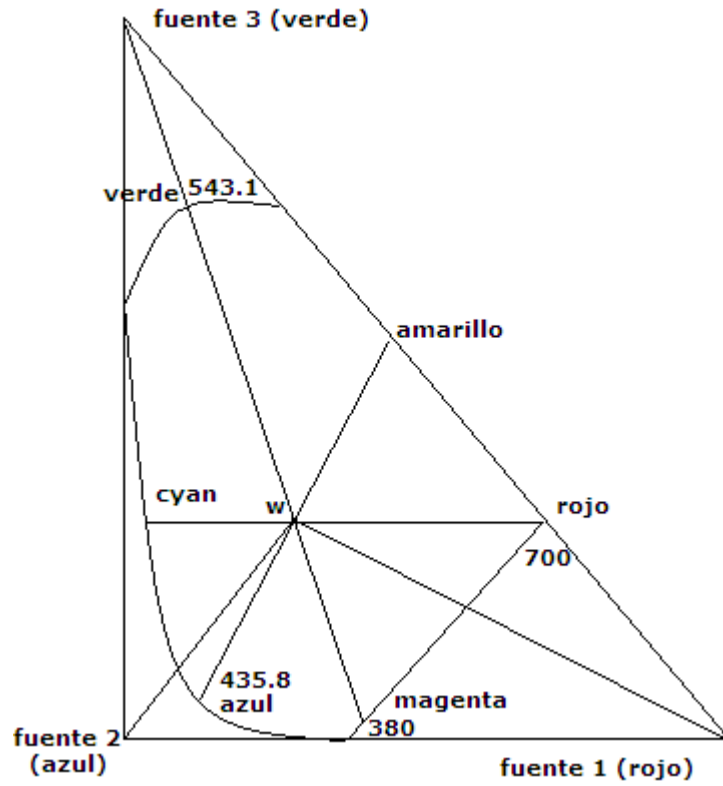


figura 2: diagrama de cromaticidad según el estándar de la CIE. La curva representa los colores visibles saturados al 100%. W representa el color blanco. Los números en la curva representan longitudes de onda.

4.2 Otros modelos de colores

Modelo CMY

Cyan, magenta y amarillo son los colores secundarios de la luz o alternativamente los primarios de los pigmentos. Por ejemplo, cuando una superficie pintada con pigmento amarillo se ilumina con luz blanca, la luz azul es absorbida por la superficie y no se refleja, esto es, el amarillo sustrae la luz azul de la luz blanca. Muchos dispositivos que depositan pigmentos coloreados, como el caso de las impresoras de color y copiadoras, requieren una entrada CMY o realizan una conversión de RGB a CMY de manera interna. Esta conversión se lleva a cabo mediante la siguiente operación:

$$C=1-R$$

$$M=1-G$$

$$Y=1-B$$

La ecuación demuestra que la luz reflejada por una superficie amarilla pura ($Y=1$, $C=0$; $M=0$) no contiene azul, ya que en este caso $B=0$ y $R=1$ y $G=1$. Como corresponde realmente al amarillo. Del mismo modo un pigmento magenta puro no refleja la luz verde y un cyan, la roja. La conversión de RGB a CMY no resulta práctica.

Modelo YIQ

El modelo YIQ se usa en la televisión comercial. Es una combinación de los valores RGB o viceversa. El modelo YIQ fue diseñado para obtener la ventaja del sistema visual humano, que es más sensible a cambios en la reflectancia (intensidad) que a cambios en el matiz o saturación.

La principal ventaja del modelo YIQ en procesamiento de imágenes es que la reflectancia (Y) y la información del color (I y Q) se pueden tratar por separado. La reflectancia es proporcional a la cantidad de luz percibida por el ojo humano. Por lo tanto, la importancia de esta separación es que la componente de reflectancia de una imagen puede ser procesada sin afectar a su contenido de color. Por ejemplo: se puede aplicar una ecualización de histograma a la componente Y sin afectar a las componentes de color.

Modelo HSI

El matiz es un atributo del color que describe su pureza (amarillo puro, rojo, naranja), mientras que la saturación proporciona una medida del grado en el que el color puro es diluido con luz blanca. El modelo de color HSI debe su utilidad a dos hechos fundamentales del color en la imagen; segundo, las componentes de matiz y saturación están íntimamente relacionadas con el modo en que los humanos perciben el color. Estas características hacen del modelo HSI una herramienta ideal para desarrollar algoritmos de procesamiento de imágenes basados en alguna de las sensaciones de color del sistema visual humano.

El modelo puede ser utilizado, por ejemplo, para diseñar sistemas que permitan verificar el grado de madurez de las frutas, o también, la inspección del acabado de color de determinados productos. La idea consiste en emplear las propiedades del color tal cual lo haría la persona encargada de dicha tarea.

Las fórmulas de conversión para pasar de RGB a HSI y a la inversa son considerablemente más complicadas que en los modelos precedentes.

CAPÍTULO SEGUNDO: Representación manipulación y almacenamiento de imágenes

5 Representación de imágenes

Las imágenes se pueden representar, mediante retículas de celdillas a las que vamos asignando valores. Esta es la base de todas las imágenes impresas y de buena parte de las digitales.

Las imágenes digitales en dos dimensiones se realizan creando una retícula de cuatro lados, iguales de dos a dos (ancho y alto, siempre en ese orden

Cada una de las celdillas de dicha retícula se llama "píxel". Un píxel, es un concepto inmaterial que no tiene una medida concreta. No podemos decir si un píxel mide 1 cm. o 1 km. En principio, es solamente una medida de división en celdillas.

De este modo, podemos hablar de una imagen que tenga 200×100 píxeles sin saber que tamaño real y físico tiene. Lo único que sabemos es que la hemos dividido en 20.000 celdas. Sin embargo, cuando le asignemos a esa imagen una resolución, entonces sí sabremos qué tamaño tiene esa imagen. Por ejemplo, si decimos que tiene 100 píxeles por pulgada, querrá decir que cada 2,54 cm. (pues eso es lo que mide una pulgada), habrá 100 celdas, con lo que cada píxel equivaldrá a 2,54 mm. Si dijéramos que esa imagen tiene una resolución de 1 píxel por pulgada, lo que sabríamos es que ahora esa celda tomaría el valor de 2,54 cm.

Todo esto significa que el píxel es sólo una unidad de división sin un tamaño real concreto. Sólo cuando asignamos una resolución a la imagen de la que hablamos estamos dándole un tamaño concreto al píxel.

Hay imágenes de mayor resolución e imágenes de más baja resolución. A mayor resolución, mayor nitidez del dibujo y mejor se reflejan los detalles. Sin embargo, hay que tener presente que cualquier resolución que supere la que el dispositivo de salida (pantalla, impresora, etc.) es capaz de representar no hace más que sobrecargar el sistema y hacer lento el trabajo.

Una forma muy importante de clasificar las imágenes de mapa de bits es según la cantidad y tipo de información que se asigne a cada píxel.

5.1 1 bit por píxel (2 tonos)

En este tipo de imágenes cada celda (píxel) sólo puede tener uno de dos valores: Uno o cero. Como basta 1 bit para definir esa alternativa, se las llama "imágenes de 1 bit" (también se les llama "imágenes de mapa de bits, de alto contraste, o imágenes de línea".)

5.2 8 bits por píxel

Habitualmente un valor simple es asociado a cada píxel de la imagen. Es normal emplear 8 bits para la cuantización, en cuyo caso, el valor de cada píxel puede guardarse usando un arreglo en memoria.

Cada píxel puede tener 256 valores diferentes (las 256 posibilidades combinatorias de un byte u octeto). Este es el modo de las imágenes digitales de blanco y negro "normales". En ellas sólo se distinguen hasta 256 tonos diferentes de gris.

5.3 15 bits por píxel

En algunos modos de vídeo se utiliza una variación en la cual se emplean 15 bits para el color, lo que significa 5 bits para cada color. En este caso la diferencia no es notoria.

5.4 16 bits por píxel

El color a 16 bits utiliza 2 bytes para almacenar los colores, partiendo los 16 bits en 5 bits para el azul, 5 bits para el rojo y 6 bits para el verde, ya que el ojo humano requiere de más tonos verdes, lo cual se interpreta en colores como 32 tonos de azul y rojo y 64 para el verde, lo cual resulta en una pequeña pérdida de calidad de la imagen.

La representación con 16 bits permite 65,536 (2¹⁶) valores distintos.

5.5 24 bits por píxel

Si tomamos un píxel y le asignamos 3 bytes, dispondremos de 24 bits en tres grupos de ocho, podemos "colorearlo" siguiendo el sistema de color de los monitores de televisión, que se basan en tres "canales" de luz de color (Rojo, Azul y Verde). De este modo podemos distinguir hasta 16.777.216 tonos de color (256 Rojo × 256 Azul × 256 Verde). En realidad, lo que estamos haciendo es superponer tres canales de luz, uno rojo, otro verde y otro azul, cada uno con 256 posibilidades de tono.

5.6 32 bits

Si a cada píxel le asignamos 4 bytes, podríamos representar (teóricamente), los valores CMYK propios de la cuatricromía profesional (1 byte para el cian, otro para el magenta, otro para el amarillo y un cuarto para el negro). Esto es "teóricamente" porque la representación del color que la pantalla de un ordenador puede hacer es mediante imágenes RGB, ya que ese es el modo de reproducir el color de los monitores.

En la siguiente tabla se muestran las distintas opciones de representación

Profundidad de color	Colores posibles	Comentarios
1 bit por píxel	2	Arte Lineal (B&N). Modo Mapa de Bits
4 bits por píxel	16	Modo Escala de Grises
8 bits por píxel	256	Modo Escala de Grises. Modo Color Indexado. Cantidad estándar de colores que admiten los formatos GIF y PNG-8, así como muchas aplicaciones multimedia.
16 bits por píxel	65.536	High Color
24 bits por píxel	16.777.216	True Color. Modo RGB 8 bits por canal (8x3=24). Modo Lab 8 bits por canal
32 bits por píxel	4.294.967.296	Modo CMYK

Tabla 1: distintas representaciones de color

6 Manipulación de imágenes digitales

Cabe mencionar la diferencia entre *análisis de imágenes digitales* y transformación de *imágenes digitales*. El proceso puede ser visto como una transformación de una imagen en otra, es decir, a partir de una imagen se obtiene otra imagen modificada. Podemos decir que el análisis es una transformación de una imagen en algo que no es una imagen, En consecuencia, el análisis es un determinado tipo de información representando una descripción o decisión. En cualquier caso, las técnicas de análisis de imágenes digitales son aplicadas a imágenes que han sido procesadas previamente. Desde el punto de vista de un observador humano, el análisis de imágenes es una tarea fácil y rápida, algo que no ocurren en visión artificial. Igualmente la capacidad de percepción humana permite procesar rápidamente una imagen para detectar en ella características de interés, por ejemplo: bordes o regiones.

No obstante, desde el punto de vista de los sistemas de visión artificial, el único propósito del procesamiento de imágenes es hacer el análisis posterior más simple y más fiable. Por consiguiente, el procesamiento de imágenes debe facilitar la extracción de información para un posterior análisis. En esta fase son típicas las transformaciones basadas en la aplicación de una función sobre el valor del nivel de gris de los píxeles.

Cabe distinguir como operaciones de vecindad, las encaminadas a eliminar el ruido subyacente de la imagen conocidas como suavizado y las de extracción de bordes. Existen otras operaciones que no se pueden clasificar como individuales ni como de vecindad, ya que transforman la imagen por otros procedimientos. Son transformaciones que operan globalmente sobre los valores de intensidad de la imagen cuyo efecto es un realzado de la imagen original.

Procesamiento básico de imágenes

El procesamiento de datos en el sistema de visión puede enfocarse desde dos perspectivas:

- 1) Alteración píxel a píxel de los datos es una escala global (individuales).
- 2) Operaciones basadas en múltiples puntos (vecindad).

La generación de un nuevo píxel en una nueva imagen será función bien del valor de cada píxel en su localización individual, o bien de los valores de los píxeles en la vecindad de un píxel dado.

Operaciones individuales

Las operaciones individuales implican la generación de una nueva imagen modificando el valor del píxel en una simple localización basándose en una regla global aplicada a cada localización de la imagen original. El proceso consiste en obtener el valor del píxel de una localización dada en la imagen, modificándolo por una operación lineal o no lineal y colocando el valor del nuevo píxel en la correspondiente localización de la nueva imagen. El proceso se repite para todas y cada una de las localizaciones de los píxeles de la imagen original.

Función inverso o negativo

Esta función crea una imagen nueva de salida que es la imagen inversa de la original. Esta función es de suma utilidad en diversas aplicaciones tales como imágenes médicas. Para una imagen con valores de gris en el rango que va de 0 a 255 la función de transformación es:

$$q = 255 - p$$

Función umbral

Esta clase de transformaciones crean una nueva imagen de salida binaria a partir de un margen de grises. Donde el nivel de transición está dado por el parámetro de entrada p_1 , la función de transformación es la siguiente:

$$q = \begin{cases} 0 & \text{para } p \leq p_1 \\ 255 & \text{para } p > p_1 \end{cases}$$

Función intervalo de umbral binario

Esta clase de transformación crea una imagen de salida binaria a partir de una imagen en escala de grises, donde los valores de gris cuyo nivel esté en el intervalo definido por p_1 y p_2 son transformados a 0 y todos los valores fuera de ese intervalo son transformados a 255. La función de transformación es la siguiente:

$$q = \begin{cases} 255 & \text{para } p \leq p_1 \text{ o } p \geq p_2 \\ 0 & \text{para } p_1 < p < p_2 \end{cases}$$

Función intervalo de umbral binario invertido

Esta clase de transformación crea una imagen de salida binaria a partir de una imagen de grises, donde todos los valores de gris cuyos niveles están en el intervalo definido por p_1 y p_2 son transformados a 255 y todos los valores fuera de ese intervalo son transformados a 0. La función de transformación es la siguiente:

$$q = \begin{cases} 0 & \text{para } p \leq p_1 \text{ o } p \geq p_2 \\ 255 & \text{para } p_1 < p < p_2 \end{cases}$$

Función umbral de escala de grises

Esta clase de transformación crea una imagen de salida con los únicos valores de nivel de gris comprendidos en el intervalo dado por p_1 y p_2 y el resto 255. La función de transformación es la siguiente:

$$q = \begin{cases} 255 & \text{para } p \leq p_1 \text{ o } p \geq p_2 \\ 0 & \text{para } p_1 < p < p_2 \end{cases}$$

Función umbral de escala de grises invertido

Esta clase de transformación crea una imagen de salida con los únicos valores de nivel de gris invertidos comprendidos en el intervalo definido por p_1 y p_2 y el resto a 255. La función de transformación es la siguiente:

$$q = \begin{cases} 255 & \text{para } p \leq p_1 \text{ o } p \geq p_2 \\ 0 & \text{para } p_1 < p < p_2 \end{cases}$$

Función de extensión

Esta clase de operadores proporciona una imagen de salida con la escala de grises completa correspondiente al intervalo de entrada definido por p_1 y p_2 y suprime todos los valores fuera de ese rango. La función de transformación es la siguiente:

$$q = \begin{cases} 0 & \text{para } p \leq p_1 \text{ o } p \geq p_2 \\ (p - p_1) \frac{255}{p_2 - p_1} & \text{para } p_1 < p < p_2 \end{cases}$$

Función de reducción de nivel de gris

Esta clase de operadores proporciona una imagen de salida con un menor número de niveles de gris respecto de la imagen original de entrada. La imagen de entrada es reducida a $n+1$ niveles de gris con la siguiente transformación:

$$q = \begin{cases} 0 & \text{para } p \leq p_1 \\ q_1 & \text{para } p_1 < p \leq p_2 \\ q_2 & \text{para } p_2 < p \leq p_3 \\ \dots & \\ q_n & \text{para } p_{n-1} < p \leq 255 \end{cases}$$

7 Medios de almacenamiento

Los medios de almacenamiento para imágenes pueden dividirse en los siguientes tres grupos:

- 1) corto plazo**
- 2) en línea**
- 3) archivo**

El almacenamiento a corto plazo está basado en memoria, empleando la memoria RAM de la computadora.

El almacenamiento en memoria suele ser más rápido, pero es volátil: la imagen se pierde una vez que la computadora se apaga. También, el almacenamiento en memoria, implica un apreciable costo.

El almacenamiento en línea es no volátil y permite accesos a la información de la imagen de forma relativamente rápida. Es típicamente provisto por el disco duro de una computadora. El almacenamiento basado en disco suele ser lento y económico, mucho más que el almacenamiento en memoria.

Los medios de archivo son piezas de hardware dedicadas a proveer espacio de almacenamiento para una o más imágenes así como herramientas para el rápido acceso a las mismas desde una computadora cliente. Suelen proveer herramientas tales como zoom instantáneo (similar a pasar una lupa sobre una foto). Se emplea para archivar información de imágenes a largo plazo. Normalmente utiliza medios removibles. El acceso a la información es lento en comparación con el acceso a la información contenida en un disco duro local o el acceso a memoria volátil. La ventaja suele ser la posibilidad de compartir la información fácilmente.

Los medios removibles empleados suelen ser: CDs, pen drives, disquetes.

8 Formatos de archivo

Cuando se almacenan imágenes en medios en línea o de archivo, es importante seleccionar el formato de archivo correcto. Esto determinará no sólo cómo la información de las

imágenes es guardada, sino también qué tipo de información adicional se guardará junto con cada píxel de la imagen. Muchos formatos de archivo consisten de un segmento cabecera y segmentos de datos. El encabezado contiene, entre otras cosas, el alto y el ancho de la imagen. Porque es imposible mostrar la imagen o procesarla sin conocimiento de sus dimensiones. El encabezado de la mayoría de los formatos de archivo de imágenes contiene una firma o números mágicos que es una secuencia corta de bytes que representan o identifican ese formato.

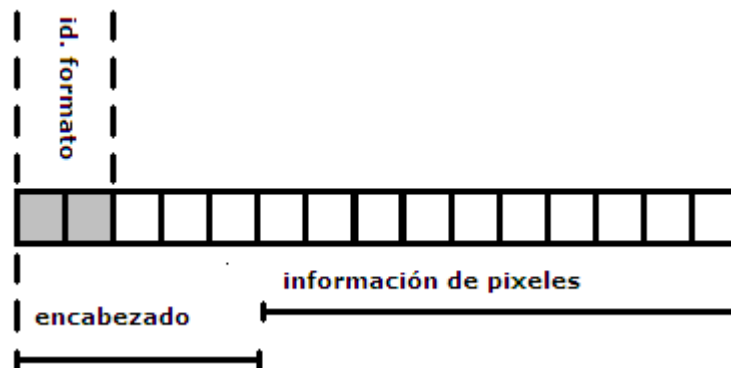


figura 4: estructura básica de un archivo de imagen

Los formatos de archivo de imágenes se pueden agrupar en tres categorías:

1) Orientados a dispositivo

Fueron desarrollados para ser usados con piezas de hardware específicas. La estructura del archivo de imagen puede ser elegida para facilitar la carga rápida en cierto tipo de computadora. Las desventajas de este formato incluyen la falta de portabilidad, la ineficacia cuando son usadas con otro hardware y la tendencia a cambiar el formato cuando el hardware cambia.

2) Orientados a software

Son diseñados por una compañía de software para ser usados con un programa particular o con un tipo de programas particulares. Ejemplos de este tipo de formatos incluyen los archivos PCX y los BMP encontrados en máquinas Macintosh y Windows respectivamente. Las ventajas y desventajas de este formato son similares a las del formato orientado a dispositivo.

3) Formatos de intercambio

Son diseñados para facilitar el intercambio de información de imágenes entre usuarios. Por medio de medios removibles o redes de computadoras. Es esencial que sea posible utilizarlos en el mayor rango de hardware y software posibles. Suelen emplear compresión, que reduce el tamaño físico de los archivos y facilita el intercambio reduciendo tiempos de transmisión. Ejemplos de formatos de intercambio son:

GIF: Graphic Interchange Format

PNG: Portable Network Graphics

JFIF: JPEG File Interchange Format

TIFF: Tagged Image File Format

PGM: Portable Grey Map

FITS: Flexible Image Transport System

Los formatos GIF y JFIF predominan en el ámbito Internet. El formato PNG se desarrolló para reemplazar el GIF por problemas legales de patente del formato GIF.

Los formatos de intercambio suelen permitir almacenar una cantidad arbitraria de datos definidos por el usuario junto a la información de la imagen. Esto permite el procesamiento de imágenes en las más variadas situaciones. En astronomía, por ejemplo, es necesario conocer las circunstancias exactas en las cuales se adquirieron los datos de las imágenes para un análisis posterior. El formato FITS se emplea en el ámbito astronómico para garantizar imágenes con un alto volumen de información extra en su encabezado.

Los formatos de archivos simples contienen encabezados de tamaño fijo. Las piezas de información necesarias para procesar la imagen están contenidas en posiciones conocidas a priori dentro del encabezado.

Los formatos etiquetados no requieren almacenar información en posiciones fijas de la imagen, en vez de ello hacen uso de cadenas de caracteres especiales que codifican segmentos de datos particulares. Ejemplos de formatos de archivo etiquetados es FITS, el cual emplea caracteres ASCII para formar las cadenas de etiquetas. PNG emplea secuencias de 4 bytes de caracteres mayúsculas y minúsculas, TIFF utiliza códigos numéricos.

8.1 BMP

El formato Windows Bitmap (BMP) es un formato pensado principalmente para plataformas Windows y posee una cabecera que especifica el ancho y el alto de la imagen. Aunque permite compresión, se usa normalmente sin compresión. Los datos se guardan de abajo hacia arriba y los píxeles en el orden Azul/Verde/Rojo. Cada píxel está representado por 24 bits, que corresponden a 8 bits por cada una de las ya conocidas tres bandas R (rojo), G (verde), B (azul). Lo habitual es extraer la información de los ficheros BMP en forma de las tres bandas de colores mencionadas. También soporta la existencia de una sola banda en formato de grises con píxeles de tamaño de 8 bits.

Es útil para imágenes ráster. No obstante, dado que se suele usar sin compresión su tamaño es grande para un intercambio vía e-mail.

Los ficheros que contienen este tipo de datos poseen la extensión `.bmp`. No obstante, existen ficheros con extensión `.bmp` que sólo contiene una banda cuyo rango de valores varía entre el 0 y 255 para imágenes de 8 bits. Cada valor tiene asociado un color dentro de una paleta de colores de forma similar a como se hace en el formato GIF.

8.2 JPG

El formato *Joint Photographic Experts Group (JPEG)* es uno de los formatos más utilizados para intercambio de información gráfica vía Internet. Esto es así porque se trata de un formato comprimido, precisamente usando la compresión de su mismo nombre. Las razones de compresión son elevadas y en consecuencia, el tamaño de los datos se reduce considerablemente. Utiliza un método de compresión con pérdidas basado en la transformada del coseno.

Sobre la imagen original se aplica un algoritmo en el que interviene la mencionada transformada del coseno.

El algoritmo JPEG permite elegir la calidad de la imagen comprimida en términos de porcentaje.

Los ficheros que contienen este tipo de datos tienen la extensión `.jpg` o `.jpeg`. También soporta imágenes en escala de grises e imágenes basadas en una paleta de colores.

Dado que se pueden obtener razones de compresión altas, este formato resulta apropiado para intercambios por e-mail o Internet. Resulta apropiado para imágenes ráster. Sin embargo, hay que tener ciertas precauciones con las razones de compresión ya que la imagen puede deteriorarse considerablemente como ocurre en las imágenes que contiene bordes nítidos.

8.3 TIFF

El formato *Tag Interchange File Format (TIFF)* es un formato basado en etiquetas (tags en inglés). La cabecera de los ficheros TIFF contiene una serie de etiquetas, que informan de los datos relativos a la imagen, por ejemplo, número de líneas, columnas, número de bits por píxel, etc. Una de dichas etiquetas resulta de especial interés, concretamente la que se refiere el tipo de compresión utilizada. Así TIFF admite, entre otras, compresión sin pérdidas (LZW) o con pérdidas (JPEG).

Las etiquetas tienen asignado un valor numérico y los números con valores iguales o superiores a 32768 son privados, esto significa que se usan para que un determinado usuario inserte en ellas datos de su interés. Esto puede suponer un problema a la hora de descifrar los datos en formato TIFF ya que existen programas que no pueden interpretar dichas etiquetas privadas y por lo tanto el formato no es correctamente interpretado. Otros informan de la existencia de dichas etiquetas y de que serán ignoradas y en la mayoría de los casos, sencillamente las ignoran sin informar de su existencia. Las etiquetas por debajo de aquel valor corresponden a las definidas por la especificación TIFF. Como dato de interés, un archivo en formato TIFF puede contener más de una imagen.

Como en el caso del formato BMP, TIFF también soporta imágenes de grises e imágenes basadas en una paleta de colores.

Cuando la compresión utilizada es sin pérdidas, el tamaño de los archivos es relativamente grande, por lo que no es apropiado para intercambios a través de email o Internet. Resulta apropiado para imágenes ráster.

Los ficheros que contienen datos de este tipo poseen la extensión .tif o .tiff.

8.4 GIF

El formato *Graphic Interchange Format (GIF)* está limitado a un número reducido de colores, por ejemplo, 256, que se conoce como mapa o paleta de colores. Esto hace que el formato GIF sea poco útil para representaciones de datos ráster, donde la variedad de colores es significativa. Es el formato por excelencia para imágenes vectoriales tales como mapas, dibujos, etc. El formato GIF utiliza también compresión LZW.

Los 256 colores de la paleta de colores en un formato GIF pueden ser cualesquiera de los 16.7 (256 al cubo) millones de colores posibles que se pueden derivar. Cada color está representado por tres valores (R,G,B) y cada fichero GIF contiene su propia paleta de colores. La

imagen contiene valores en el rango de 0 a 255 y cada uno de esos valores de la imagen representa el índice para extraer el color correspondiente de la paleta. Así, un valor de imagen de 123 extrae la información de color de la posición 123 de la paleta. Si se cambia la paleta de colores, es evidente que se cambia también el contenido de color de la imagen. Existe una versión del formato GIF que admite la posibilidad de representar una secuencia de imágenes dentro de un simple GIF, dando lugar a lo que denomina GIF animado.

Los ficheros que contienen este tipo de datos tienen la extensión *.gif*.

8.5 PNG

El formato PNG (Portable Network Graphics) es un formato gráfico basado en un algoritmo de compresión sin pérdida para mapas de bits libre de patentes. Este formato fue desarrollado en buena parte para solventar las deficiencias del formato GIF y permite almacenar imágenes con una mayor profundidad de color.

El formato PNG es un formato de archivo de imagen de prestaciones muy usado para el diseño web. Se trata de un formato estándar que acepta una profundidad de color de hasta 24 bits (frente a los 8 bits del formato GIF), y nos brinda la posibilidad de utilizar un efecto transparente de gran calidad gracias al canal alfa. Mediante este efecto se consigue una transparencia realmente buena, frente a la capacidad limitada del extendido formato GIF que se limita a marcar como transparente un solo color de la paleta de colores, con lo cual no es posible crear semitransparencias como lo hace el formato PNG.

Las imágenes PNG usan la extensión *.png* y han obtenido un tipo MIME (*image/png*) aprobado el 14 de octubre de 1996.

9 Tratamiento de imágenes digitales

Una imagen digital puede considerarse como una matriz de números. Considerándolo de esta forma, podemos realizar las mismas operaciones que se efectúan con las matrices numéricas, con las debidas limitaciones derivadas del hecho de la cuantización de los niveles de gris, es decir, una imagen sólo va a tomar valores en el rango preestablecido. Como ya sabemos, generalmente en el intervalo [0..255]. Esas operaciones se explicarán a continuación y son: *suma, resta, multiplicación, división, producto, etc.*

Además, dado que la representación de los datos en una computadora finalmente llega a ser una representación binaria, siempre es posible realizar cualquier tipo de operación lógica binaria sobre ellos, dando lugar a las denominadas operaciones lógicas elementales: *and*, *or*, *xor*, *not*, *etc.*

Operaciones aritméticas

Vistas las imágenes como matrices de valores numéricos, cabe la posibilidad de llevar a cabo sobre ellas diferentes y diversas operaciones de naturaleza aritmética. Estas operaciones aritméticas pueden considerarse un determinado tipo de transformación.

En efecto, se trata de transformaciones que utilizan la información contenida en la misma localización (posición de los píxeles) de dos imágenes de entrada A y B para crear una nueva imagen C. La extensión a más de dos imágenes es trivial. La dimensión de las imágenes es la misma. La función de la transformación f_d puede ser lineal o no. Esta función se aplica a todos los pares de píxeles en las imágenes de entrada, esto es, la información en una localización de píxel de una imagen se combina con la información de la correspondiente localización de píxel de la imagen resultante. La función característica está dada por la siguiente ecuación:

$$C_{x,y} = f_D(a_{x,y}, b_{x,y})$$

donde f_D es una función de dos variables y los subíndices x e y varían de 0 a M y N , respectivamente (que son las dimensiones de las imágenes). La función f_d puede ser adición, sustracción, multiplicación, división, exponenciación, máximo, o cualquier otra función que opere con dos valores numéricos. La función deberá tener un factor de escala apropiado para k para mantener los valores de salida dentro del rango de niveles de gris o de intensidad adecuado, así como para evitar desbordamientos y valores negativos.

La transformación implica dos variables asociadas con los pares de píxeles:

$$R(x,y) = f_d(a(x,y), b(x,y))$$

donde a y b son las matrices de entrada, f_d es el operador funcional y R es la matriz resultante de salida. Este proceso se presenta en la figura siguiente donde aparecen dos imágenes de entrada de la misma dimensión y la localización de píxel, que es la misma en ambas imágenes. Los niveles

de intensidad correspondientes a sendas localizaciones espaciales son los parámetros de entrada a la función de transformación. Una vez aplicada la transformación, se obtiene un nuevo valor en la misma localización espacial de las imágenes de entrada. De esta forma, tras aplicar la función de transformación a todos y cada uno de los píxeles de las imágenes de entrada, se obtiene una imagen resultante que representa la transformación global llevada a cabo sobre toda la imagen.

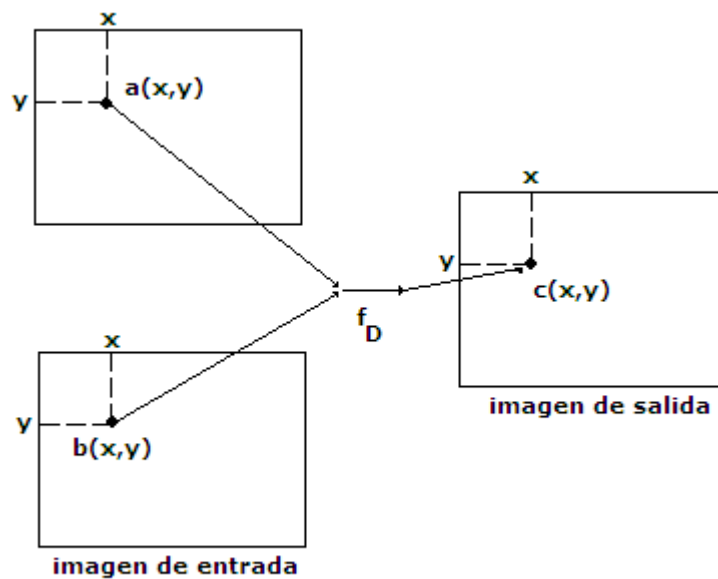


figura 5: representación de la transformación de dos imágenes en una

Cuando las operaciones se aplican sobre imágenes en color, en cuyo caso cada imagen consta de más de una banda, dichas operaciones se llevan a cabo banda a banda. Así, dadas dos imágenes en color A y B en el formato RGB, las imágenes de entrada que aparecen en la figura anterior serían separadas para cada una de las bandas. En efecto, las imágenes de entrada son en este caso A y B. En el caso de las imágenes en color, la i toma los valores R, G o B dependiendo de la banda en cuestión. El resultado sería también una imagen multibanda, de forma que se obtiene un resultado para cada una de las bandas. Combinados convenientemente, estos resultados para cada una de las bandas, se obtiene la imagen resultante final. En el caso que nos ocupa de una imagen RGB, obtendríamos tres imágenes resultantes C_i , una por cada banda R, G o B. La combinación C_r , C_g y C_b daría lugar a la correspondiente imagen resultante final RGB en color C.

Adición de imágenes

La adición puede utilizarse para reducir los efectos del ruido en la imagen . El valor de salida está dado por la fórmula siguiente:

$$c_{x,y}=(a_{x,y} + b_{x,y})/k$$

donde k es dos para el caso de las dos imágenes de entrada. Si la suma abarcase más de dos imágenes, k tomaría el valor del número de imágenes usadas. Los valores de salida finales deben redondearse por defecto o por exceso.

Una aplicación importante de la adición de imágenes es precisamente la conversión del modelo RGB a YIQ, de forma que la imagen de intensidad de este último modelo se obtiene mediante la adición de las tres bandas R, G, B que componen la imagen original RGB. En este proceso de transformación también está implicada una operación de división de imágenes por cociente de valor 3.

Sustracción de imágenes

Esta es una técnica útil para detectar el cambio producido en dos imágenes que han sido captadas en dos instantes de tiempo distintos. Los datos de ambas imágenes también pueden representar pérdidas de calor o frío si la fuente de datos es el espectro infrarrojo. Puesto que en la representación de imágenes se emplean números positivos, es necesario definir la salida de alguna forma que haga todos los valores positivos. Esto implica necesariamente un reescalado donde el valor más negativo se corresponde con el 0 y el máximo con 255 para la escala de grises del 0 al 255.

La relación está dada por la fórmula:

$$c_{x,y}=k(a_{x,y} - b_{x,y})$$

donde k es una función no lineal de forma que el valor mínimo que toma $c_{x,y}$ es 0 y el máximo es 255.

Como hemos expresado anteriormente, los valores de las imágenes de entrada varían en el rango de 0 a 255. Por ello, es posible que a la hora de realizar la sustracción de imágenes aparezcan diferencias significativas entre ambas con valores tanto positivos como negativos. Imaginemos que en un momento dado, $a_{x,y}=200$ y $b_{x,y}=20$ con $k=1$ el resultado sería finalmente $c_{x,y}=180$. Si se invierten los valores de forma que $a_{x,y}=20$ y $b_{x,y}=200$ entonces ahora resultaría $c_{x,y}=-180$. En ambos casos, la diferencia es igual de significativa, por consiguiente, en numerosas aplicaciones una forma de apreciar estas diferencias consiste en elegir el valor absoluto de la diferencia como el valor resultante de salida.

Multiplicación y división de imágenes

En el contexto del tratamiento de imágenes, estas operaciones no tienen mucho sentido desde el punto de vista de operaciones matriciales, ya que multiplicar una imagen por otra carece de significado práctico.

Sin embargo, lo que sí tiene sentido dentro de este contexto es la multiplicación / división de una imagen por un determinado factor. Esta operación tiene como finalidad corregir de alguna forma la imagen original.

También pueden aplicarse correcciones locales, es decir, variar el factor de corrección dependiendo del contenido de la imagen en una determinada región de ella. Por ejemplo: puede usarse el concepto de varianza local, calculada en la región de interés, de forma que, en función de su valor, se aplique la corrección correspondiente. Bien multiplicando, o bien dividiendo por un factor dado.

Detección de movimiento

Puede resultar útil para aplicaciones tales como el campo de la vigilancia estática. En efecto, la simple sustracción de imágenes adquiridas en diferentes instantes de tiempo puede detectar la presencia del movimiento, suponiendo que la cámara está estacionaria y que la iluminación es constante. Una imagen de diferencias $d(i,d)$ es una imagen binaria donde los valores de 0 representan áreas de la imagen con movimiento, es decir, áreas donde existe una diferencia sustancial entre los niveles de intensidad de las imágenes consecutivas $f_1(i,j,t_1)$ y $f_2(i,j,t_2)$

$$d(i, j) = \begin{cases} 0 & \text{si } |f_1(i, j, t_1) - f_2(i, j, t_2)| \leq \varepsilon \\ 1 & \text{caso contrario} \end{cases}$$

donde ε es un umbral predeterminado. La imagen de diferencias puede basarse en características más complejas de las imágenes, tales como nivel medio de intensidad en alguna vecindad, texturas locales, etc. Es claro que con esta aproximación se puede detectar el movimiento de cualquier objeto distinto de su fondo (entendiendo por detección de movimiento su verificación sin realizar medida alguna).

En la siguiente figura hemos ilustrado los conceptos expuestos. En (a) se muestra una imagen sintética de referencia tomada en el instante t_1 que contiene un objeto de intensidad constante moviéndose sobre una superficie de fondo, también con intensidad constante. En (b) aparece el mismo objeto captado en el instante t_2 . El movimiento es de translación (aproximación) hacia la cámara. En (c) se representa la imagen de diferencias calculada mediante la ecuación anterior.

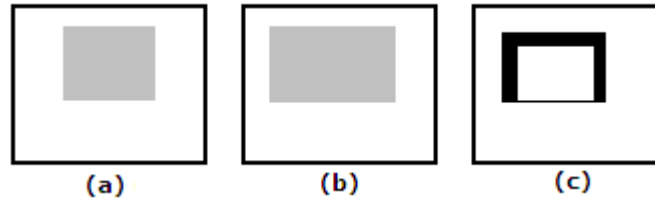


figura 6: (a) imagen original sintética inicial; (b) imagen original sintética en un instante posterior, donde el cuadrado se ha acercado a la cámara; (c) diferencia de las dos imágenes. El movimiento se detecta por el contorno negro

Sean $f_1(i, j, t_1)$ y $f_2(i, j, t_2)$ dos imágenes consecutivas separadas por un intervalo de tiempo, un elemento $d(i, j)$ de la diferencia de imágenes entre f_1 y f_2 puede tener un valor 0, debido a alguna de las siguientes razones:

- 1) $f_1(i, j)$ es un píxel de un objeto en movimiento y $f_2(i, j)$ es un píxel estático del fondo o viceversa.
- 2) $f_1(i, j)$ es un píxel de un objeto en movimiento y $f_2(i, j)$ es un píxel de otro objeto en movimiento

- 3) $f_1(i,j)$ es un píxel de un objeto en movimiento y $f_2(i,j)$ es un píxel de una parte diferente del mismo objeto en movimiento.
- 4) Ruido o imprecisiones en el posicionamiento de la cámara estacionaria.

Las trayectorias detectadas utilizando imágenes de diferencias no revelan cuál es la dirección del movimiento. Si se necesita conocer la dirección, se puede construir una *imagen de diferencias acumulativa*. Esta imagen de diferencias acumulativa contiene información sobre la dirección del movimiento y otras propiedades del movimiento relacionadas con el tiempo. La imagen de diferencias acumulativa D_{acum} se construye a partir de una secuencia de n imágenes, considerando la primera imagen f_1 como la de referencia. Los valores de la imagen de diferencias acumulativa reflejan cómo de rápido varían los niveles de gris con respecto a la imagen de referencia:

$$D_{\text{acum}}(i,j) = \sum a_k |f_1(i,j) - f_k(i,j)|, k=1..n$$

donde a_k proporciona la importancia de las imágenes en la secuencia de las n imágenes, a las imágenes más recientes se les puede dar mayores pesos para reflejar la importancia del movimiento actual y especificar la localización del objeto en un instante determinado.

Operaciones lógicas

Es posible efectuar sobre los datos de las imágenes una serie de operaciones lógicas derivadas de la representación binaria de dichos datos. Para ilustrar este concepto, veamos un sencillo ejemplo: consideremos dos niveles de intensidad $a_{x,y}=2000$ y $b_{x,y}=157$, que en representación binaria resultan los siguientes valores $a_{x,y}=11001000$ y $b_{x,y}=10011101$. Si sobre ellos realizamos una simple operación lógica *and*, el resultado sería un nuevo valor binario obtenido por la aplicación de dicha operación bit a bit.

Es decir, $c_{x,y}=a_{x,y} \text{ and } b_{x,y}=100010000$ lo que en representación decimal resulta ser $c_{x,y}=136$

En la imagen de la figura (a) y (b) aparecen dos imágenes originales correspondientes a radiografías del tórax, en (c) y (d) se muestra el resultado de las operaciones lógicas *and* y *or*.

Por otro lado, y gracias al proceso de binarización, es posible disponer de imágenes binarias lógicas a la entrada para su posterior manipulación mediante las correspondientes operaciones lógicas. Esto es un caso especial del anterior, como puede deducirse fácilmente sin más que considerar el cero lógico como el valor binario 00000000 y el uno lógico como 11111111.

Por lo tanto, con ambos tipos de imágenes es posible realizar toda clase de operaciones lógicas, entre ellas *negación*, *or*, *and*, *xor*. También, dadas dos imágenes, es posible realizar sobre ellas otro tipo de operaciones relacionadas tales como $<$, $>$, \leq , \geq . Por ejemplo: dadas las imágenes A y B , una nueva imagen $C = A \leq B$ se obtiene realizando la comparación dada píxel a píxel y obteniendo el valor de 1 para C en aquellos píxeles donde se cumple la relación y 0, donde no se cumple.

Operaciones morfológicas

La morfología matemática está basada en la geometría y la forma. Las operaciones morfológicas simplifican las imágenes y preservan las formas principales de los objetos.

En visión artificial se usa la morfología para el tratamiento de regiones en el sentido de determinar cómo se pueden cambiar, contar o evaluar.

La morfología puede emplearse para las siguientes tareas:

- Suavizar los bordes de una región. Esto es útil, por ejemplo, si se necesita mejorar un borde, ya que, usando técnicas de segmentación estándar, los bordes se presentan generalmente ruidosos debido tanto al proceso de captura como a los procesos de tratamiento apareciendo sobre ellos pequeños valles o protuberancias, que pueden suprimirse mediante transformaciones morfológicas.
- Separar determinadas regiones que el proceso de segmentación las presenta unidas.
- Unir regiones que han sido separadas durante la segmentación.
- Como consecuencia de los dos puntos anteriores, facilitar el cómputo de regiones en una imagen.

Las operaciones matemáticas tienen su principal aplicación en imágenes binarias y los fundamentos matemáticos fueron concebidos desde el punto de vista de la posición de cada píxel binario en la imagen en lugar de hacerlo desde el punto de vista de la intensidad.

Transformaciones geométricas básicas

Para el análisis de las escenas representadas en las imágenes, se requiere poder investigar más específicamente un área dentro de dicha imagen, llamada *región de interés*. Para lograr esto, necesitamos operaciones geométricas que modifiquen las coordenadas espaciales de la imagen. Indiscutiblemente, el principal objetivo de una operación geométrica es la transformación de los valores de una imagen, tal y cómo podría observarse desde otro punto de vista. De este modo, las operaciones de magnificar o reducir una imagen no son otra cosa que aproximar o alejar el punto de vista: rotarla equivale a girar el punto de observación y trasladarla es hacer lo propio con dicho punto.

El paso previo a toda operación geométrica comienza por observar la distribución espacial de los píxeles en la imagen original y en la transformada. Generalmente, en la imagen original asumirán una estructura matricial (tal y como están ordenados en la memoria de la computadora). Esta disposición podría asemejarse a la que aparece en la siguiente figura:

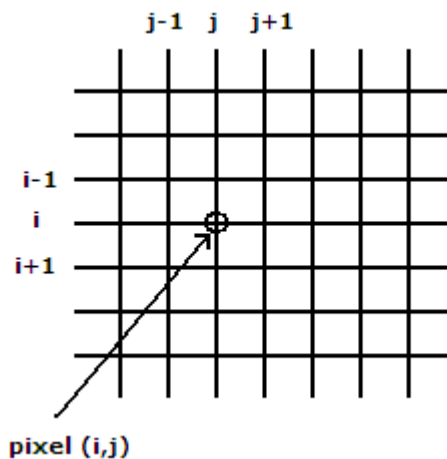


figura 7: disposición en forma de rejilla de los píxeles de una imagen

Dado que la imagen digital es discreta, no existen valores de intensidad entre los valores discretos de las coordenadas i y j , que coinciden con las intersecciones de los valores discretos tanto horizontales como verticales. Al transformar esta rejilla según un desplazamiento, un giro,

un acercamiento con cualquier otra transformación geométrica, los nuevos píxeles ya no tienen por qué quedar situados sobre tales intersecciones y caerán en general sobre puntos intermedios de ellos. Al tener que proyectar estos píxeles sobre los de la imagen final, que deben asumir una estructura similar a la de la imagen original, es necesario pasar de las coordenadas *pseudo discretas* a las discretas definitivas y, en concreto, es preciso calcular los valores de los píxeles finales en función de los transformados.

Observemos algunos conceptos expuestos mediante un sencillo e ilustrativo ejemplo. En la siguiente figura se muestra una transformación geométrica, que en este caso concreto resulta ser una rotación. La rejilla continua muestra la disposición convencional de píxeles de la imagen original, mientras que la rejilla discontinua ilustra cómo queda la rejilla original después de girarla un ángulo dado alrededor de un punto determinado, es decir, después de rotarla.

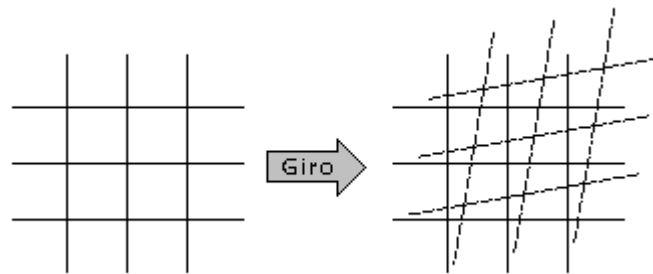


figura 8: ante cualquier transformación geométrica, los píxeles de la rejilla transformada (línea punteada) no tienen por qué coincidir con los de la rejilla destino o final

Por consiguiente, para llegar al resultado final es preciso:

- 1) Determinar las coordenadas de cada píxel (i,j) perteneciente a la rejilla original en la rejilla transformada (en líneas discontinuas). En general, estos nuevos píxeles (i',j') obtenidos tras la transformación no serán valores enteros y, por lo tanto, no coincidirán con píxeles de la rejilla destino
- 2) Calcular los valores de los píxeles (x,y) finales en la rejilla destino a partir de los valores conocidos de píxeles (i',j')

El primer paso depende de la transformación a realizar, es decir, de la función de transformación que se aplique en cada momento, mientras, que el segundo se corresponde con una operación de *interpolación*. Tras la transformación geométrica, es preciso obtener los valores de intensidad asociados con dicha transformación de forma que la imagen original aparezca transformada geométricamente pero con los valores de intensidad obtenidos de la imagen original.

Primero veremos el concepto de interpolación junto con el procedimiento de calcular los valores de intensidad asociados a una determinada transformación geométrica y posteriormente abordaremos y estudiaremos diferentes tipos de funciones de transformación.

INTERPOLACIÓN

La interpolación puede considerarse como el cálculo del valor de intensidad de un píxel en una posición cualquiera como una función de los píxeles que lo rodean (y que en nuestro caso ocuparán las posiciones enteras de la rejilla destino.)

Supongamos que queremos calcular el valor del píxel de coordenadas (x,y) dado en la figura siguiente, en función de los valores de los píxeles de la rejilla (i,j) .

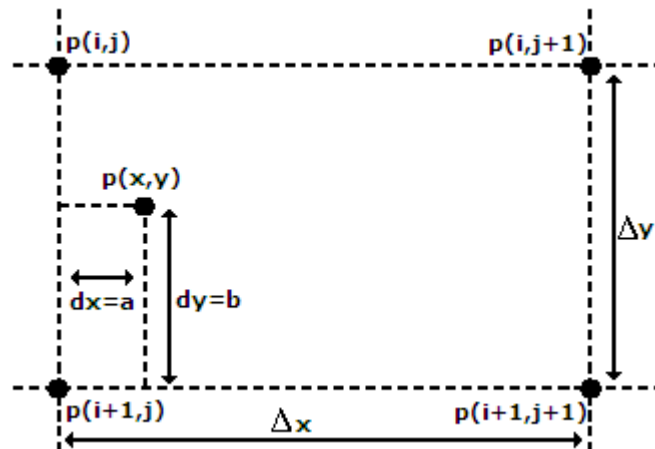


figura 9: esquema gráfico de la interpolación

La formalización matemática de la interpolación puede entenderse como sigue:

$$p(x, y) = \sum_{i=-n}^n \sum_{j=-m}^m p(i, j) h(x - i, y - j)$$

donde $p(i,j)$ es el valor de un píxel de la imagen original en la posición espacial (i,j) , $p(x,y)$ es el valor del píxel imagen resultante de la interpolación a partir de $p(i,j)$ y $h(x,y)$ es lo que se conoce como núcleo de interpolación.

Existen diferentes núcleos de interpolación, de entre los cuales los más utilizados en el tratamiento geométrico de imágenes son: vecino más próximo, bilineal y bicúbico.

Interpolación por vecino más próximo

Una forma de hacerlo es suponer que el píxel, para ser interpolado, toma el mismo valor que el más cercano de entre los cuatro que lo rodean; de aquí el nombre de vecino más próximo. Para decidir cuál es el más cercano, se puede utilizar la distancia euclídea. En la *figura 9* tomaría el valor $p(i,j)$. Otra posibilidad es asignarle la intensidad media asociada a los dos píxeles más cercanos, uno de la dirección horizontal x y el otro en la dirección vertical y . Con referencia a la misma figura tomaría el valor medio entre $p(i,j)$ y $p(i+1,j)$.

El núcleo de interpolación $h(x,y)$ estaría definido como sigue:

$$h(x) = \begin{cases} 1 & \text{si } 0 < |x| < 0.5 \\ 0 & \text{caso contrario} \end{cases}$$

Interpolación bilineal

Esta es otra forma de interpolar con mejores resultados pero, por el contrario, con mayor costo computacional. La interpolación bilineal asigna al píxel en cuestión un valor medio ponderado de las intensidades de los cuatro píxeles que lo rodean. Los factores de ponderación vienen dados por la distancia entre el píxel y los del entorno. Los factores de ponderación se calculan de la forma siguiente en relación con la *figura 9*.

$$a_1 = \left(\frac{1 - dx}{\Delta x} \right) \left(\frac{1 - dy}{\Delta y} \right)$$

$$a_2 = \frac{dx}{\Delta x} \left(\frac{1 - dy}{\Delta y} \right)$$

$$a_3 = \left(\frac{1-dx}{\Delta x} \right) \frac{dy}{\Delta y}$$

$$a_4 = \frac{dx}{\Delta x} \frac{dy}{\Delta y}$$

donde $0 \leq dx \leq 1, 0 \leq dy \leq 1, \Delta x = 1, \Delta y = 1$, por lo tanto obtenemos:

$$a_1 = (1-dx)(1-dy)$$

$$a_2 = dx(1-dy)$$

$$a_3 = (1-dx)dy$$

$$a_4 = dxdy$$

Finalmente, el valor de intensidad del píxel interpolado, en función de los cuatro valores de intensidad de los píxeles de su entorno, queda:

$$p(x, y) = a_1 p(i, j) + a_2 p(i, j + 1) + a_3 p(i + 1, j) + a_4 p(i + 1, j + 1)$$

En este caso el núcleo de interpolación $h(x,y)$ estaría definido como sigue:

$$h(x, y) = \begin{cases} 1 - |x| & \text{si } 0 < |x| < 1 \\ 0 & \text{caso contrario} \end{cases}$$

Interpolación bicúbica

Una tercera forma de interpolar con mejores resultados incluso que la interpolación bilineal es la interpolación bicúbica, en la que intervienen 16 puntos vecinos del píxel que se está interpolando.

La función de interpolación viene definida por la siguiente expresión:

$$h(x) = \begin{cases} 1 - 2|x|^2 + |x|^3 & \text{si } 0 < |x| < 1 \\ 4 - 8|x| + 5|x|^2 - |x|^3 & \text{si } 1 < |x| < 2 \\ 0 & \text{de otro modo} \end{cases}$$

Transformaciones básicas

La morfología matemática aprovecha las propiedades de los conjuntos de puntos, los resultados de la geometría integral y la topología. La premisa inicial consiste en suponer que las imágenes reales pueden ser modeladas utilizando conjuntos de puntos de cualquier dimensión (por ejemplo, el espacio euclídeo N-dimencional). El espacio euclídeo 2D (E2) y su sistema de subconjuntos es un dominio natural para la descripción de formas planas. Desde la perspectiva de conjuntos se consideran las operaciones habituales en ellos, entre las que se incluyen: *inclusión*, *unión*, *intersección*, *complementario* o *conjunto vacío*. La diferencia de conjuntos se define por $X-Y=X \text{ intersección } Y \text{ complemento}$

El tratamiento de las imágenes por computadora emplea el equivalente digital del espacio euclídeo, conjuntos cuyos elementos son pares de números enteros en el contexto de las imágenes binarias.

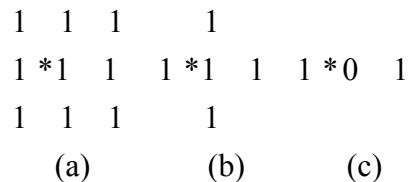
Un punto se representa por un par de enteros que corresponden a las coordenadas de la imagen digital. Por lo tanto, una imagen binaria puede tratarse como un conjunto de puntos @D. Los puntos que pertenecen a los objetos en la imagen representados un conjunto X. En imágenes binarias, esos puntos son píxeles con valor binario uno. Los puntos del conjunto complementario de X corresponden al fondo con valores binarios iguales a cero. Fijándonos en la imagen de la siguiente figura, el origen marcado con un punto tiene coordenadas (0,0) y las coordenadas de cualquier punto se interpretan como posición en la fila, posición en la columna con respecto al origen. Además en dicha figura los puntos que pertenecen al objeto se marcan con 1. Cualquier punto de una imagen discreta X puede considerarse un vector con respecto al origen (0,0).

$$\begin{bmatrix} *0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

ejemplo de conjunto de puntos

En este caso el conjunto de puntos X considerando exclusivamente los valores lógicos 1 vendría dado por $X=\{(0,1),(0,2),(0,3),(1,1),(1,2),(1,3),(2,2),(2,3),(3,2),(3,3)\}$

Una transformación morfológica f viene dada por la relación de la imagen (conjunto de puntos X) con otro pequeño conjunto de puntos B llamado elemento estructural. B se expresa con respecto a un origen local O (llamado punto representativo o elemento director). Algunos elementos estructurales típicos (obviamente también desde el punto de vista binario) se muestran la siguiente figura en la cual el punto representativo se marca con un *. El elemento estructural de la figura (a) tiene la siguiente expresión en forma de conjunto de puntos: $B = \{(-1,-1),(-1,0),(-1,1),(0,-1),(0,0),(0,1),(1,-1),(1,0),(1,1)\}$ mientras que la expresión para el correspondiente de la figura (b) es: $B = \{(-1,0),(0,-1),(0,0),(0,1),(1,0)\}$ La figura (c) ilustra la posibilidad de que el punto O no sea un miembro del elemento estructural B , cuya representación en forma de conjunto sería $B = \{(0,-1),(0,1)\}$; obsérvese que no aparece el punto de coordenadas $(0,0)$ al tener un valor binario cero y por consiguiente, no considerarse un miembro del conjunto estructural B .



elementos estructurales típicos

La transformación morfológica $f(X)$ aplicada a la imagen X significa que el elemento estructural B se desplaza por toda la imagen. Suponiendo que B se posiciona sobre algún punto de la imagen, el píxel de la imagen correspondiente al punto representativo O de B se denomina píxel actual. El resultado de la relación (que puede ser cero o uno) entre la imagen X y el elemento estructural B en la posición actual se guarda en el píxel actual de la imagen.

Dilatación

La transformación morfológica de la dilatación Θ combina dos conjuntos utilizando la adición de vectores. La dilatación $X \Theta B$ es el conjunto de puntos de todas las posibles adiciones vectoriales de pares de elementos, uno de cada conjunto X y B .

$$X \Theta B = \{d \in E^2 : d = x + b \text{ para cada } x \in X \text{ y } b \in B\}$$

A continuación, un ejemplo de dilatación:

$$X = \{(0,1),(1,2),(2,0),(2,1),(3,0), (3,1)\}$$

$$B = \{(0,0),(0,1)\}$$

$$X \oplus B = \{(0,1),(1,2),(2,0),(2,1),(3,0),(3,1),(0,2),(1,3),(2,2),(3,2)\}$$

$$\begin{bmatrix} *0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \oplus \begin{bmatrix} *1 & 1 \end{bmatrix} = \begin{bmatrix} *0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

figura 10: dilatación

Erosión

La transformación morfológica de la erosión \ominus combina dos conjuntos usando la sustracción de vectores. Es dual de la dilatación. Ni la erosión ni la dilatación son transformaciones invertibles:

$$X \ominus B = \{d \in E^2 : d + b \in X \text{ para cada } b \in B\}$$

Esta expresión dice que cada punto d del conjunto X , que para nosotros será la imagen, es comprobado; el resultado de la erosión está dado por los puntos d para los cuales todos los posibles $d + b$ están en X . La figura siguiente muestra un ejemplo del conjunto de puntos X erosionados por el elemento estructural B :

$$X = \{(0,2),(1,2),(2,0),(2,1),(2,2),(2,3),(3,2),(4,2)\}$$

$$B = \{(0,0),(0,1)\}$$

$$X \ominus B = \{(2,0),(2,1),(2,2)\}$$

Apertura y cierre

La erosión y dilatación son transformaciones no invertibles. Si una imagen es erosionada y luego dilatada, la imagen original no se recupera. En efecto, el resultado es una imagen simplificada y menos detallada de la imagen original.

La erosión seguida de una dilatación crea una transformación morfológica importante llamada *apertura*. La apertura de una imagen X por el elemento estructural B se denota por $X \circ B$ y se define como:

$$X \circ B = \{X \oplus B\} \ominus B$$

La dilatación seguida de una erosión crea una transformación morfológica llamada *cierre*. El cierre de una imagen X por un elemento estructural B se denota por

$$X \cdot B = \{X \ominus B\} \oplus B$$

Si una imagen X permanece invariable por apertura con respecto al elemento estructural B , se dice que es *abierto* con respecto a B . Análogamente, si una imagen X permanece invariable por cierre con respecto al elemento estructural B , se dice que es *cerrada* con respecto a B .

La apertura y el cierre con un elemento estructural isótropo se emplea para eliminar detalles específicos de la imagen más pequeños que el elemento estructural. La forma global de los objetos no se distorsiona. El cierre conecta objetos que están próximos entre sí, rellena pequeños huecos y suaviza el contorno del objeto rellenando los pequeños valles mientras que la apertura produce el efecto contrario. Los conceptos de pequeño y próximo están relacionados con la forma del elemento estructural.

9.1 Manipulación de colores

Modelos de color:

El propósito de un modelo de color es facilitar la especificación de los colores de alguna forma estándar. En esencia, un modelo de color es una especificación de un sistema de

coordenadas 3-D y un subespacio dentro de dicho sistema donde cada color se representa por un punto.

Muchos modelos de color en uso hoy en día están orientados hacia el hardware (monitores en color e impresoras) o bien hacia aplicaciones donde la manipulación del color es un objetivo (creación de gráficos en color para animación). Los modelos más comunes orientados al hardware usados en la práctica son el RGB (rojo, verde, azul) para monitores en color y una amplia gama de videocámaras; el CMY (cyan, magenta, amarillo), para impresoras en color y el YIQ, para TV en color.

En el modelo YIQ la coordenada Y corresponde a la reflectancia o intensidad, I y Q son dos componentes cromáticas que se denominan infase y cuadratura respectivamente. En procesamiento de imágenes se emplean RGB, YIQ, HSV (matiz, saturación, valor), HSI (matiz, saturación, intensidad).

Aunque el modelo CMY se usa para impresión, más que para tratamiento de imágenes, también se considera aquí para proporcionar una visión de conjunto sobre modelos de color.

9.2 Conversiones entre representaciones

Conversión de RGB a HSI

El modelo RGB está definido con respecto a un tetraedro unitario. No obstante, las componentes de color del modelo HSI (matiz y saturación) están definidas con respecto al triángulo de color del modelo mostrado en la figura siguiente.

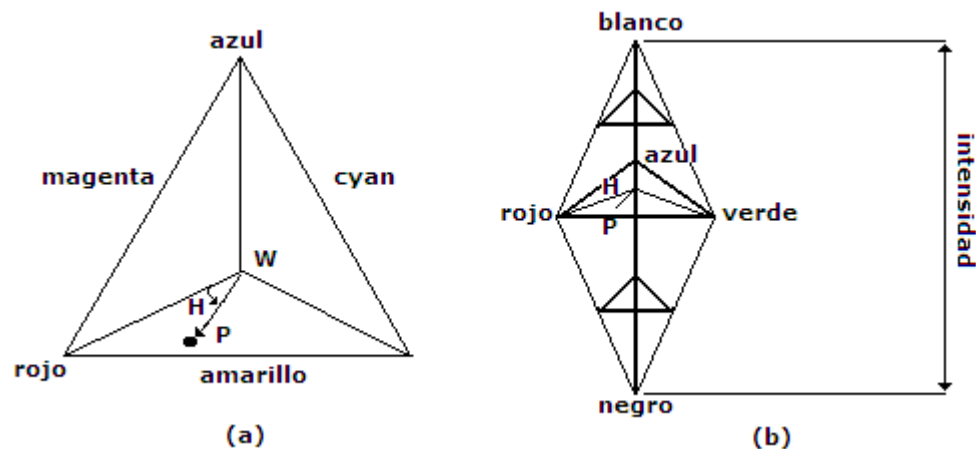


figura 11: (a) triángulo de color HSI; (b) color e intensidad en el modelo HSI

Esta figura (a) es un triángulo en el plano, de forma que no existe perspectiva alguna y todos los puntos en él son coplanarios. Además el punto W es punto de intersección de sus medianas. Todos los colores que se obtienen por combinación de tres colores dados caen dentro de un triángulo cuyos vértices están definidos por los tres colores iniciales. En la figura anterior (b) se observa que el matiz (H) de un punto de color P es el ángulo del vector mostrado con respecto al eje rojo. Por lo tanto, cuando $H=0^\circ$ el color es rojo, mientras que cuando $H=60^\circ$, el color es amarillo. Con $H=120^\circ$ el color es verde y así sucesivamente.

La saturación S del punto de color P es el grado de dilución del color con blanco y es proporcional a la distancia de P al centro del triángulo. A medida que P se aleja de W y se aproxima cualquiera de los lados del triángulo, mayor es el grado de saturación del color.

La intensidad en el modelo HSI se mide con respecto a una línea perpendicular al triángulo y pasando a través de su centro. Las intensidades a lo largo de esta línea por debajo del triángulo progresan hacia el negro y por encima, hacia el blanco.

Combinando matiz, saturación e intensidad en un espacio de color 3-D se obtiene la representación piramidal de la figura anterior (b), donde cada triángulo marcado con líneas más gruesas corresponde a un determinado nivel de intensidad. Cualquier punto en la superficie de la estructura piramidal representa un color saturado puro, ya que son puntos de los lados de los triángulos. El matiz de dicho color se determina por su ángulo con respecto al eje rojo y su intensidad, por su distancia perpendicular desde el punto correspondiente al negro, esto es, cuanto mayor es la distancia desde el negro, mayor la intensidad del color. Todo lo anterior es aplicable a puntos dentro de la estructura, la única diferencia es que los colores se hacen menos saturados a medida que se aproximan al eje vertical que une blanco con negro.

Los colores en el modelo HSI se definen con respecto a los valores normalizados de rojo, verde y azul, dados en términos de los colores primarios RGB por:

$$r = \frac{R}{R + G + B} ; g = \frac{G}{R + G + B} ; b = \frac{B}{R + G + B}$$

donde:

$$r + g + b = 1 \quad (A)$$

Mientras que R, G y B pueden todos ellos tomar el valor 1 simultáneamente, las variables normalizadas deben satisfacer la ecuación (A). En efecto, ésta es la ecuación del plano que contiene el triángulo HSI.

Para cualesquiera tres componentes de color R, G y B, cada una en el rango [0,1] la componente de intensidad en el modelo HSI se define como:

$$I = \frac{1}{3}(R + G + B)$$

Cuyos valores están en [0,1]. Estos valores pueden escalarse dependiendo de la resolución en número de bits del sensor.

El siguiente paso consiste en obtener H y S. En ambos casos es preciso recurrir a construcciones geométricas un tanto complejas en el triángulo definido anteriormente. Finalmente se llega a las siguientes expresiones para H y S en términos de R, G y B:

$$H = \cos^{-1} \left[\frac{1/2[(R - G) + (R - B)]}{[(R - G)^2 + (R - B)(G - B)]^{1/2}} \right] \quad (B)$$

$$S = 1 - \frac{3}{R + G + B} [\min(R, G, B)] \quad (C)$$

Es verdadera para cualquier punto en el triángulo HSI.

Algunas veces la ecuación para el matiz se expresa en términos de la tangente usando la identidad trigonométrica $\cos^{-1}(x) = 90^\circ - \tan^{-1}(x/\sqrt{1-x^2})$. La fórmula anterior es más simple matemáticamente hablando y además es más eficiente en términos de implementación de hardware.

Por lo tanto, los valores de HSI se obtienen con las ecuaciones (A), (B) y (C) haciendo $H=360^\circ-H$ si $B/I > G/I$. Para normalizar el matiz en el rango [0,1], se hace $H=H/360^\circ$. Finalmente si $S=0$ el matiz no está definido y la saturación está indefinida si $I=0$.

Conversión de HSI a RGB

Para valores de S en $[0,1]$, se trata de encontrar la correspondencia de los valores RGB en el mismo rango. Comenzamos transformando el valor de H por $H=360^\circ(H)$, es decir, multiplicando H por 360° , que devuelve el matiz al rango $[0^\circ,360^\circ]$.

Para el sector RG ($0^\circ < H \leq 240^\circ$), se obtiene:

$$\begin{aligned}H &= H - 120^\circ \\r &= 1/3(1 - S) \\g &= 1/3 \left[1 + \frac{S \cos H}{\cos(60^\circ - H)} \right] \\b &= 1 - (r + b)\end{aligned}$$

Las componentes del color recién obtenidas son normalizadas en el sentido de la ecuación (A) del punto anterior. Las componentes RGB se obtienen a partir de las ecuaciones (A), (B) y (C) del punto anterior, teniendo en cuenta las igualdades $R=3Ir$, $G=3Ig$ y $B=3Ib$.

Para el sector GB ($120^\circ < H \leq 240^\circ$) se obtiene:

$$\begin{aligned}H &= H - 120^\circ \\r &= 1/3(1 - S) \\g &= 1/3 \left[1 + \frac{S \cos H}{\cos(60^\circ - H)} \right] \\b &= 1 - (r + g)\end{aligned}$$

Los valores R , G y B se obtienen a partir de r , g y b de la forma descrita anteriormente. Para el sector BR ($240^\circ < H \leq 360^\circ$),

$$\begin{aligned}H &= H - 240^\circ \\g &= 1/3(1 - S) \\b &= 1/3 \left[1 + \frac{S \cos H}{\cos(60^\circ - H)} \right] \\r &= 1 - (g + b)\end{aligned}$$

Los valores R , G y B se obtienen a partir de r , g y b de la forma descrita anteriormente.

Pseudocolor y conversión de escala de grises a color

Pseudocolor

Es una técnica que consiste en asignar un color a imágenes monocromas basándonos en varias propiedades del contenido de nivel de gris de la imagen original.

Sin embargo, cabe también la posibilidad de asignar nuevos colores a una imagen de color mediante esta técnica. Esta operación se realiza transformando la imagen de color RGB al modelo YIQ o al HSI y aplicando la técnica del pseudocolor sobre la imagen de intensidad Y o I, respectivamente, según el modelo.

Conversión del nivel de gris a color

La idea subyacente consiste en realizar tres transformaciones independientes del nivel de gris en colores separados de rojo, verde y azul para cualquier píxel de entrada. Este método produce una imagen compuesta cuyo contenido del color es modulado por la naturaleza de las funciones de transformación. Este método se muestra en la figura a continuación.

Un conjunto de funciones de transformación es el siguiente:

$$R(x, y) = \cos^2(\alpha f(x, y) + \Pi / 2)$$

$$G(x, y) = \cos^2(\alpha f(x, y))$$

$$B(x, y) = \text{sen}^2(\alpha f(x, y))$$

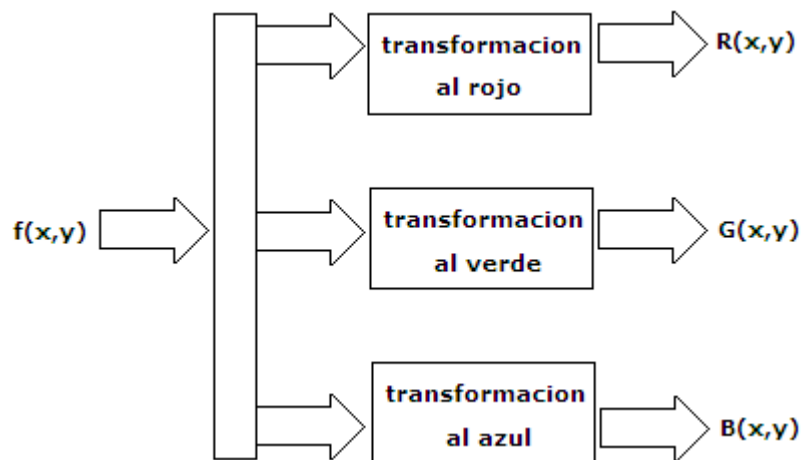


figura 12: diagrama de bloques para la transformación de una imagen de grises a pseudocolor. $R(x,y)$, $G(x,y)$ y $B(x,y)$ son las salidas de las respectivas componentes RGB de color.

9.3 Escala de grises

Una escala de grises es una escala empleada en las imágenes digitales en las que el valor de cada píxel posee un valor equivalente a una graduación de gris. Las imágenes representadas de esta forma están compuestas por sombras de grises, que van desde el negro más profundo variando gradualmente en intensidad de grises hasta llegar al blanco.

Cuando se convierte o se toma una foto en escala de grises se puede representar un conjunto de colores en un tono de gris, o incluso poner cada color en una intensidad. Las escalas de grises son diferentes de las fotografías en blanco y negro, en las que los colores se codifican en blanco o en negro, la escala de grises ofrece una gama de tonalidades de gris entre ambos.

Las imágenes en escala de grises, emplean 8 bits para representar cada píxel lo que sólo permite una escala con 256 intensidades (o escalas de gris), o sea, 2 valores posibles para cada bit (0 y 1) elevado a 8 bits que se emplean para representar cada píxel, nos da 256 tonos de color diferentes que pueden representarse en una imagen en escala de grises

9.4 Cálculo del histograma de una imagen

El histograma de una imagen es una función discreta que representa el número de píxeles de la imagen en función de los niveles de intensidad, g.

La probabilidad $P(g)$ de ocurrencia de un determinado nivel g se define como:

$$P(g) = \frac{N(g)}{M}$$

Donde M es el número de píxeles en la imagen y $N(g)$ es el número de píxeles en el nivel de intensidad g.

Como con cualquier distribución de probabilidad, todos los valores de $P(g)$ son menores o iguales que 1 y la suma de todos los valores de $P(g)$ da como resultado 1.

Consideremos a continuación un sencillo ejemplo ilustrativo, así, dada la imagen de la figura (a) con 10 niveles de gris o intensidad variando en el rango de valores de 0 a 9, el

histograma obtenido a partir de los niveles de intensidad de dicha imagen se muestra en la figura (b).

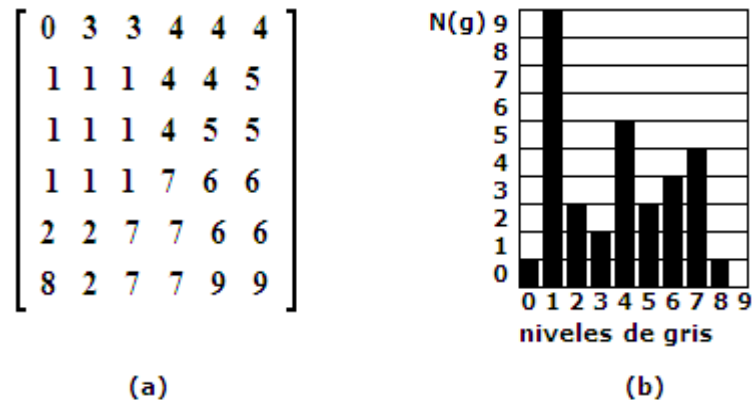


figura 13: representación de histograma

Propiedades estadísticas del histograma de una imagen

A continuación se introduce un conjunto de propiedades que nos informan sobre la distribución de los niveles de gris en la imagen basándonos en su histograma.

Media: es el valor medio de los niveles de gris y nos informa sobre el brillo general de la imagen: esta definida por la siguiente formula:

$$\bar{g} = \sum_{g=0}^{L-1} gP(g) = \frac{\sum_i \sum_j I(i, j)}{M}$$

L es el numero total de niveles de gris, así, para una imagen con valores de gris entre 0 y 255, L seria 256. Una imagen brillante tendrá una medida alta mientras que por el contrario una imagen oscura poseerá una media baja.

Varianza: Mide la disposición de los alrededores de la media, esta definida por la siguiente formula:

$$\sigma^2 = \sum_{g=0}^{L-1} (g - \bar{g})^2 P(g)$$

Una varianza alta corresponde a una imagen con contraste alto, y viceversa.

Asimetría: mide la propiedad de su mismo nombre. Tiene como referencia la media en la distribución de niveles de gris, esta definida por la formula:

$$a = \sum_{g=0}^{L-1} (g - \bar{g})^3 P(g)$$

Un valor absoluto alto de a indica una gran asimetría y un valor bajo, una asimetría pequeña.

Energía: nos informa sobre la distribución de los niveles de gris:

$$E = \sum_{g=0}^{L-1} (P(g))^2$$

La energía tiene valor máximo 1 para una imagen con un único nivel de gris y disminuye a medida que aumenta el número de niveles de gris.

Entropía: también nos informa sobre la distribución de los niveles de gris

$$e = - \sum_{g=0}^{L-1} P(g) \log_2(P(g))$$

Cuanto mayor es el número de niveles de gris en la imagen, mayor es la entropía. Esta medida tiende a variar inversamente con la energía.

9.5 Ecuación del histograma

Una de las técnicas más usadas para la mejora del contraste de la imagen original es la conocida indistintamente como *ecualización* o *igualación de histograma*. La base del realzado es también una manipulación o modificación del histograma, que se consigue mediante las técnicas del mismo nombre. El efecto del realzado se traduce en un aumento del contraste de la imagen, y, en consecuencia, una mejora de la calidad de la imagen. Lo cual se aprecia porque los objetos en la imagen se manifiestan con mayor nitidez.

El objetivo es encontrar una función $F(g)$ que realce el contraste general en la imagen original expandiendo la distribución de los n niveles de gris. Dicha expansión debe ser lo más suave posible en el sentido de que idealmente debería haber el mismo número de píxeles en cada nivel de intensidad, es decir, el objetivo es distribuir los niveles de intensidad de gris. Se puede deducir la función $F(g)$ mediante la simple inspección del histograma original, pero es deseable una función analítica.

Dado que el número de píxeles en una imagen de dimensión $N \times M$ es precisamente este producto y el número de niveles de gris sobre el cual se va a realizar la expansión es N_g , un histograma ideal sería plano, con el mismo número de píxeles en cada nivel de gris, es decir:

$$\text{número ideal de píxeles en cada nivel de intensidad} \frac{N \times M}{N_g}$$

Por otra parte, a partir del histograma podemos definir la función de densidad de probabilidad como sigue:

1) Suponiendo una imagen de 256 niveles de gris en el rango 0 a 255, se cumple:

$$\sum_{g=0}^{g=255} N(g) = N \times M$$

2) La probabilidad por cada nivel de gris g viene dada por:

$$p(g) = \frac{N(g)}{N \times M}; g=0,1,\dots,255$$

se traduce fácilmente que $\sum_{g=0}^{g=255} p(g) = 1$

3) La función de densidad de probabilidad resulta ser:

$$P_x(x) = \sum_{g=0}^x p(g)$$

4) Se trata de realizar una transformación entre funciones de densidad de probabilidad $P_x(x)$ y $P_y(y)$; si se impone la condición de que la función de transformación sea monótona creciente, Para cada valor de x e y se cumple:

$$\int_0^{F(g)} P_y(y)dy = \int_0^g P_x(x)dx = \sum_{g=0}^x p(g)$$

Ésta es la expresión general que se emplea para generar las ecualizaciones.

CAPÍTULO TERCERO: Segmentación de Imágenes en general

10 Introducción

Las unidades de las imágenes son los píxeles. Las únicas propiedades de un píxel son su posición y su nivel o niveles de intensidad, según que la imagen sea de grises o de color, respectivamente.

En las imágenes aparecen ciertas áreas o zonas caracterizadas por el hecho de que constituyen agrupaciones de píxeles conexos entre sí, pero además de la conexión, dichos píxeles presentan propiedades o características comunes, por ejemplo: tienen el mismo color. Estas agrupaciones se llaman regiones.

El desarrollo de algoritmos que segmentan imágenes surgió como consecuencia de la necesidad de contar con mecanismos automatizados de interpretación y análisis de imágenes.

La segmentación involucra dividir una imagen en regiones homogéneas con significado. De forma tal que todos los píxeles en cada región posean las mismas propiedades. Dichas propiedades incluyen: nivel de gris, contraste, textura, etc. El resultado de la segmentación es un conjunto de regiones homogéneas, identificadas de forma unívoca con una etiqueta. Podríamos definir -en consecuencia- a una imagen como un conjunto de regiones homogéneas conectadas y no solapadas entre sí, de forma tal que cada píxel de la imagen pueda ser etiquetado de forma unívoca con la etiqueta de la región a la cual pertenece.

Tras la segmentación de una imagen, los objetos de interés que aparecen en ella se someten a operaciones de procesamiento tales como clasificación de objetos y descripción de escena.

La segmentación de una imagen R involucra la identificación de un conjunto finito de regiones $\{R_1, R_2, \dots, R_n\}$ tales que:

- 1 $R = R_1 \cup R_2 \dots R_n$
- 2 $R_i \cap R_j = \Phi, \forall i \neq j$
- 3 $P(R_i) = \text{verdadero}, \forall i$
- 4 $P(R_i \cup R_j) = \text{falso}, i \neq j$

Puede existir un gran número de particiones, pero la selección de un conjunto apropiado de regiones depende de la correcta elección del predicado P . También se ha de tener en cuenta el

tipo de conectividad entre píxeles empleada. Los algoritmos de segmentación se basan en dos propiedades básicas de los niveles de gris: *discontinuidad* y *similitud entre píxeles*.

Los algoritmos de segmentación basados en discontinuidad particionan la imagen teniendo en cuenta abruptos cambios en los niveles de gris. De forma tal de poder detectar bordes y líneas en la imagen para luego extraer los bordes y unirlos, finalmente la región queda descrita por un borde de contorno que la rodea.

Los algoritmos de segmentación que se basan en similitud entre píxeles tienen en cuenta el parecido entre los píxeles.

Las mejores técnicas de segmentación son:

- Umbralización basada en histograma
- Crecimiento de regiones
- División y mezcla
- Clasificación (también conocida como clustering)
- Basados en teoría de grafos
- Basados en reglas y conocimientos

10.1 Binarización por detección de umbral

Una de las principales técnicas en los sistemas de visión industrial para la detección de objetos es la utilización de los umbrales en el tratamiento de imágenes, especialmente en aplicaciones que requieran procesar una cantidad elevada de información.

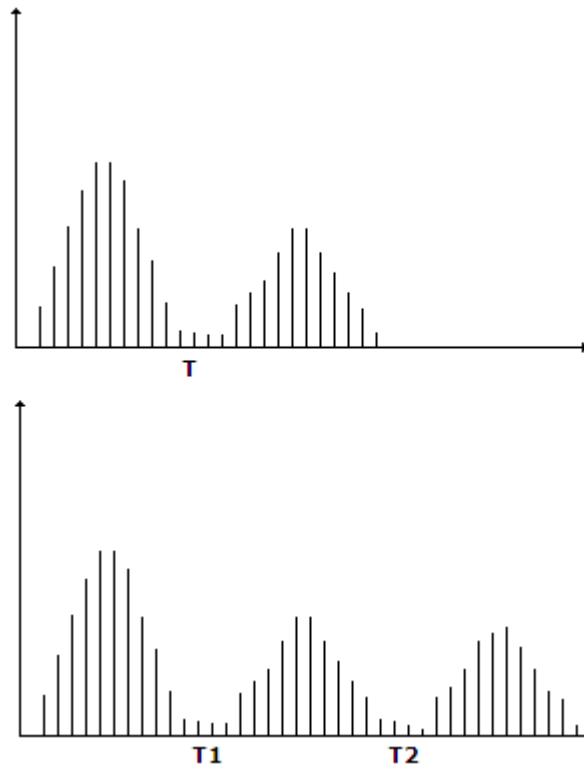


figura 14: histogramas de intensidad; parte superior: umbral único y parte inferior: umbrales múltiples

Supongamos que el histograma de intensidad mostrado en la figura anterior superior corresponde a una imagen $f(x,y)$ compuesta por objetos claros sobre un fondo oscuro, teniendo los píxeles del objeto y del entorno intensidades agrupadas en dos tonos dominantes. Una forma obvia de extraer los objetos del entorno es seleccionar un nivel T que separe los dos niveles de intensidad. De esta forma un píxel (x,y) para el cual $f(x,y) > T$ será un píxel del objeto en caso contrario, será del entorno. En la figura inferior se muestra un caso ligeramente más general del uso de esta técnica. Ahora el histograma se caracteriza por tres modos dominantes (por ejemplo, dos tipos de objetos sobre un fondo oscuro). El proceso será similar: se clasifica un punto como perteneciente a una de las clases de objetos si $T_0 < f(x,y) \leq t_2$ a la otra clase de objeto si $f(x,y) > T_1$ y al entorno si $f(x,y) \leq T$

Los procedimientos que usan este umbral multinivel suelen ser menos fiables que aquellos que usan el umbral simple debido a la dificultad que existe para establecer umbrales múltiples que aislen efectivamente regiones de interés.

Basándonos en los conceptos anteriores, podemos considerar la fijación de umbral como una operación que implica pruebas con respecto a una función T como sigue:

$$T = T[x, y, p(x, y), f(x, y)]$$

Donde $f(x, y)$ es la intensidad en el punto (x, y) y $p(x, y)$ es alguna propiedad local del punto, por ejemplo, la intensidad media de un entorno de vecindad centrado en (x, y) . Se creará una imagen binaria $g(x, y)$ definiendo

$$g(x, y) = \begin{cases} 0 & \text{si } f(x, y) > T \\ 1 & \text{si } f(x, y) \leq T \end{cases}$$

Examinando $g(x, y)$ se ve que los píxeles a los que se les asigna valor 0 corresponden a los objetos, mientras que los que corresponden al entorno tienen valor 1.

Cuando T depende sólo de $f(x, y)$, se llama *umbral global*. Si T depende tanto de $f(x, y)$ como de $p(x, y)$, entonces el umbral se llama *local*. Si T depende de las coordenadas espaciales x e y se llama *umbral dinámico*.

10.2 Selección del umbral óptimo

Hemos mencionado anteriormente que es muy difícil determinar cuál es el umbral óptimo para poder llevar a cabo una binarización adecuada.

Es posible considerar un histograma como formado por la suma de funciones de densidad de probabilidad. En el caso del histograma bimodal, la función global que aproxima el histograma viene dada por:

$$P(z) = P_1 p_1(z) + P_2 p_2(z)$$

Donde z es una variable aleatoria que representa la intensidad, $p_1(z)$ y $p_2(z)$ son las funciones de densidad de probabilidad y P_1 y P_2 son las probabilidades a priori. Éstas dos últimas

son las probabilidades de ocurrencia de los dos tipos de niveles de intensidad de la imagen. Por ejemplo, consideremos una imagen simple cuyo histograma es el dado en la figura siguiente:

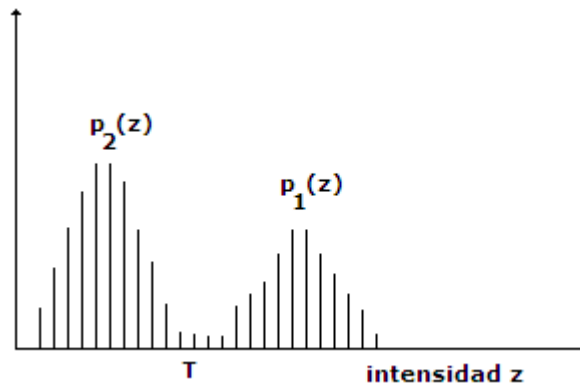


figura 15: histograma de intensidad como suma de dos funciones de densidad de probabilidad

El histograma global se puede aproximar por la suma de dos funciones de densidad de probabilidad. Si se sabe que los píxeles más oscuros representan objetos y se sabe también que el 80% del área de la imagen está ocupada por píxeles de objeto, entonces $P_1=0.8$. Se debe cumplir que $P_1+P_2=1$ que viene a decir simplemente que, en este caso, el 20% restante son píxeles de fondo.

Formemos dos ecuaciones de z :

$$d_1(z)=P_1p_1(z) \text{ y } d_2(z)=P_2p_2(z)$$

Sabemos por la teoría de decisiones que el error medio de clasificar equivocadamente un píxel de objeto como de fondo, o viceversa, se minimiza usando la siguiente regla: dado un píxel con valor de intensidad z , se sustituye este valor de z en la ecuación anterior. Clasificamos el píxel como píxel de objeto si $d_1(z) > d_2(z)$ o como píxel de fondo en caso contrario. El valor óptimo viene dado entonces por el valor de z para el cual se cumple la igualdad $d_1(z)=d_2(z)$, esto es, haciendo $z=T$ en la ecuación anterior. Deducimos que el umbral óptimo satisface la ecuación siguiente:

$$P_1p_1(z) = P_2p_2(z) \text{ (A)}$$

Supongamos que $p_1(z)$ y $p_2(z)$ son funciones de densidad gaussianas y por lo tanto conocidas, esto es,

$$p_1(z) = \frac{1}{\sqrt{2\pi}\sigma_1} \exp\left[-\frac{(z - m_1)^2}{2\sigma_1^2}\right]$$

$$p_2(z) = \frac{1}{\sqrt{2\pi}\sigma_2} \exp\left[-\frac{(z - m_2)^2}{2\sigma_2^2}\right]$$

Haciendo $z=T$ en estas expresiones, sustituyendo en la formula (A), se obtiene la ecuación de segundo grado en T :

$$AT^2 + BT + C = 0$$

donde:

$$A = \sigma_1^2 - \sigma_2^2$$

$$B = 2(m_1\sigma_2^2 - m_2\sigma_1^2)$$

$$C = \sigma_1^2 m_2^2 - \sigma_2^2 m_1^2 + 2\sigma_1^2 \sigma_2^2 \ln \frac{\sigma_2 P_1}{\sigma_1 P_2}$$

La existencia de dos posibles soluciones indica que se pueden requerir dos valores de umbral para obtener una solución óptima.

Si las desviaciones estándar son iguales, $\sigma_1 = \sigma_2 = \sigma$, será suficiente un umbral único:

$$T = \frac{m_1 + m_2}{2} + \frac{\sigma}{2} \ln \frac{P_2}{P_1}$$

Si $\sigma = 0$ o $P_1 = P_2$, el nivel óptimo es el valor medio de las medias. La primera condición viene a decir que tanto la intensidad del objeto, como la del entorno, son constantes en toda la

imagen. La última condición significa que los píxeles del objeto y del entorno tienen igual probabilidad de ocurrir, condición ésta que se cumple siempre que el número de píxeles del objeto sea igual al número de píxeles del entorno en una determinada imagen.

El procedimiento desarrollado anteriormente es aplicable a la selección de múltiples niveles. Supóngase que pudimos modelar un histograma multinivel como la suma de n funciones de densidad de probabilidad como sigue:

$$p(z) = P_1 p_1(z) + \dots + P_n p_n(z)$$

La aplicación de los métodos de umbral se podría ver como una clasificación de los píxeles entre n categorías posibles. La regla de decisión del error mínimo estará basada ahora en n funciones de la forma:

$$d_i(z) = P_i p_i(z) \quad i=1,2,\dots,n$$

Un píxel dado con una intensidad z se asigna a la k -ésima categoría si $d_k(z) > d_j(z)$ siendo $j=1,2,\dots,n$ con $j \neq k$. Como antes, el umbral óptimo entre la categoría k y la categoría j , llamado T_k se obtiene resolviendo la ecuación:

$$P_k p_k(T_{kj}) = P_j p_j(T_{kj})$$

10.3 Umbrales basados en varias variables

Las técnicas vistas hasta ahora trabajan con una única variable, la intensidad. En algunas aplicaciones es posible usar más de una variable para caracterizar cada píxel de una imagen. Realizando así no solamente la capacidad de diferenciar objetos y entorno, sino también la de distinguir entre los mismos objetos. Un ejemplo sobresaliente es la percepción del color, donde las componentes rojo, verde y azul (RGB) se usan para formar una imagen en color. En este caso, cada píxel se caracteriza por tres valores y resulta posible construir un histograma tridimensional. El procedimiento básico es el mismo que el que usa para una sola variable. Por ejemplo: dadas tres imágenes de grises con 16 niveles de intensidad que corresponden a las componentes RGB

de un sensor color, formaremos un tetraedro de dimensiones $16 \times 16 \times 16$ e insertaremos en cada celda de este tetraedro los píxeles cuyas componentes RGB tengan intensidades que correspondan a las coordenadas definidas por la posición de esta celda particular. El contenido de cada celda se puede dividir por el número total de píxeles de la imagen para obtener así un histograma normalizado.

El concepto de selección de umbral se traduce ahora en la búsqueda de puntos del espacio tridimensional, cuyas características de intensidad sean análogas a un tono dominante del histograma de una variable. Supongamos que encontramos dos grupos de puntos en un histograma dado, correspondiéndose un grupo a objetos y el otro al entorno. Teniendo en cuenta que cada píxel tiene ahora tres componentes y que, por lo tanto, puede ser tratado como un punto del espacio tridimensional, podemos segmentar la imagen usando el siguiente procedimiento: para cada píxel de la imagen calculamos la distancia entre este píxel y el centro de cada grupo. Si el píxel está cercano al centro del grupo de objetos, lo etiquetamos con un 1, caso contrario lo etiquetamos con un 0. Este concepto es fácilmente ampliable a más componentes del píxel y más grupos. La principal dificultad es encontrar grupos con significado.

10.4 Crecimiento de Regiones

Cuando hablamos del crecimiento de regiones nos estamos refiriendo al proceso que agrupa píxeles o subregiones en regiones mayores. Arrancando con un conjunto de píxeles llamados semilla, las regiones crecen a partir de ellas, incluyendo cada vecino de la semilla con similares propiedades de intensidad, niveles de gris, textura, color, etc.

Es un proceso repetido, donde cada semilla crece iterativamente hasta que todos los píxeles sean procesados y de esta manera, forman diferentes regiones cuyos límites o contornos quedan atrapados en polígonos.

Los puntos importantes de este algoritmo son:

- Selección inicial de los píxeles semilla que representan regiones y determinación de propiedades que permitan la inclusión del resto de los píxeles de la imagen para formar las distintas regiones.
- Hacer crecer las regiones basándose solamente en ciertas propiedades de la imagen puede llevar a resultados incorrectos, por eso debe tenerse en cuenta la información de conectividad o adyacencia de píxeles durante el proceso.

- Similitud: denota la mínima diferencia en el nivel de gris observado entre dos píxeles adyacentes o nivel de gris promedio entre un conjunto de píxeles, los cuales denotan regiones diferentes. Si la diferencia es menor al valor de similitud del umbral, los píxeles pertenecen a la misma región.
- Área de la región: el área de mínimo umbral está asociada con la región más pequeña en píxeles. En la imagen segmentada, ninguna región debe ser menor a éste umbral, que es definido por el usuario.

Post-procesamiento en segmentación basada en crecimiento de regiones

Si el algoritmo no está óptimamente parametrizado la técnica de crecimiento de regiones puede llevar a un crecimiento por debajo o por encima de lo normal.

Se han desarrollados diversos postprocesadores que combinan información de segmentación obtenida del crecimiento de regiones con segmentación basada en detección de bordes. Algunos de ellos se basan en heurísticas generales para reducir el número de regiones pequeñas en la imagen segmentada que no pueden ser mezcladas con regiones adyacentes de forma correcta siguiendo el criterio de homogeneidad aplicado.

Grafo de regiones adyacentes

La relación de adyacencia entre las regiones de una escena puede ser presentada por medio de un *grafo de regiones adyacentes* o *GRA*. Las regiones en la escena son representadas por un conjunto de nodos $N = \{N_1, N_2, \dots, N_n\}$ en el *GRA*. Donde el nodo N_i representa la región R_i en la imagen y las propiedades de la región se guardan en la estructura de datos del nodo N_i .

El borde $e_{i,j}$ entre N_i y N_j representa la adyacencia entre las regiones R_i y R_j .

Dos regiones R_i y R_j son adyacentes si existe un píxel en la región R_i y un píxel en R_j los cuales son adyacentes entre sí. La adyacencia pueden estar basada en E_4 (4 vecinos) u E_8 (8 vecinos). La relación de adyacencia es reflexiva, simétrica pero no necesariamente transitiva. En la figura a continuación se muestra un grafo de regiones adyacentes de una imagen con cinco regiones distintas.

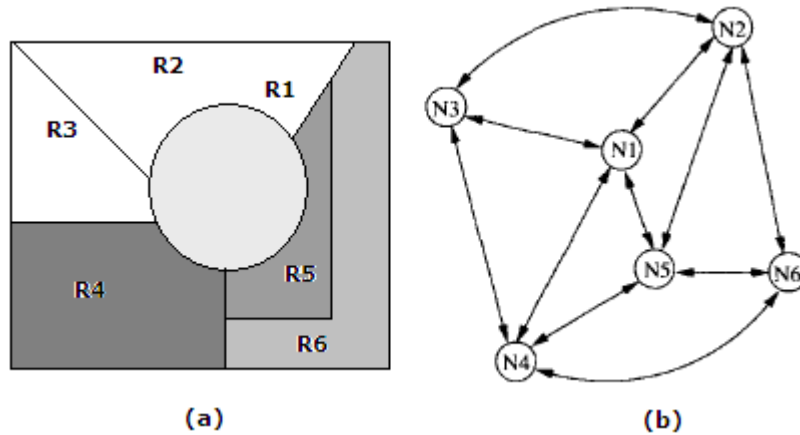


figura 16: (a) una escena con seis regiones distintas, (b) el grafo de adyacencia de la escena

Una matriz binaria A es llamada una matriz de adyacencia cuando representa un grafo de regiones adyacentes o GRA. Cuando los nodos N_i y N_j en el GRA son adyacentes, $a_{i,j}$ en A vale 1. Como la relación de adyacencia es reflexiva, los elementos de la diagonal de la matriz valen todos 1. A continuación se muestra una matriz de adyacencia de una escena con múltiples regiones.

$$A = \begin{bmatrix} & N1 & N2 & N3 & N4 & N5 & N6 & N7 \\ N1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ N2 & 1 & 1 & 1 & 0 & 1 & 0 & 1 \\ N3 & 1 & 1 & 1 & 1 & 0 & 0 & 1 \\ N4 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ N5 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ N6 & 1 & 1 & 0 & 1 & 1 & 1 & 0 \\ N7 & 1 & 0 & 1 & 1 & 1 & 0 & 1 \end{bmatrix}$$

figura 17: matriz de adyacencia

10.5 Crecimiento de regiones mediante adición de píxeles

Se trata de una técnica que agrupa píxeles o subregiones en regiones más grandes. El método más simple es la agregación o adición de píxeles, en donde comenzamos con un conjunto de puntos “semilla” y vamos haciendo crecer las regiones añadiendo a cada punto semilla aquellos píxeles vecinos que tengan propiedades similares, por ejemplo, intensidad, textura o color.

Cuando las imágenes son en escala de grises, las tres bandas de color tienen el mismo valor de intensidad para una localización de píxel dado (x,y) .

El procedimiento es interactivo, en el sentido que el usuario operador coloca las semillas y elige el umbral. Sin embargo, cuando esta técnica se intenta automatizar surge el problema de la selección de las semillas iniciales, que representan adecuadamente regiones de interés.

Advertimos otro problema en la elección de las propiedades adecuadas para asignar los píxeles a las diversas regiones. En cualquier caso, tanto si se elige la opción interactiva como automática, el criterio para obtener el algoritmo consiste en dejar de hacer crecer una región cuando no existan más píxeles que satisfagan el criterio de inclusión en esa región.

10.6 Crecimiento y división

Sea R la región que incluye la imagen completa, podemos ver la segmentación que divide a R en n subregiones R_1, R_2, \dots, R_n cumpliéndose:

$$1) \bigcup_{i=1}^n R_i = R$$

2) R_i es una región conectada, siendo $i=1, 2, \dots, n$

$$3) R_i \cap R_j = \emptyset \quad \forall i, i \neq j$$

4) $P(R_i)$ es verdadero para $i=1, 2, \dots, n$

5) $P(R_i \cup R_j) = \text{falso}$ para $i \neq j$

Donde $P(R_i)$ es un predicado lógico definido sobre los puntos del conjunto R_i y \emptyset es el conjunto vacío.

La condición 1 indica que la segmentación debe ser completa, esto es, todos los píxeles deben estar en alguna región determinada. La segunda condición obliga a que todos los píxeles de una región estén conectados. La condición 3 indica que las regiones deben ser disjuntas. La condición 4 nos indica las propiedades que los píxeles de una región segmentada deben satisfacer. Por ejemplo, $P(R_i)$ es verdadero si todos los píxeles de R_i tienen la misma intensidad. Por último la condición 5 nos indica que las regiones R_i y R_j son distintas según el criterio del predicado P . En los algoritmos de segmentación se hace uso de estas condiciones.

10.7 División y fusión de imágenes

El procedimiento descrito anteriormente hace crecer las regiones partiendo de un conjunto de puntos semillas dados. Una alternativa consiste en dividir inicialmente la imagen en un conjunto de regiones arbitrarias disjuntas para después fusionar y/o dividir estas regiones con objeto de satisfacer las condiciones mencionadas anteriormente. A continuación se explica un algoritmo de fusión y división que trabaja de forma iterativa.

Llamemos R a la región que corresponde a la imagen completa y seleccionamos un predicado P . Suponiendo que la imagen es cuadrada, un método para segmentar R consiste en la división sucesiva de regiones cuadradas cada vez más pequeñas de forma que, para cualquier región R_i , $P(r_i)$ sea verdadero. El procedimiento comienza con la región R . Si $P(R)$ es falso, dividiremos la imagen en cuadrantes, Si P es falso para algún cuadrante, subdividiremos ese cuadrante en subcuadrantes y así sucesivamente. Esta técnica de división se puede representar en la forma de los llamados árboles cuaternarios (esto es, un árbol en el cual cada nodo tiene exactamente cuatro descendientes.) En la figura siguiente se muestra esta forma de representación. La raíz del árbol se corresponde con la imagen completa y cada nodo se corresponde a una subdivisión. En este caso sólo R_i ha sido subdividido posteriormente.

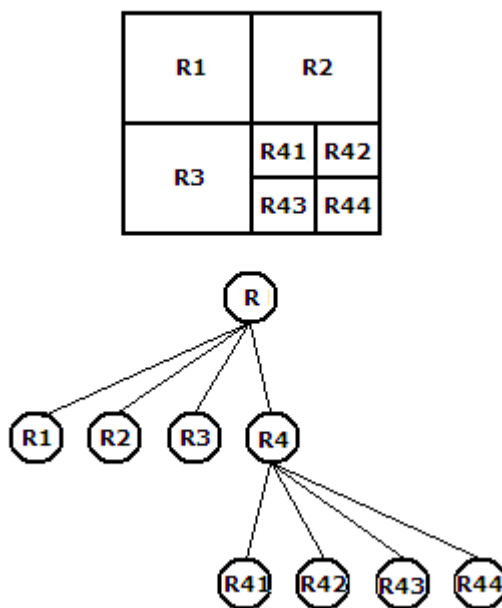


figura 18: parte superior: imagen dividida; parte inferior: árbol cuaternario correspondiente

Si usásemos sólo la división, la partición final estaría formada por regiones adyacentes con idénticas propiedades. Esto se puede evitar empleando también la fusión. Para cumplir las condiciones de segmentación, fusionaremos sólo regiones adyacentes cuyos píxeles combinados satisfagan el predicado P , esto es, fusionaremos dos regiones adyacentes R_i y R_j sólo si $P(R_i \cup R_j) = \text{verdadero}$. El procedimiento anterior se puede esquematizar de la siguiente manera:

- 1) dividir en cuatro cuadrantes disjuntos aquellas regiones para las que $P(R_i) = \text{falso}$
- 2) fusionar las regiones adyacentes R_i y R_j para las cuales se cumple que $P(R_i \cup R_j) = \text{verdadero}$
- 3) para cuando no sea posible realizar más fusiones ni divisiones

Se conocen muchas variaciones a este procedimiento básico, por ejemplo, una posibilidad es dividir inicialmente la imagen en un conjunto de bloques cuadrados. La división posterior se realizará como se explicó anteriormente, pero la fusión está limitada inicialmente a grupos de cuatro bloques que sean descendientes en la representación del árbol cuaternario y que satisfagan el predicado P . Cuando no es posible realizar más fusiones de este tipo, el procedimiento se termina haciendo una fusión de las regiones que cumplan el paso 2. En este punto, las regiones que se han fusionado pueden ser de distintos tamaños. La principal ventaja de este método es que se usa el mismo árbol cuaternario para la división y la fusión, hasta que se lleva a cabo el último paso de la fusión.

En la figura siguiente se muestra una ilustración intuitiva de los algoritmos de división y fusión. Se trata de un objeto único y el fondo. Suponiendo que tanto el objeto como el fondo, tienen intensidades constantes y que $P(R_i) = \text{verdadero}$ si todos los píxeles de R_i tienen la misma intensidad. De esta forma, para la región R de la imagen completa $P(R) = \text{falso}$ dividiéndose por tanto la imagen como se muestra en (a). En el siguiente paso, sólo la región superior izquierda cumple el predicado, quedándose como está y dividiendo en subcuadrantes las otras tres regiones, como se muestra en la figura (b). A partir de ahora, algunas regiones se pueden fusionar, con la excepción de los dos subcuadrantes de la parte inferior del objeto; éstas regiones no cumplen el predicado y deben ser divididas posteriormente. El resultado de la operación de fusión y división se muestra en (c). En este punto, todas las regiones hacen verdadero a P y una fusión de las regiones adecuadas obtenidas con la última operación de división da como resultado la figura (d).

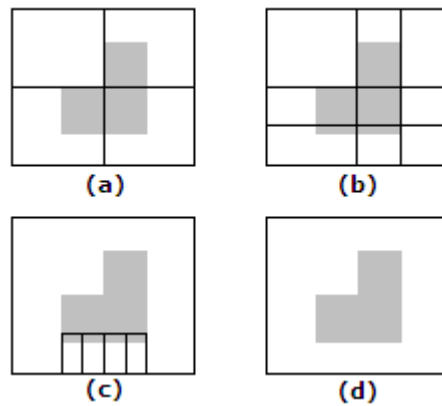


figura 19: algoritmo de fusión y división

10.8 Extracción de regiones por el color

Dos factores principales motivan la utilización del color en el procesamiento de imágenes digitales. En primer lugar, en el análisis de imágenes, el color es un potente descriptor que a menudo significa la identificación y extracción de objetos de una escena. En segundo lugar, el ojo humano puede distinguir una amplia gama de colores comparado con los niveles de gris.

Basándonos en el modelo de color RGB, se pueden extraer de la imagen aquellas regiones en las que predomine una determinada componente de color. El método consiste en elegir un determinado predicado y determinar en toda la imagen los píxeles que cumplen dicho predicado. Esos píxeles se marcan en blanco y el resto en negro. De esta forma obtenemos una imagen binaria.

En este caso también es posible determinar el área de las regiones obtenidas.

10.9 Segmentación basada en clustering

Las técnicas de segmentación de imágenes orientadas a los datos pueden ser basadas en histograma o basadas en clusters. Los algoritmos de segmentación basados en histograma producen una segmentación individual por cada característica de un conjunto de datos de múltiples características, luego solapan los resultados de la segmentación de cada característica para producir más regiones fragmentadas.

Las técnicas de segmentación basadas en clusters, pueden ser más apropiadas que aquellas basadas en histogramas, donde cada píxel posee diversos atributos y es representado por un vector. Las técnicas de clustering no son modernas, nacieron en los 60' y se han aplicado a

diversos campos entre ellos el reconocimiento óptico de caracteres u ROC (OCR en inglés), reconocimientos de huellas digitales, segmentación de imágenes médicas, etc.

En las técnicas de segmentación de imágenes basadas en clustering de minimización de costo, a cada configuración de cluster se le asigna un valor o costo para medir su eficiencia. Una función de costo apropiada mide la eficiencia del cluster. Normalmente, el costo para una configuración de clusters es la raíz cuadrada de su error (la suma euclidiana de distancias entre cada punto al centro de su cluster.) Los valores de costo bajo indican mejores resultados.

CAPÍTULO CUARTO: Imágenes Médicas

11 Tomografía Computada (TC)

11.1 Definición

La tomografía computada es un procedimiento de diagnóstico por imágenes que utiliza una combinación de radiografías y tecnología computarizada para obtener imágenes de cortes transversales (a menudo llamadas “rebanadas”) del cuerpo, tanto horizontales como verticales. La tomografía computada muestra imágenes detalladas de cualquier parte del cuerpo, incluidos los huesos, músculos, grasa y órganos. Las tomografías computarizadas muestran más detalles que las radiografías estándar. Otra de las ventajas es que mediante este procedimiento se reduce al mínimo la exposición a la radiación.

Tipos de Tomografía Computada (TC):

- **Tomografía Lineal:**

El objetivo de la tomografía lineal es la visualización preferencial de un plano del volumen examinado.

- **Tomografía Computarizada:**

El empleo de la computación para reconstruir una imagen a partir de la información obtenida cuando una fuente de rayos X gira en torno al volumen en estudio y lo examina desde distintos ángulos.

- **Tomografía Computarizada por contraste:**

El refuerzo yodado mejora las posibilidades del método pero debe indicarse según el cuadro clínico del paciente. En el sistema nervioso central es indispensable ante la sospecha de tumor, absceso, hematoma subdural, etc.

- **Tomografía computarizada dinámica:**

Técnica que permite visualizar la evolución temporal, por ejemplo, de una imagen vascular contrastada por yodo. Se usa para estudios dinámicos cerebrales, aprovechando la rápida secuencia de barrido de la TC de 4ª y 5ª generación.

- **PET:**
Tomografía de emisión de positrones.
- **SPECT:**
Tomografía de emisión de fotones.

11.2 Aplicaciones

Las tomografías computadas pueden realizarse para ayudar a diagnosticar tumores, estudiar hemorragias internas o buscar otras lesiones o daños.

11.3 Obtención de imágenes por TC

Ver cuadro comparativo en punto 15.5.

Tomógrafos:

- 1ª generación: el movimiento del tubo de rayos X es lineal, enfrentado a 1 o 2 detectores. En una segunda etapa se gira al paciente 90° y se repite la operación. Se continúa con una exploración a 180° y se termina a 270°.
- 2º generación: sigue siendo de translación lineal, pero se hace una exploración con ángulos que varían de 10° en 10°. Aumenta el número de detectores. Tiempo de exposición: 20 segundos.
- 3º generación: no hay translación lineal. El movimiento es exclusivamente en giro, el tiempo de exposición se reduce a 5 segundos.
- 4º y 5º generación: los detectores cubren todo el círculo alrededor del paciente y sólo gira el tubo de rayos X. El tiempo de exposición es de 2 segundos por corte.

Cómo captura la imagen el tomógrafo:

Elementos necesarios:

- 1 Mesa donde se pone al paciente: permite las rotaciones del tubo y los detectores
- 2 Generador de alta tensión para el tubo de rayos X.
- 3 Tubo de rayos X
- 4 Detectores que transducen la radiación en señales que la computadora puede entender.

Las estructura atravesadas absorben parte de la energía del haz según el coeficiente lineal de absorción. La radiación atenuada llega a los detectores donde se transduce primero en luz y luego en una corriente eléctrica (señal analógica) que es captada por un convertidor que la transforma en números (digitalización)

11.4 Ejemplos de TC

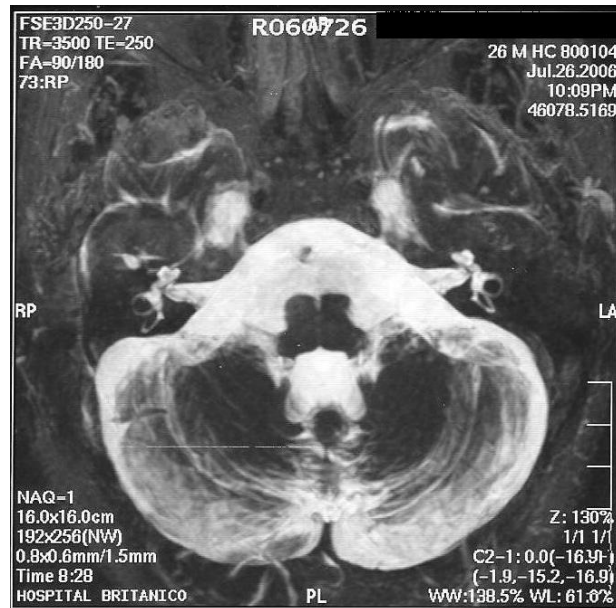


figura 20: tomografía computada

12 Resonancia Magnética (RMI)

12.1 Definición

La resonancia magnética por imágenes, también llamada RMI, es un método para producir imágenes precisas de los órganos internos del cuerpo.

Estas imágenes son de gran utilidad para el diagnóstico precoz de muchas enfermedades y para una localización precisa de las lesiones en los distintos órganos.

12.2 Aplicaciones

La RMI produce imágenes del cerebro y de la médula espinal. Por esa razón, los médicos frecuentemente usan RMI para examinar cráneo, cuello y columna. RMI es útil en el diagnóstico de muchas patologías, entre ellas:

- enfermedades del cerebro y sistema nervioso central y periférico
- enfermedades de los huesos, articulaciones y músculos
- búsqueda y localización de tumores de distintos órganos (hígado, vejiga, próstata, etc)
- Enfermedades del corazón y de los vasos sanguíneos

Debido al campo magnético del equipo, en algunos pacientes no se puede hacer un estudio de Resonancia Magnética. Esto incluye pacientes con marcapasos, neuro-estimuladores, implantes cocleares, cuerpos metálicos extraños en los ojos o clips metálicos ferromagnéticos.

Por ese motivo la historia clínica del paciente ayuda a los doctores a determinar si un estudio de Resonancia Magnética por Imágenes es o no apropiado.

12.3 Obtención de imágenes por RMI

Ver cuadro comparativo en punto 15.5.

12.4 Ejemplos de RMI



figura 21: resonancia magnética

12.5 TC vs RMI

Tomografía Computada (TC)	Resonancia Magnética (RM) (MRI en inglés)
Proceso más rápido que RM (para el cerebro puede demorar 20'' mientras que la RM para la misma región puede demorar el doble)	El proceso demora más tiempo que el de TC
Cortes más grandes. Los cortes de TC son mas pequeños, de 0.5 a 1 mm.	Cortes más pequeños que en TC. El corte mínimo en le RM es de 1 mm aunque para realizar una simulación en 3D se requieren cortes mas chicos, de 0,5 mm, porque cortes grandes distorsionan el resultado. Se requieren mas de 150 cortes para obtener una simulación en 3D aceptable.
Las tomografías computadas pueden realizarse para ayudar a diagnosticar tumores, estudiar hemorragias internas o buscar otras lesiones o daños.	Para ver tejidos se usa RM.
TC utiliza rayos X	RM se basa en campos magnéticos. Como no se usa radiación, durante un procedimiento de imágenes por resonancia magnética no existen riesgos asociados a la exposición a la radiación. Debido al uso de un fuerte campo magnético, la resonancia magnética no puede practicarse en pacientes con marcapasos, clipaje de aneurisma intracraneal, implantes cocleares, determinadas prótesis, bombas implantadas de infusión de medicamentos, neuroestimuladores, estimuladores del crecimiento óseo, determinados dispositivos de anticoncepción intrauterinos o cualquier otro tipo de implantes metálicos a base de hierro. La resonancia magnética también está contraindicada en presencia de objetos metálicos internos, como balas o esquirlas, así como grapas quirúrgicas, clavos, placas, tornillos, suturas metálicas o mallas de alambre.
En los rayos X estándar, se dirige un rayo de energía a la parte del cuerpo que se está estudiando. Una placa situada detrás de la zona del cuerpo captura las variaciones del rayo de energía después de que éste pasa a través de la piel, el hueso, el músculo y otros tejidos.	La máquina de resonancia magnética es grande y cilíndrica (con forma de tubo) y crea un fuerte campo magnético alrededor del paciente. El campo magnético, junto con una radiofrecuencia, altera el alineamiento natural de los átomos de

<p>Aunque puede obtenerse mucha información de los rayos X regulares, no pueden obtenerse muchos detalles de los órganos internos y otras estructuras.</p> <p>En la tomografía computarizada, el haz de rayos X se mueve en círculo alrededor del cuerpo. Esto permite tener vistas diferentes del mismo órgano o estructura, y proporciona muchos más detalles. La información de los rayos X es enviada a una computadora que interpreta los datos de los rayos X y los presenta en forma bidimensional en un monitor. La tecnología y software informático más modernos hacen posible la obtención de imágenes tridimensionales (3-D).</p> <p>Las tomografías pueden realizarse con o sin contraste. "Contraste" se refiere a una sustancia administrada oralmente o inyectada por vía intravenosa (IV) que hace que se vea más claramente el órgano o tejido en particular que se está estudiando. Los exámenes por contraste pueden requerir que usted ayune durante un determinado período de tiempo antes del procedimiento. Su médico le avisará de esto antes del procedimiento.</p>	<p>hidrógeno en el organismo. Luego, se utiliza una computadora para formar una imagen bidimensional (2D) de una estructura u órgano del cuerpo en función de la actividad de los átomos de hidrógeno. Para detectar más detalles, se pueden obtener vistas transversales. La resonancia magnética no utiliza radiación, a diferencia de las radiografías o la tomografía computarizada.</p> <p>Se crea un campo magnético y se envían pulsos de ondas de radio desde un escáner. Las ondas de radio golpean el núcleo de los átomos de su cuerpo, desplazándolos fuera de su posición normal. Mientras los núcleos se vuelven a alinear en la posición correcta, envían señales de radio. Estas señales son recibidas por una computadora que las analiza y las convierte en una imagen de la parte del cuerpo que está siendo examinada. Esta imagen aparece en la pantalla del monitor. Algunas máquinas de resonancia magnética parecen túneles angostos, mientras que otras son más abiertas.</p>
<p>Pasos del procedimiento de TC:</p> <ol style="list-style-type: none"> 1. Cuando el paciente llega para someterse a la TC, se le pedirá que se quite toda la ropa, joyas o cualquier objeto que pueda interferir con el escáner. 2. Si el paciente va a someterse al procedimiento con contraste, se le pondrá una línea intravenosa en la mano o en el brazo para introducir el medicamento de contraste. Para el contraste por vía oral, se le dará un medicamento al paciente para que lo trague. 3. El paciente se acostará en una mesa de estudio que se desliza en una abertura larga y circular de la máquina de escáner. 	<p>Pasos del procedimiento de RM:</p> <ol style="list-style-type: none"> 1. El paciente debe quitarse todas las joyas y objetos de metal, como horquillas y pasadores para el pelo, audífonos, gafas o anteojos, y piezas dentales. Esto es debido al fuerte campo magnético. 2. Si el medicamento de contraste (yodo) será administrado por vía intravenosa, se le colocará una vía intravenosa en la mano o el brazo. Si el líquido de contraste se administrará por vía oral, se le entregará al paciente para que lo trague. 3. El paciente se acostará en una mesa que se desliza en un túnel

<ol style="list-style-type: none"> 4. El personal de TC estará en otro cuarto donde están situados los controles del escáner. Sin embargo, el paciente estará constantemente a la vista del personal a través de una ventana. Los altavoces localizados dentro del escáner permitirán que el personal se comuniquen con el paciente y lo escuche. El paciente tendrá una campanilla para poder avisar al personal si tiene algún problema durante el procedimiento. 5. A medida que el escáner empieza a girar alrededor del paciente, pasarán bajas dosis de rayos X a través del cuerpo durante cortos períodos de tiempo. 6. Los rayos X absorbidos por los tejidos del cuerpo serán detectados por el escáner y transmitidos a la computadora. 7. La computadora transformará la información en una imagen que será interpretada por el radiólogo. 8. Es muy importante que el paciente permanezca totalmente inmóvil durante el procedimiento. 9. El tecnólogo estará vigilando al paciente en todo momento y estará en constante comunicación. 10. Es posible que el paciente tenga que esperar un momento, mientras el radiólogo observa las imágenes para asegurarse de que estén claras. Si las imágenes no están suficientemente claras para obtener la información adecuada, es posible que sea necesario realizar estudios adicionales. 	<p>dentro del escáner.</p> <ol style="list-style-type: none"> 4. El personal de RM estará en otro cuarto donde están situados los controles del escáner. Sin embargo, el paciente estará constantemente a la vista del personal a través de una ventana. Los altavoces localizados dentro del escáner permitirán que el personal se comuniquen con el paciente y lo escuche. El paciente tendrá un timbre para poder avisar al personal si tiene algún problema durante el procedimiento. 5. Durante el proceso de escaneo, se escuchará un chasquido a medida que se crea el campo magnético y los pulsos de las ondas de radio son enviados desde el escáner. 6. Es muy importante que el paciente permanezca totalmente inmóvil durante el examen. Puede anular el proceso si se mueve. 7. A intervalos, se le pedirá que contenga la respiración o que no respire durante algunos segundos, según la parte del cuerpo que se evalúe. Luego, se le indicará cuándo puede volver a respirar. El paciente tendrá que contener la respiración sólo durante algunos segundos, de manera que no le resultará incómodo. 8. El tecnólogo estará vigilando al paciente en todo momento y estará en constante comunicación.
--	--

Tabla 2: tomografía computada VS resonancia magnética

Características comunes en imágenes de TC y RM

- 1 Son en escala de grises (Salvo las imágenes obtenidas mediante SPECT o PECT que son distintos tipos de tomografías con emisión de positrones que hacen uso de yodo radioactivo inyectado por vía intravenosa. El yodo produce un efecto de color “resaltador” y las imágenes son en gamas de rojo, amarillo y azul).
- 2 Una imagen de escala de grises es de 8 bits, lo que significa que cada punto puede tener un valor entre 256. Por lo tanto, un píxel en una imagen de escala de grises puede ser de color negro, blanco, o de un tono de gris entre 254 valores posibles.
- 3 Vienen en un material plástico, comúnmente se las conoce como “placas”. Cada placa cuenta con una serie de imágenes relacionadas.
- 4 Poseen información que varía de acuerdo al equipamiento pero en general las placas cuentan con información impresa relacionada al paciente, la escala, la fecha, el hospital o clínica donde fue tomada y otros datos.

13 Formatos estándar de imágenes médicas

13.1 DICOM

DICOM (*Digital Imaging and Communication in Medicine*) es el estándar reconocido mundialmente para el intercambio de imágenes médicas, pensado para el manejo, almacenamiento, impresión y transmisión de imágenes médicas. Incluye la definición de un formato de fichero y de un protocolo de comunicación de red. El protocolo de comunicación es un protocolo de aplicación que usa TCP/IP para la comunicación entre sistemas. Los ficheros DICOM pueden intercambiarse entre dos entidades que tengan capacidad de recibir imágenes y datos de pacientes en formato DICOM.

DICOM permite la integración de escáneres, servidores, estaciones de trabajo, impresoras y hardware de red de múltiples proveedores dentro de un sistema de almacenamiento y comunicación de imágenes. Las diferentes máquinas, servidores y estaciones de trabajo tienen una declaración de conformidad DICOM (conformance statements) que establece claramente las clases DICOM que soportan. DICOM ha sido adoptado ampliamente por hospitales y está haciendo incursión en pequeñas aplicación de oficinas de dentistas y de doctores.

Formato de datos DICOM

Independientemente del uso, siempre se utiliza el mismo formato, incluyendo el uso de ficheros y de red. DICOM se diferencia de otros ficheros de datos en que agrupa la información dentro de un conjunto de datos. Es decir, una radiografía de pecho actualmente contiene el ID de paciente junto con ella, de manera que la imagen no puede ser separada por error de su información.

Los ficheros DICOM consisten en una cabecera con campos estandarizados y de forma libre, y un cuerpo con datos de imagen. Un objeto DICOM simple puede contener solamente una imagen, pero esta imagen puede tener múltiples "fotogramas" (frames), permitiendo el almacenamiento de bloques de cine o cualquier otros datos con varios fotogramas. Los datos de imagen puede estar comprimidos usando gran variedad de estándares, incluidos JPEG, JPEG Lossless, JPEG 2000, LZW y Run-length encoding (RLE).

Ficheros DICOM (.dcm)

Los ficheros DICOM corresponden a la parte 10 del estándar DICOM. Describe como almacenar información de imágenes médicas en un medio extraíble. Generalmente es obligatorio incluir también los meta datos de la imagen.

La extensión empleada es .dcm.

DICOM restringe los nombre de los ficheros a nombres de 8 caracteres. Del nombre del fichero no debe de extraerse ninguna información. Ésta es una fuente común de problemas con contenidos creados por desarrolladores que no han leído la especificación cuidadosamente. Éste es un requerimiento histórico para mantener compatibilidad con antiguos sistemas existentes. También es obligatorio la presencia de un directorio de contenido, el fichero DICOMDIR, que proporciona un índice e información de resumen para cada uno de los ficheros DICOM del contenido. La información del DICOMDIR proporciona substancialmente más información sobre cada fichero de los nombres de los ficheros, de manera que hay menos necesidad de nombres de fichero con significado.

El software que se implementó para este trabajo de grado soporta imágenes en formato DICOM (archivos DCM), decimos soporta en el sentido que puede extraer de un archivo DICOM la porción de información que corresponde a la imagen.

CAPÍTULO QUINTO: Segmentación de Imágenes Médicas

14 Clasificación de técnicas de segmentación

14.1 Contextuales

Tienen en cuenta las relaciones que existen entre los objetos de la imagen. Una técnica contextual tendrá en cuenta tanto las características del píxel como su vecindad con otros. Los algoritmos contextuales están basados en los conceptos de: *discontinuidad* y *similitud*.

Las técnicas basadas en discontinuidad dividen la imagen detectando cambios abruptos en los niveles de gris de la misma.

Las técnicas orientadas a similitud intentan crear regiones uniformes agrupando píxeles que satisfacen ciertos criterios de similitud. El resultado depende fuertemente del criterio y de la definición de conectividad empleada.

14.2 No contextuales

Ignoran las relaciones que existen entre los objetos que aparecen en la imagen. Los píxeles se agrupan de acuerdo a algún atributo común como ser: intensidad, color.

Las técnicas varían en los conceptos de uniformidad, predicados. El concepto de conectividad empleado puede ser el de 4 vecinos o el de 8 vecinos, dependiendo del enfoque.

15 Algoritmos y técnicas de segmentación

15.1 Umbralización

- Técnica no contextual.
- Transforma un conjunto de valores, variando en algún rango, en otro conjunto de valores que contienen sólo los valores deseados. Realiza esto aplicando un nivel de umbralización a los datos.

- Existen 2 clasificaciones posibles para cada píxel: si el píxel está por debajo del nivel del umbral recibirá un valor y si está por encima de este otro.
- Como existen sólo 2 posibles clasificaciones, el resultado será una imagen binaria (2 tonos).
- La medida del umbral marca el nivel de gris (es un atributo global).

Ejemplo:

A continuación se incluyen ejemplos de segmentación por umbralización utilizando el software “3d doctor” de la empresa Able Soft. Este software, en su versión trial, permite realizar ciertas operaciones como umbralización.

Recordemos que el nivel de umbral varía entre 0 y 255 y la imagen debe estar en escala de grises.

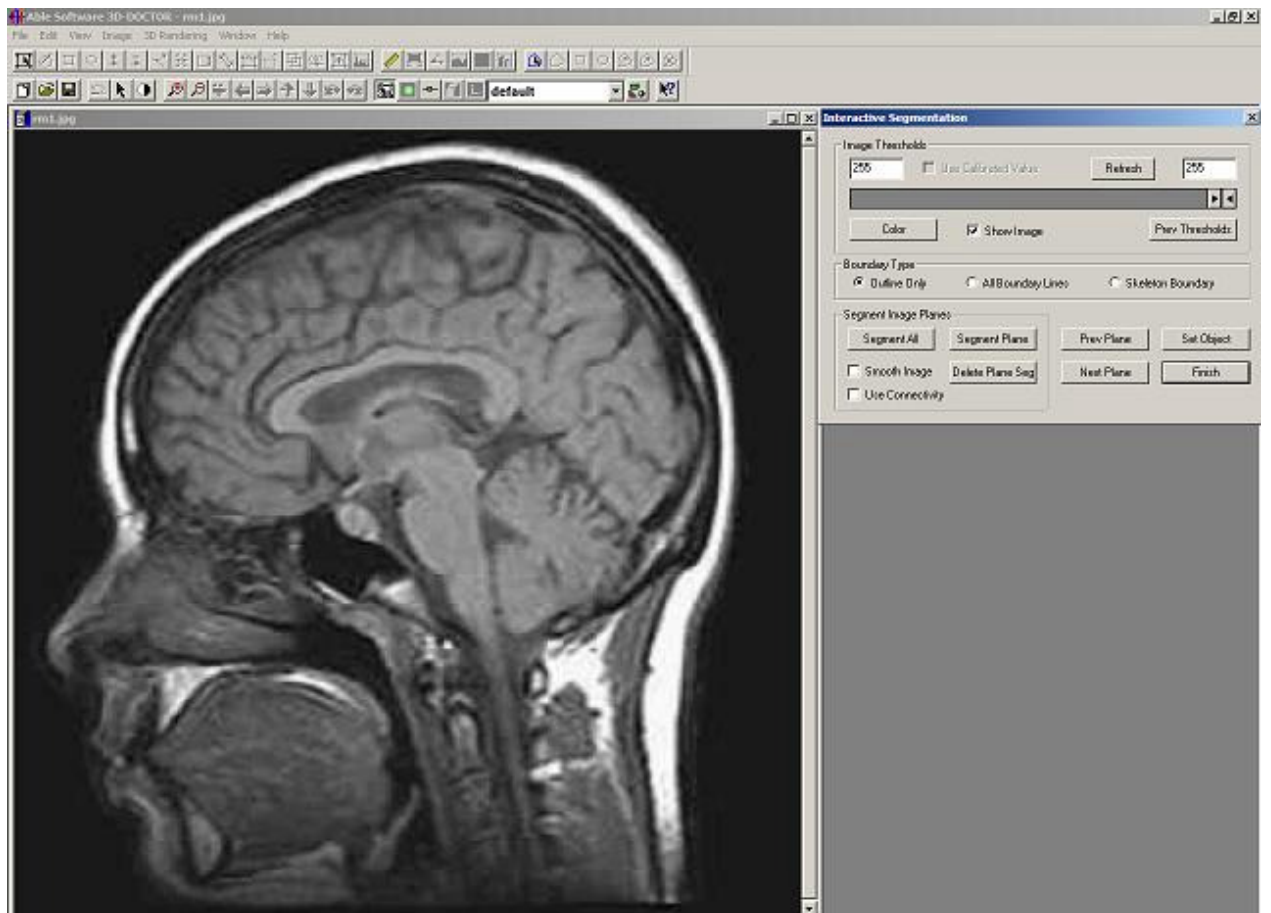


figura 22: umbralización empleado un nivel de umbral de 255

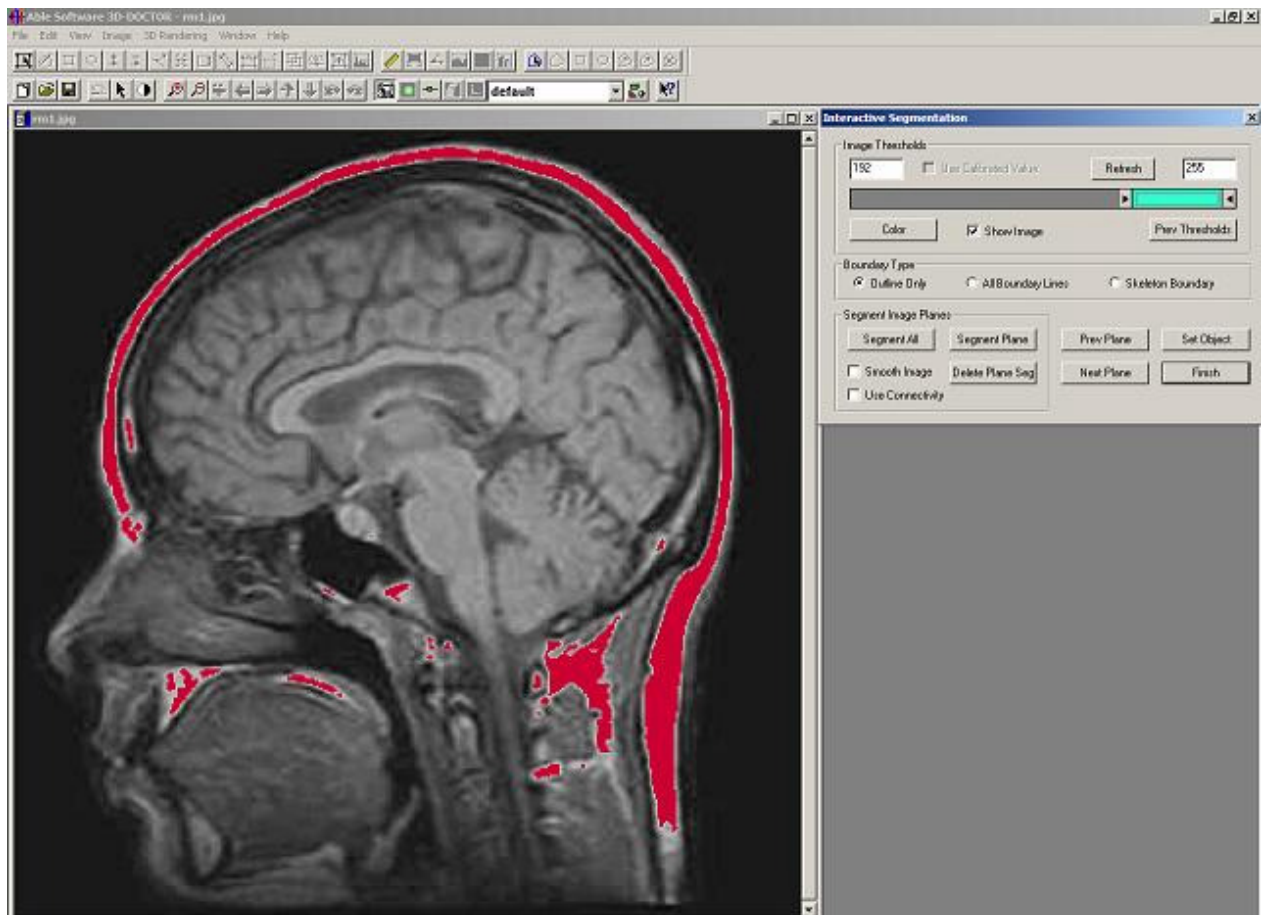


figura 23: umbralización empleado un nivel de umbral de 192

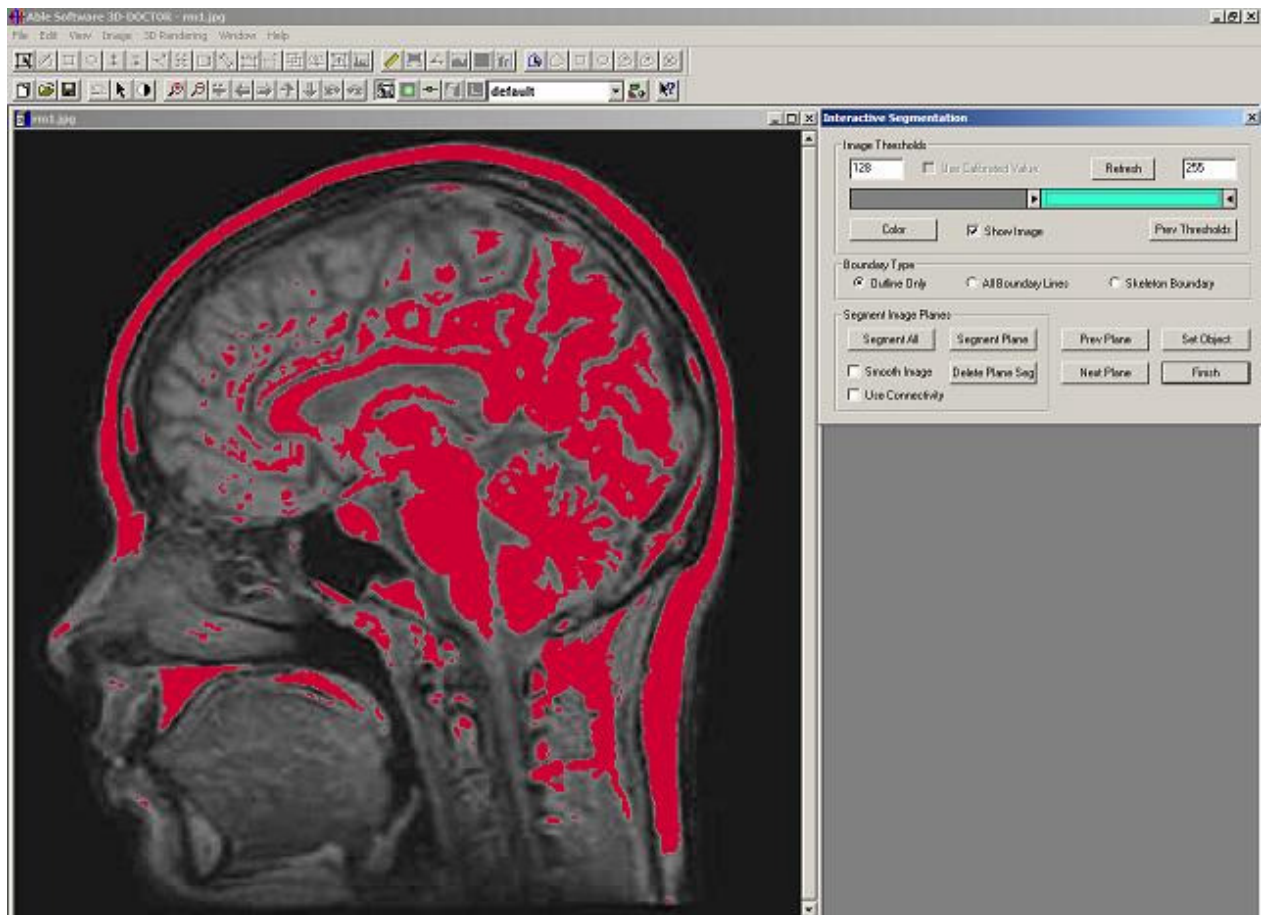


figura 24: umbralización empleado un nivel de umbral de 128

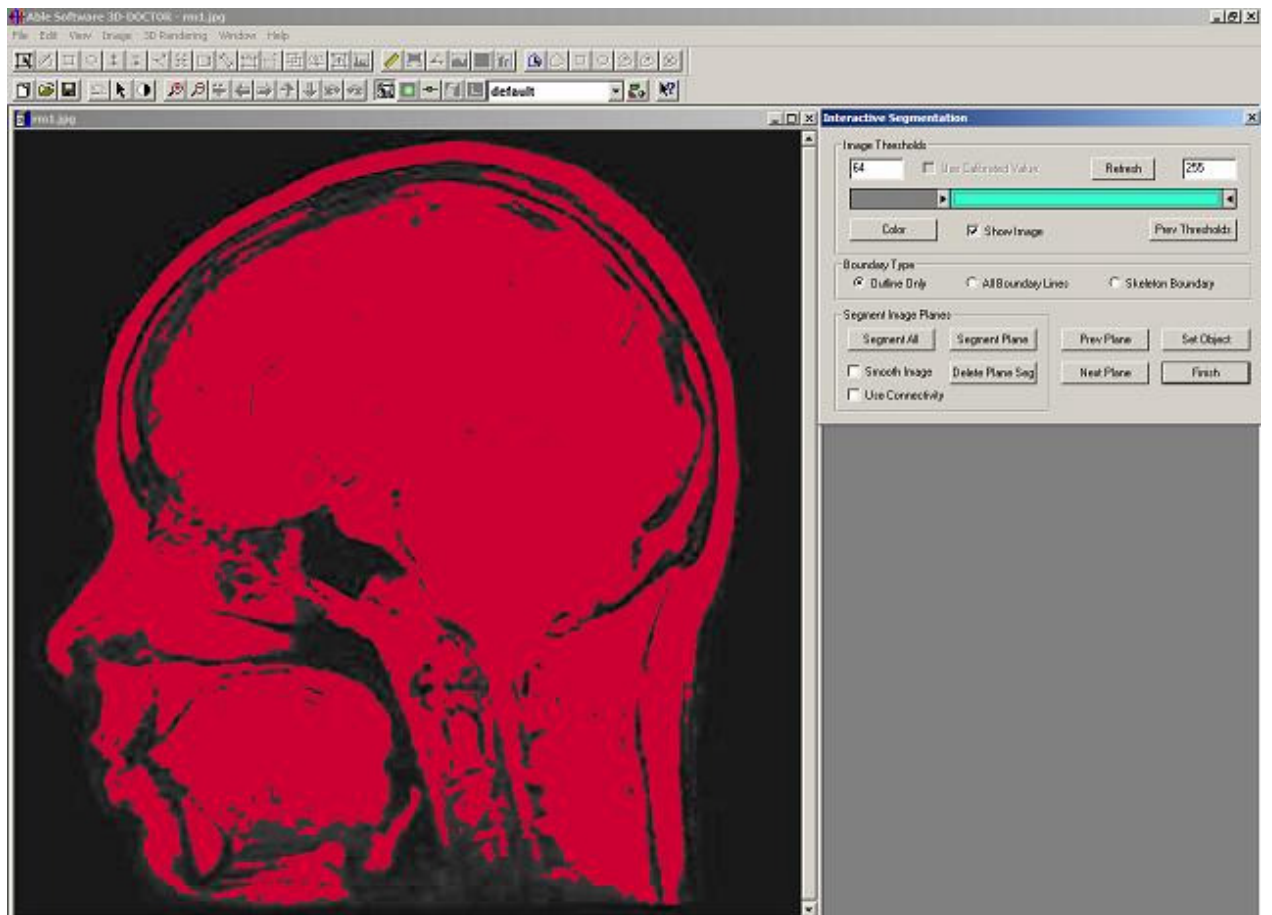


figura 25: umbralización empleado un nivel de umbral de 64

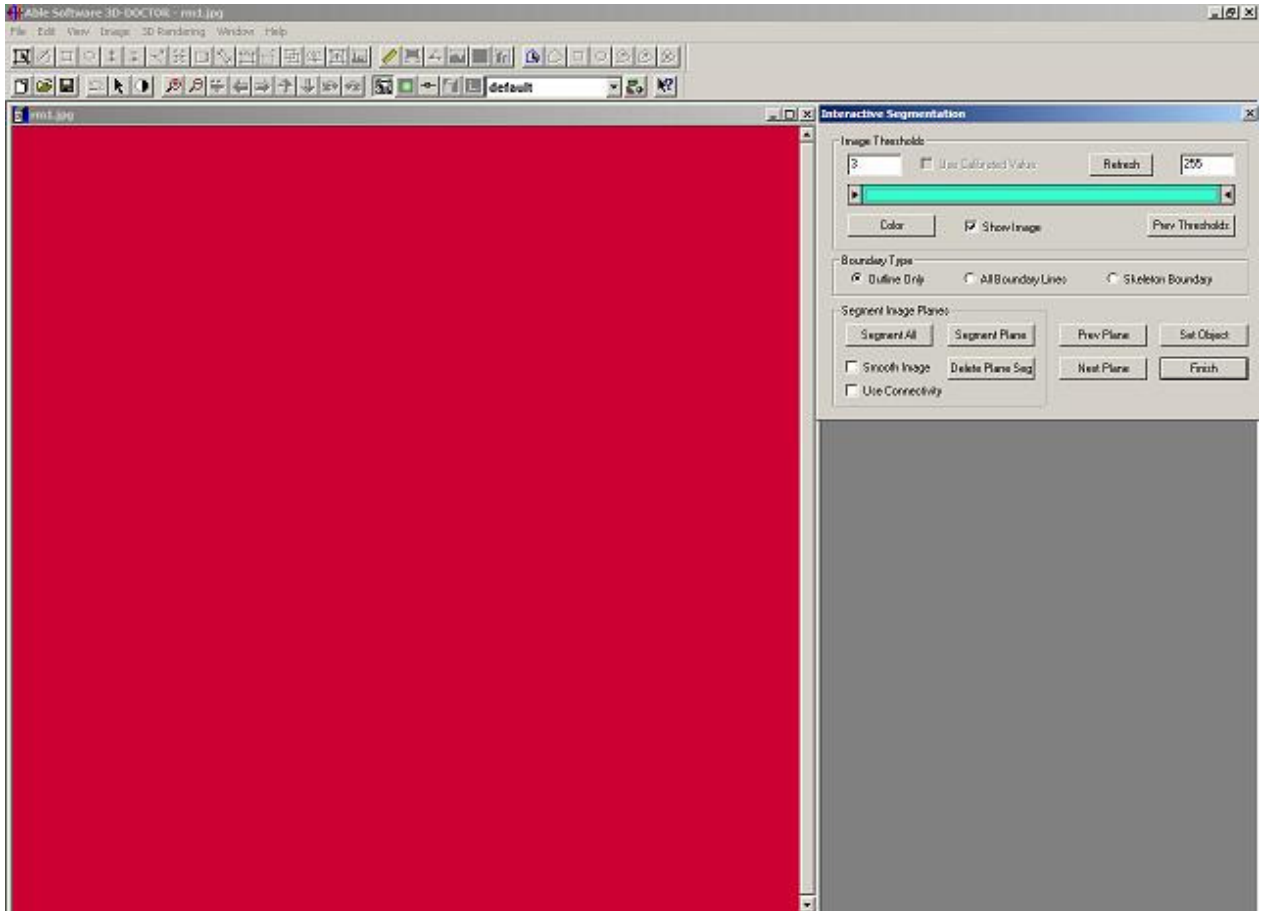


figura 26: umbralización empleado un nivel de umbral de 3

La lógica de este algoritmo es simple: asignar un color (en el ejemplo usamos el rojo) a cada píxel cuyo nivel de gris esté por encima de un nivel de umbral. Es un algoritmo que trabaja de forma global sobre toda la imagen por lo que no podemos, por ejemplo, segmentar una porción de la misma como ocurre con el algoritmo de semillas que veremos más adelante.

La ventaja que tiene es que es muy rápido computacionalmente. Esto se debe a su simpleza.

Notar que cuando el nivel de umbral tiende a 0 (límite inferior), la imagen queda completamente en rojo y cuando el umbral tiene a 255 (límite superior) la imagen no varía. Luego vemos que en los límites inferior y superior los resultados producidos por el algoritmo no son útiles.

Los algoritmos basados en regiones deben cumplir una serie de predicados que se detallan a continuación

Predicados que se deben cumplir:

- 1) Cada píxel debe ser asignado a alguna región.
- 2) Un píxel debe ser asignado a una y sólo una región.
- 3) Cada región debe estar conectada a un conjunto de píxeles.
- 4) Cada región debe ser uniforme.
- 5) Cualquier par mezclado de regiones adyacentes debe ser no uniforme.

15.2 Similitud de Regiones

- Técnica contextual.
- La uniformidad de una región se define en un predicado de similitud que debe cumplirse en todo píxel correspondiente a esa región. Normalmente el predicado se define en base al nivel de gris de los píxeles, el color u otra propiedad.
- Si un píxel cumple el predicado se le asigna un valor de verdadero, caso contrario se le asigna falso.
- El predicado se define como:

$$P(R) = \begin{cases} \text{verdadero si } |f(j, k) - f(m, n)| \leq D \\ \text{falso} & \text{en caso contrario} \end{cases}$$

Donde (j, k) y (m, n) son coordenadas de 2 píxeles vecinos en la región R . Este predicado establece que la región R es *uniforme* si y sólo la diferencia del nivel de gris de 2 píxeles vecinos no excede la constante D .

El algoritmo de segmentación extrae regiones de acuerdo a un criterio de homogeneidad. El criterio se basa generalmente en colores: el algoritmo tiende a aislar regiones de píxeles conectados que representan colores similares. Esas regiones corresponden a objetos presentes en cada imagen.

15.3 Crecimiento de Regiones

- Técnica contextual bottom-top.
- Tiene en cuenta una serie de semillas que define el operador. Las semillas son píxeles seleccionados por una persona.
- Intenta hacer crecer una región conectada uniforme originada en cada semilla.
- Un píxel se agrega a la región si:
 - No se asignó antes a otra región
 - Es vecino de la región
 - La región creada al agregar el nuevo píxel sigue siendo uniforme
- Problemas:
 - Da distintos resultados de acuerdo al concepto de vecindad empleado (4 vecinos u 8 vecinos.)
- Los resultados son muy sensibles a las semillas seleccionadas.
- Satisface los predicados 3 y 4 pero no el resto. Falla en satisfacer el 1ro. y el 2do. porque en general el número de semillas definidas por el usuario no son suficientes para crear una región para cada píxel.

Ejemplo:

A continuación se incluyen ejemplos de segmentación por crecimiento de regiones utilizando el software “A.T.I.C (Aplicación para Tratamiento de Imágenes por Computador)” de la empresa Ra-Ma.

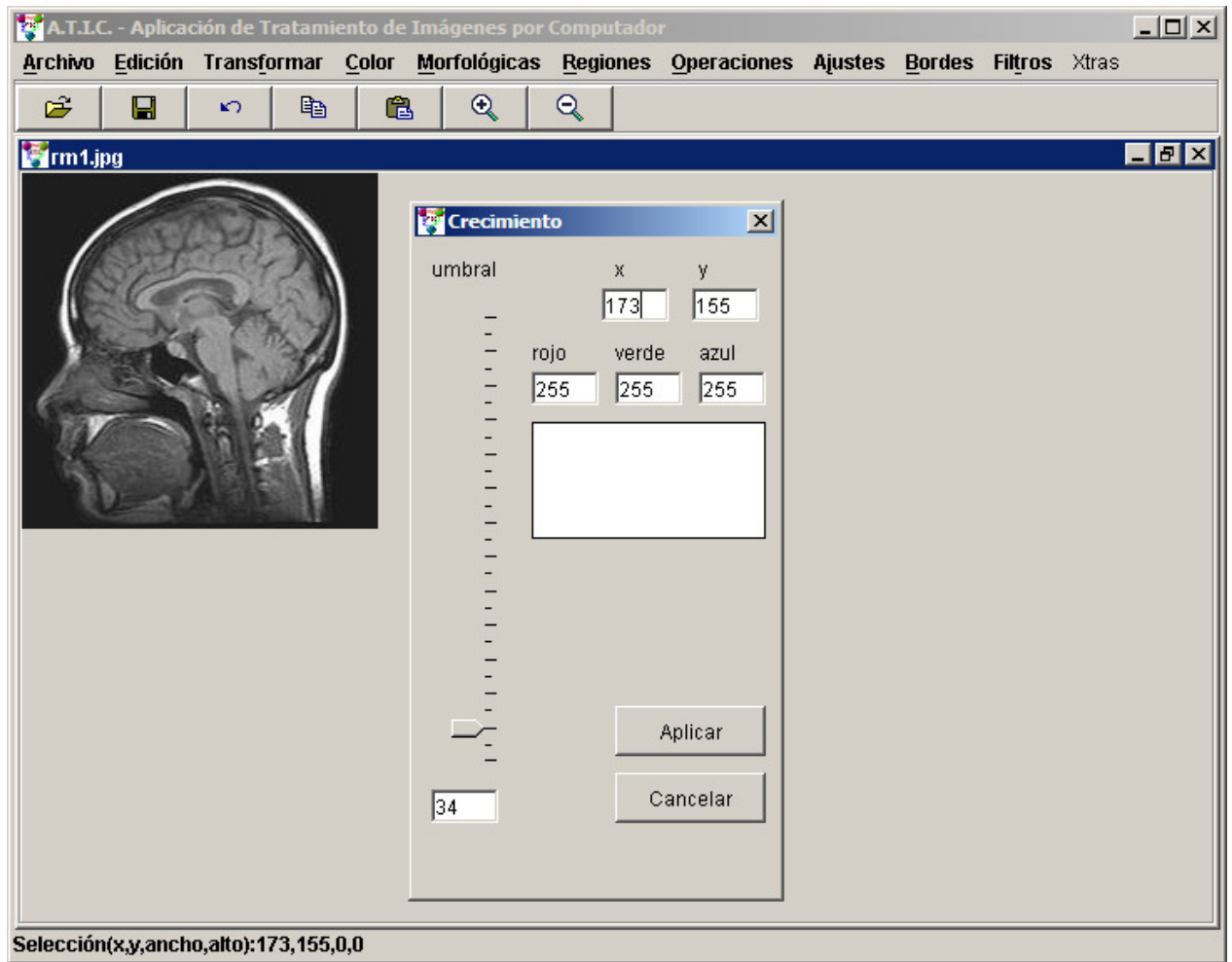


figura 27: crecimiento de regiones mediante adición de píxeles

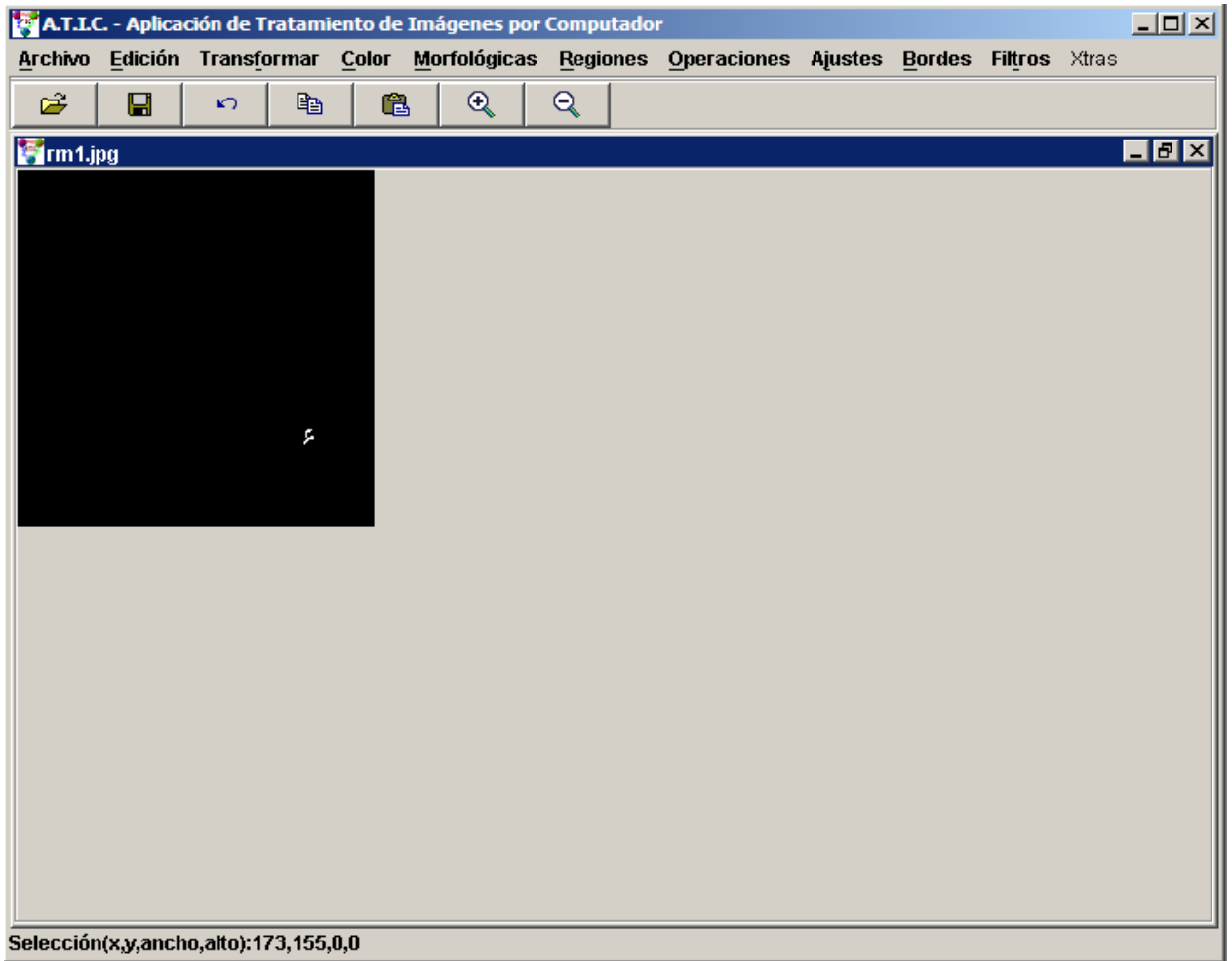


figura 28: crecimiento de regiones mediante adición de píxeles empleando umbral 0

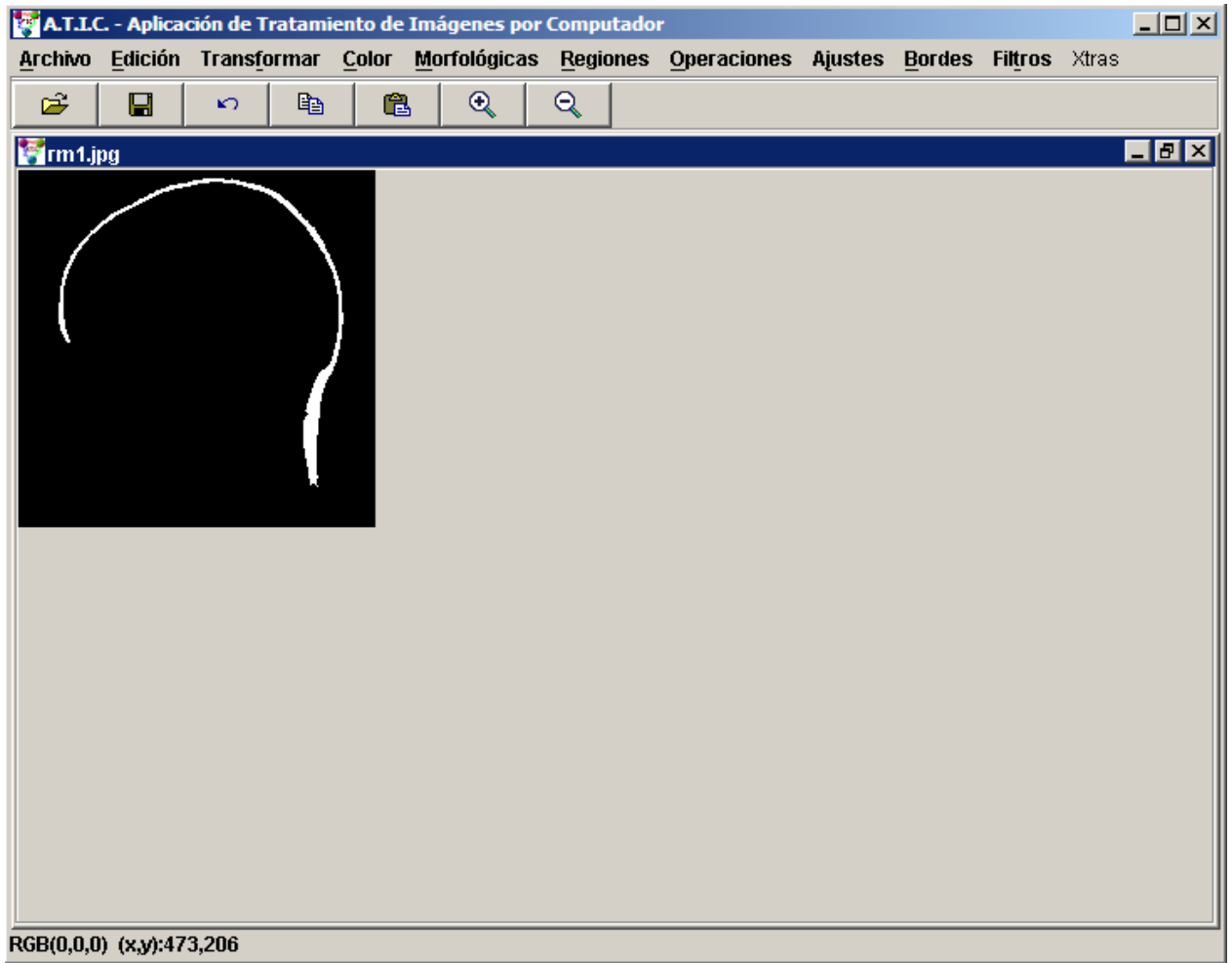


figura 29: crecimiento de regiones mediante adición de píxeles empleando umbral 64

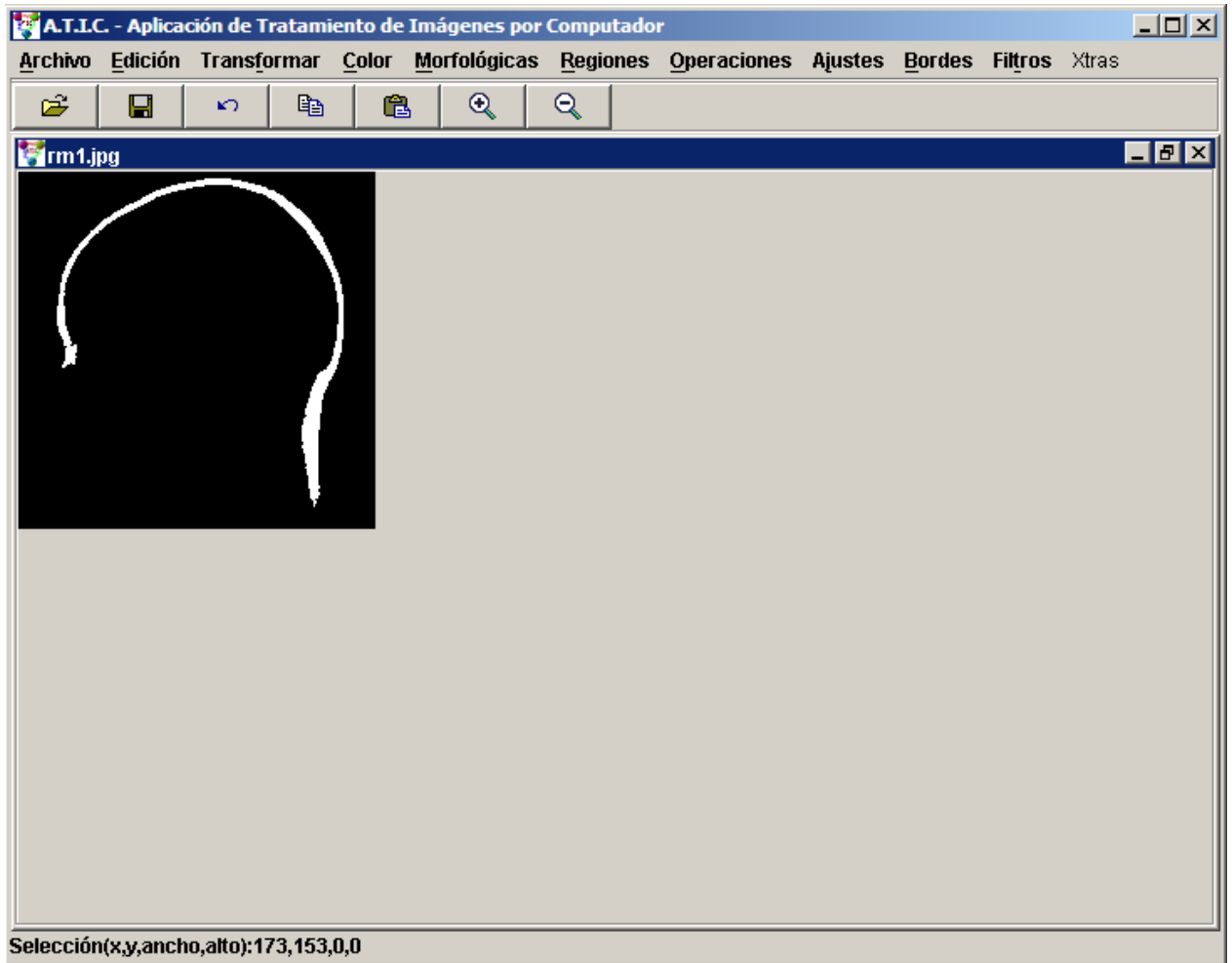


figura 30: crecimiento de regiones mediante adición de píxeles empleando umbral 128

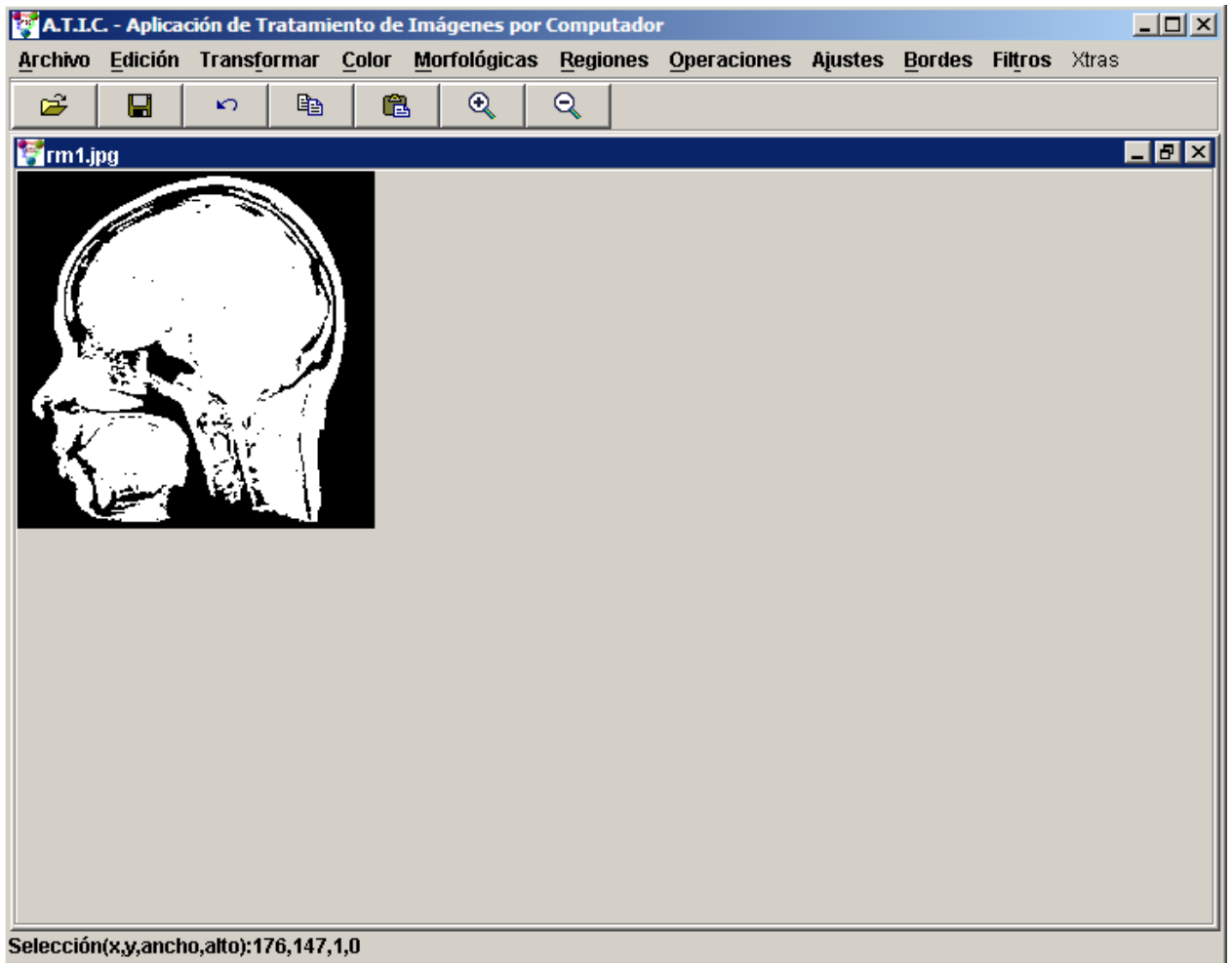


figura 31: crecimiento de regiones mediante adición de píxeles empleando umbral 192

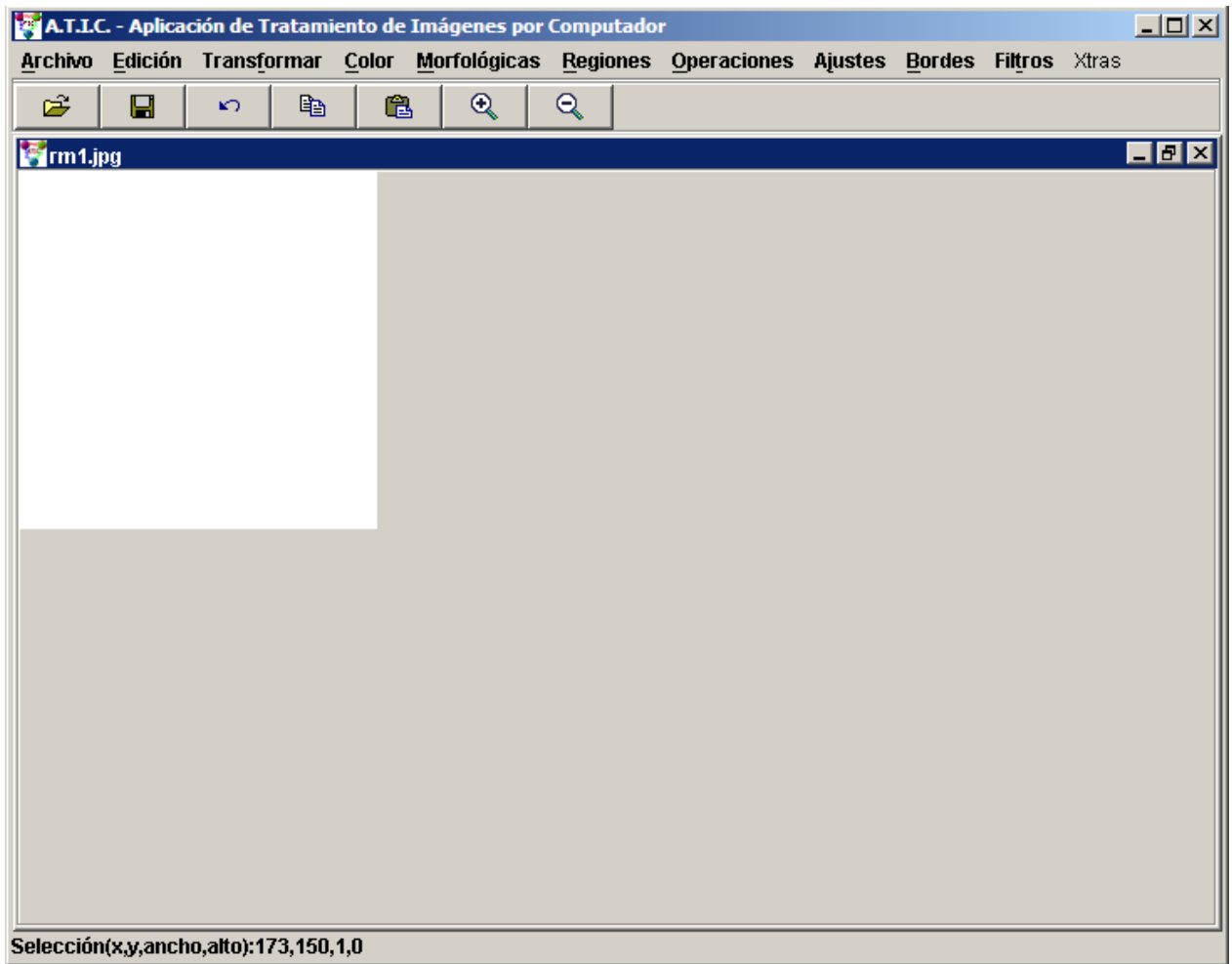


figura 32: crecimiento de regiones mediante adición de píxeles empleando umbral 255

La lógica de este algoritmo es la siguiente:

Un píxel $p = (R_p, G_p, B_p)$ pertenece a la región generada por la semilla $s = (R_s, G_s, B_s)$ si la distancia euclídea entre esos dos vales está por debajo del umbral elegido T . Es decir, se cumple:

$$d(p,s) = \sqrt{(R_p - R_s)^2 + (G_p - G_s)^2 + (B_p - B_s)^2} \leq T$$

Si las imágenes son en grises, las tres bandas de color tiene el mismo valor de intensidad para un píxel (x,y)

Notar que al igual que en el caso del algoritmo de segmentación por umbral, cuando el umbral se acerca a los límites superior e inferior se genera una única región.

15.4 División y mezcla

- Técnica top-down contextual.
- Al igual que la técnica de similitud de regiones, utiliza un predicado.
- Inicialmente la imagen se considera como una única región y el predicado es falso, entonces se divide la región en subregiones.
- Las nuevas regiones obtenidas a partir de otras son nuevamente partidas y mezcladas hasta cumplir con el criterio de uniformidad.
- El proceso se repite hasta que todas las regiones sean uniformes o hasta alcanzar un número máximo de particiones preestablecido.

15.5 Comparación de resultados

El problema de cómo comparar resultados

Para cuantificar el rendimiento de un método de segmentación, es necesario llevar a cabo experimentos de validación, lo cual generalmente consiste en comparar el resultado obtenido contra algún modelo real. El método más directo para validar es comparar la segmentación automática con una segmentación obtenida manualmente, pero este método no garantiza un modelo real perfecto debido a que el rendimiento de un usuario también puede ser deficiente.

De cualquier forma, una vez que el modelo real esté disponible, se debe establecer alguna medida de mérito para cuantificar la precisión y exactitud. La escogencia de dicha medida de mérito es dependiente de la aplicación y puede basarse en información de la región como el número de píxeles que no se clasificaron, o información de los bordes como la distancia al borde real.

No existe una forma estándar de comparar los algoritmos de segmentación. Los criterios de evaluación suelen ser altamente subjetivos, sin embargo, el factor tiempo de ejecución suele ser el comparador más usado.

Durante las últimas décadas se han desarrollado diversos algoritmos de segmentación, apareciendo nuevos algoritmo año tras año.

Sin embargo poco esfuerzo se ha aplicada a intentar evaluar la efectividad de dichos algoritmos. Normalmente, la efectividad de un algoritmo queda demostrada presentando diversas imágenes segmentadas.

Esto permite llegar a conclusiones subjetivas sobre el algoritmo.

La dificultad para evaluar algoritmos de segmentación radica en la naturaleza misma del problema. Recordemos la definición de segmentación:

"La segmentación de imágenes consiste en subdividir la imagen en las partes que la constituyen y extraer los objetos de interés"

Esto abarca la calidad procedural del algoritmo de segmentación pero deja afuera muchas preguntas, sobre todo cómo definir "partes constituyentes" y "partes de interés" y cómo medir el grado de eficiencia de un algoritmo en caso de no ser óptimo.

Inevitablemente estas preguntas deberán ser respondidas de forma dependiente de la aplicación usada y quedarán relacionadas a las etapas del proceso a ser aplicadas a los resultados de la segmentación para lograr el objetivo.

Hoy por hoy no existen sistemas para comparar diferentes algoritmos de segmentación, debido a su costo, y debido a que las propiedades del algoritmo determinan las etapas del proceso de segmentación que se deben tomar. En este punto es donde nos interesa evaluar algoritmos de segmentación sin implementar sus etapas de procesamiento.

En [4] Zhang propone para clasificar los algoritmos de segmentación métodos "analíticos", "empíricos con métrica de certeza" y de "discrepancia analítica".

Métodos analíticos:

- Los métodos analíticos intentan caracterizar al algoritmo en términos de principios, requerimientos, complejidades, etc. sin referirse concretamente a una implementación del algoritmo o a un conjunto de datos de prueba. Por ejemplo: podemos definir la complejidad del tiempo del algoritmo, o su respuesta a un modelo teórico de datos como *step edge*. Mientras que en dominios como la detección de bordes puede resultar útil, en general la falta de una teoría de segmentación limita estos métodos.

Métodos empíricos con métrica de certeza:

- Evalúan el algoritmo computando una métrica de “certeza” bien conocida sobre la imagen segmentada sin un conocimiento a priori del resultado de segmentación deseado. Por ejemplo: en imágenes con bajo nivel de gris una métrica podría ser la medida de nivel de grises. La ventaja de este método es que sólo requiere que el usuario defina su métrica de certeza, no requiere de imágenes segmentadas manualmente como base de conocimiento y puede ser empleado en aplicaciones en línea de modo que la efectividad del algoritmo pueda ser monitoreada durante la ejecución. La desventaja de los métodos empíricos de certeza es que son heurísticos y pueden estar fuertemente ligados a ciertos tipos de algoritmos. Por ejemplo: la métrica de nivel de grises causará que cualquier algoritmo de segmentación que forma regiones de texturas uniformes sea evaluado de forma negativa.

Métodos empíricos de discrepancia:

- Calculan una media de discrepancia entre la imagen segmentada producida por la segmentación de un algoritmo y la correcta segmentación deseada para la imagen de entrada. En el caso de imágenes sintéticas, la segmentación correcta puede ser obtenida automáticamente por un proceso de generación de imágenes. Mientras que en el caso de imágenes reales debe ser producida manualmente o semi-automáticamente por un operador experto. Esta es la desventaja principal, ya que la creación manual de imágenes demanda mucho tiempo. Análogamente al método empírico con métrica de certeza, también debe definirse una métrica de discrepancia de forma explícita. Una métrica de discrepancia sería por ejemplo calcular la distancia entre píxeles mal segmentados y píxeles cercanos bien segmentados.

16 Solución propuesta

La solución propuesta consiste en implementar dos versiones del algoritmo de segmentación de imágenes por crecimiento de regiones, también conocido como método semilla, para luego comparar los resultados. De ahora en adelante nos referiremos a estos dos programas como **clásica** y **mejorada**. Como sus nombres lo indican, clásica se trata de una versión clásica del algoritmo de segmentación por crecimiento de regiones mientras que mejorada es una versión optimizada del mismo, que introduce algunos cambios.

Ambas implementaciones fueron realizadas en plataforma JAVA, siendo la versión **clásica** basada en JFrames (aplicación stand-alone o de escritorio) y la versión **mejorada** embebida en plataforma web (applet).

La diferencia funcional radica en que **clásica** debe ser ejecutada desde una computadora como si fuera un programa de escritorio, mientras que **mejorada** se ejecuta desde cualquier navegador que soporte Java. Naturalmente, el equipo donde corran las aplicaciones debe tener instalada y configurada la plataforma Java. No siendo esto un obstáculo dado que hoy en día todos los equipos PC compatibles ya vienen con la plataforma Java instalada.

Otra diferencia funcional importante es que mejorada, al estar basada en plataforma web, se puede ejecutar desde cualquier computadora con acceso a Internet dado que el procesamiento es en línea. Esto presenta una ventaja importante que es poder *compartir el servicio de segmentación en una red*.

El cuadro a continuación muestra las diferencias entre ambas versiones del algoritmo.

Versión clásica	Versión mejorada
Implementado en plataforma JAVA como una aplicación de escritorio (JFrame).	Implementado en plataforma JAVA como una aplicación basada en web (Applet).
Requiere configuración extra para ejecutarse (configuración de los parámetros de ejecución como por ejemplo la ruta donde se encuentran los archivos con el código fuente del programa.).	No requiere configuración extra para ejecutarse.
No requiere acceso a un navegador para correr	Requiere estar conectado a Internet para correr ó tener instalado un servidor web con la aplicación

Es una versión clásica del algoritmo de segmentación por crecimiento de regiones.	Es una versión mejorada del algoritmo de segmentación por crecimiento de regiones. Introduce algunas modificaciones al algoritmo que lo hacen ser más veloz.
Configuración del algoritmo, soporta: <ul style="list-style-type: none"> • Selección del método de conectividad (4 vecinos u 8 vecinos) • Selección del nivel de umbral (va de 0 a 255) 	Configuración del algoritmo, soporta: <ul style="list-style-type: none"> • Selección del método de conectividad (4 vecinos u 8 vecinos) • Selección del nivel de umbral (va de 0 a 255)
Formatos de imágenes soportados: <ul style="list-style-type: none"> • BMP • PNG • JPEG • TIFF • GIF 	Formatos de imágenes soportados: <ul style="list-style-type: none"> • PNG • JPEG • TIFF • GIF • DICOM (DCM)

Tabla 3: versión clásica VS versión mejorada

16.1 Método de segmentación elegido

El método de segmentación elegido fue el de crecimiento de regiones. Este método consiste en seleccionar un conjunto de puntos iniciales llamados semilla y hacer crecer regiones conectadas uniformes a partir de las semillas. Un píxel se agrega a la región si y sólo si cumple con las siguientes condiciones:

- No ha sido asignado aún a otra región
- Es un vecino de la región
- La región obtenida luego de agregar el nuevo píxel sigue siendo uniforme

Versión clásica del algoritmo de segmentación por crecimiento de regiones

El algoritmo de crecimiento de regiones clásico se basa en un predicado de uniformidad que debe mantenerse en todas las regiones. Este predicado establece que una región es uniforme si y sólo si los píxeles pertenecientes a dicha región son similares en términos de niveles de gris, color u otra propiedad.

El predicado de uniformidad se puede representar con la siguiente fórmula:

$$P(R) = \begin{cases} \text{verdadero si } |f(j, k) - f(m, n)| \leq \Delta \\ \text{falso en caso contrario} \end{cases}$$

Donde (j,k) y (m,n) son coordenadas de píxeles vecinos en la región R. Este predicado dice que una región R es uniforme si y sólo si la diferencia en los niveles de gris de cualquier par de píxeles vecinos en R no excede Δ .

Sin embargo este predicado tiene un problema, restringe las variaciones en niveles de gris en una región a un rango de ancho Δ . Pequeños cambios en los niveles de gris entre píxeles vecinos que satisfagan el predicado pueden ser acumulados produciendo una gran diferencia en niveles de gris entre lados opuestos de una región larga.

Entonces modificaremos el predicado teniendo en cuenta la media del nivel de gris de los píxeles en la región R, la fórmula se presenta a continuación:

$$P(R) = \begin{cases} \text{verdadero si } |f(j, k) - \mu_R| \leq \Delta \\ \text{falso en caso contrario} \end{cases}$$

Donde $f(j,k)$ es el nivel de gris de un píxel en la región R con coordenadas (i,j) y μ_R es la media del nivel de gris de todos los píxeles en R excepto el píxel ubicado en (j,k).

Sea f la imagen de la cual se pretende hacer crecer regiones.

Definimos un conjunto de regiones, R_1, R_2, \dots, R_n cada una conteniendo una semilla simple.

Repetir

Para $i = 1$ a n hacer

Para cada píxel, p , en el borde de R_i hacer

Para todos los vecinos de p hacer

Sean x, y las coordenadas vecinas

Sea μ_i la media del nivel de gris en R_i

Si el vecino no esta asignado y $|f(x, y) - \mu_i| \leq \Delta$ luego
Agregar vecino a R_i
Actualizar μ_i
Fin Si
Fin Para
Fin Para
Fin Para
Hasta que no existan mas píxeles para asignar a regiones

Como vemos, se trata de un algoritmo iterativo de orden de ejecución n^3 .

Ejemplo:

Empleando el software “Region Growing Tool” de MC Soft vamos a segmentar una imagen de resonancia magnética en escala de grises.

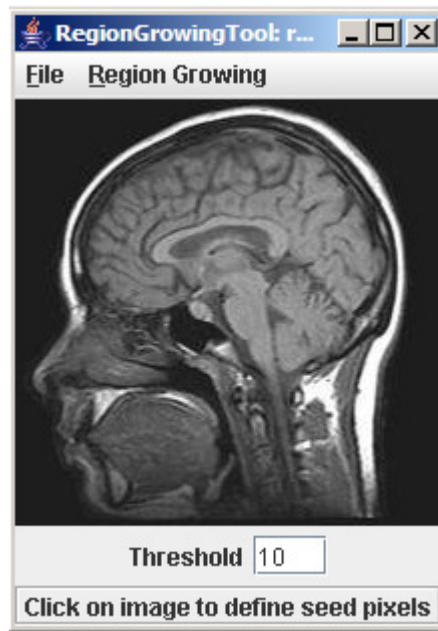


figura 33: imagen a segmentar

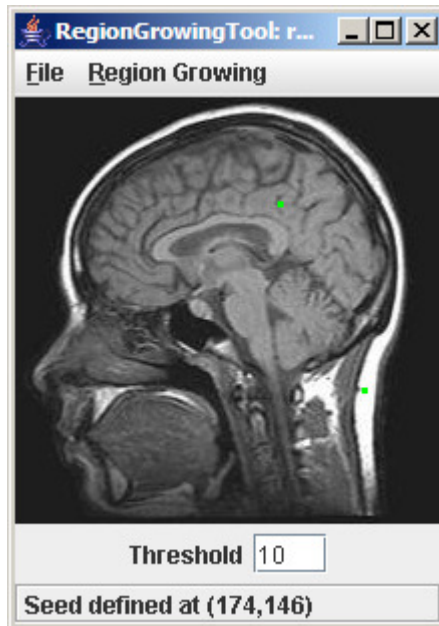


figura 34: colocando las semillas

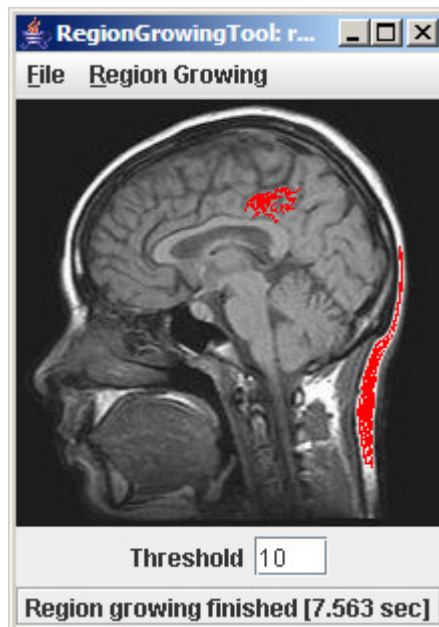


figura 35: segmentando con conectividad 8-vecinos, el tiempo de ejecución fue de 7.5 segundos

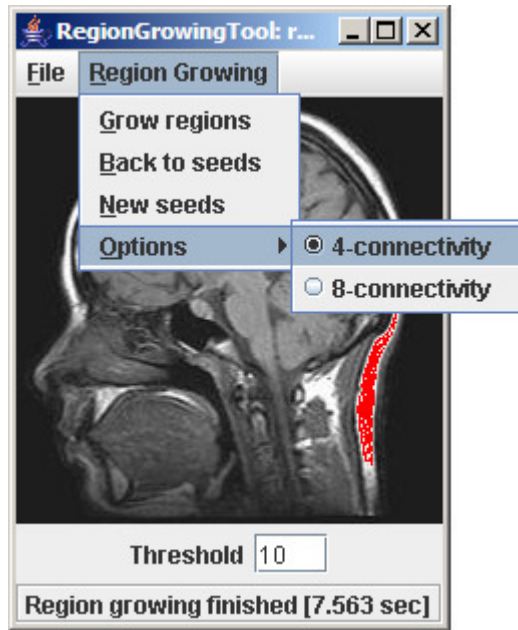


figura 36: configurando el programa para emplear 4-conectividad en lugar de 8-conectividad

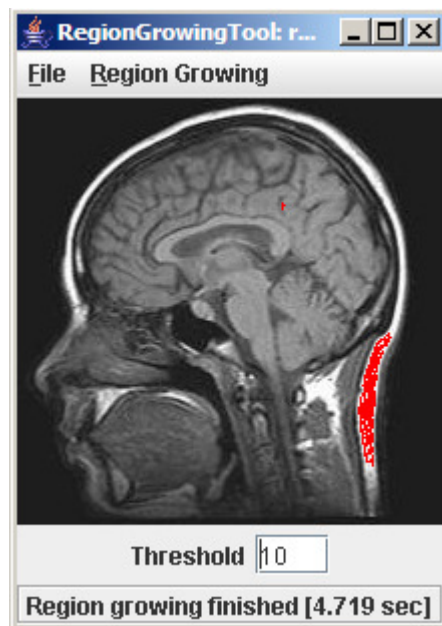


figura 36: el tiempo de ejecución fue de 4.7 segundos

Comparando los resultados obtenidos con el programa vemos a simple vista que la conectividad hace variar los resultados. La 8-conectividad cubre más la región y también demora más tiempo.

16.2 Mejoras realizadas al método de segmentación

La idea principal es extender el algoritmo clásico de segmentación de imágenes por crecimiento de regiones, que intenta aislar y extraer componentes individuales de imágenes médicas.

Esto es una parte importante del estudio de volumetría en imágenes médicas, donde una vez que se reconoce una forma se miden varias cantidades como por ejemplo el tamaño de un tumor o las paredes del corazón. Realizar las mediciones a mano es una tarea que consume mucho tiempo. En su lugar, se emplean técnicas de segmentación que encuentran las regiones de forma automática.

Supongamos que tenemos la siguiente imagen de una resonancia magnética.



figura 37: resonancia magnética

Supongamos que queremos extraer un objeto de interés dentro de la imagen, en este caso, el cerebelo. Una forma de realizar esto sería mirando en los lugares donde hay un gran salto de intensidades entre los píxeles vecinos, *esto es lo que hace la versión clásica del algoritmo de segmentación de imágenes mencionado anteriormente.*

Sin embargo, es difícil elegir un valor adecuado como referencia para los cambios de intensidad: si es demasiado pequeño se sobredimensiona al objeto y si es demasiado grande se dejan afuera partes del objeto.

Otro problema con la versión clásica del algoritmo de segmentación de imágenes médicas basado en crecimiento de regiones es que puede tener problemas si la imagen contiene mucho ruido.

En la versión mejorada del algoritmo se emplea la técnica de *contornos activos*^[3], combinada con la técnica clásica de semillas. La técnica de contornos activos consiste en colocar una pequeña semilla circular en una región de interés y hacerla crecer hacia afuera de la región hasta alcanzar los límites de la misma.

Empleando el método de *fast marching*^[1] para propagar la semilla inicial hacia afuera, seguida del *level set method*^[2] para ajustar los resultados. La idea principal radica en hacer crecer las curvas hacia fuera dejando de lado los píxeles internos de la región:

- Cuando la curva pasa por lugares donde el gradiente de la imagen (o sea, el cambio de valor entre un píxel y sus vecinos) es pequeño, hacemos crecer rápido a la curva.
- Cuando la curva pasa sobre lugares donde el gradiente de la imagen es largo, se espera que los bordes estén cerca, y hacemos crecer lenta a la curva.
- La versión mejorada no tiene en cuenta los píxeles internos de la región, sólo intenta detectar los bordes de la misma.

Comparación de resultados

A continuación, se comparan ambas versiones del algoritmo. Se incluyen pantallas demostrativas.

Para la comparación se tuvo en cuenta la siguiente imagen que corresponde a la hemicara obtenida por medio de resonancia magnética. La imagen está en formato PNG y tiene 8 bits de resolución. Midiendo 426 x 426 píxeles.

El objetivo es segmentar la imagen de forma tal de separar la región que corresponde a hueso craneal del resto de la imagen.

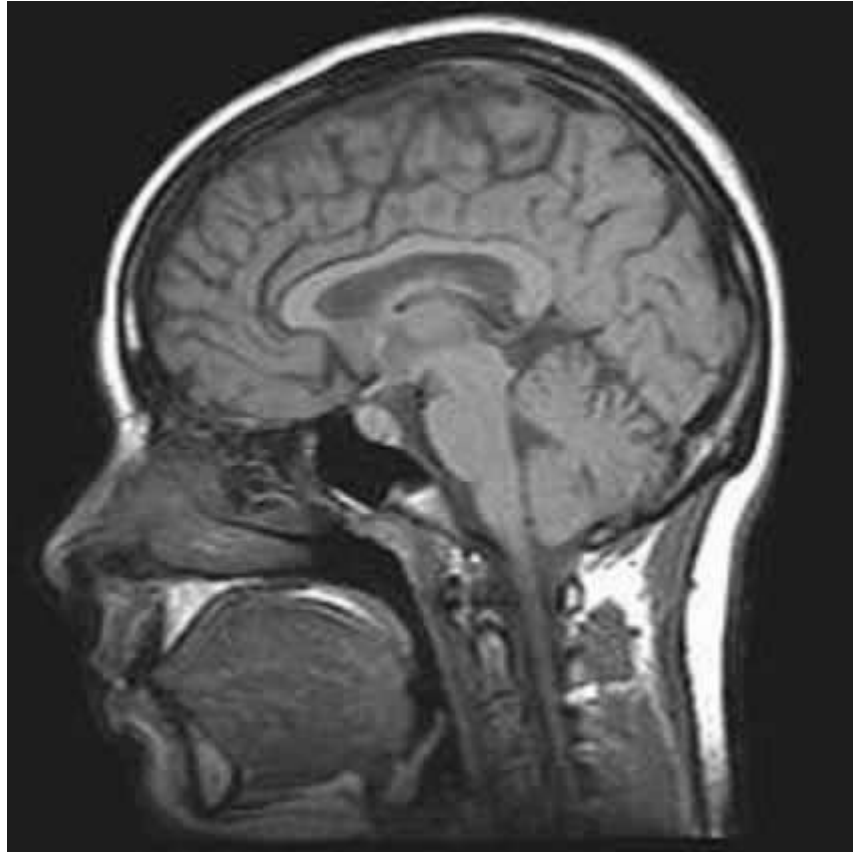


figura 38: resonancia magnética de hemicara derecha: imagen empleada para la comparación de los algoritmos. En este ejemplo se pretende segmentar la imagen separando la región correspondiente a hueso craneal del resto de la imagen.

Pasos a seguir para segmentar la imagen empleando el algoritmo clásico

Paso 1, versión clásica:

Iniciamos el programa (FramePpal.java), vamos luego al menú “Archivo->Abrir” y seleccionamos la imagen guardada en disco.

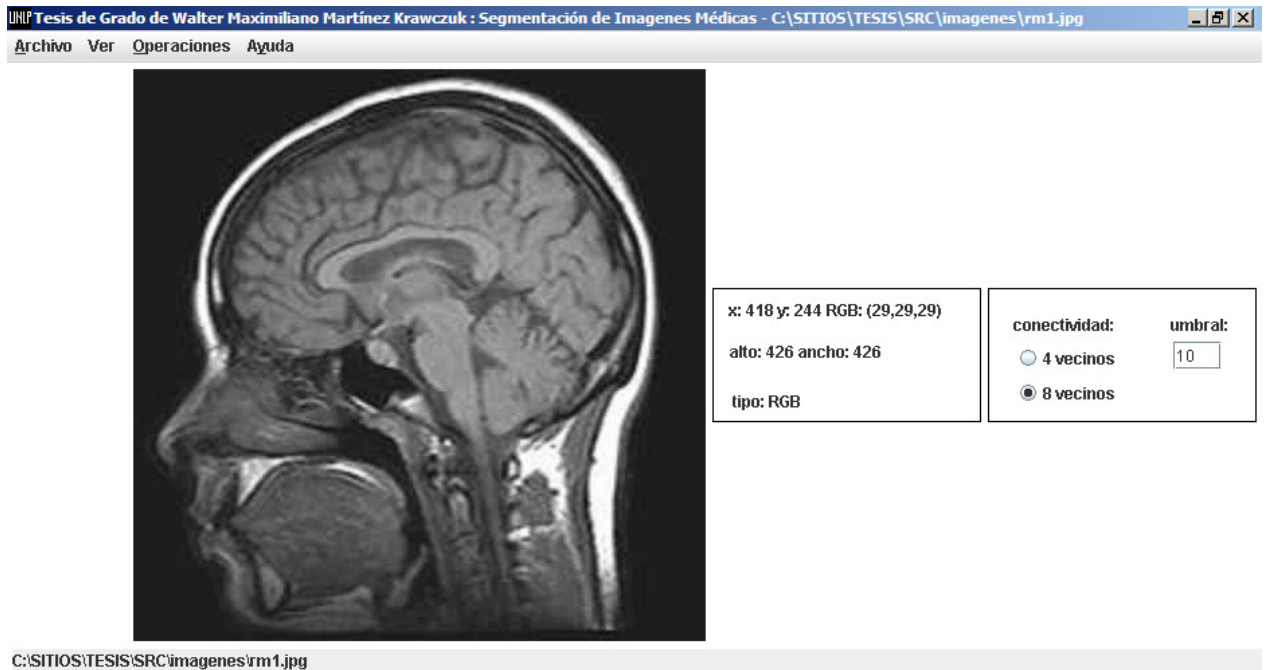


figura 39: paso 1, clásica: seleccionamos la imagen a segmentar

Paso 2, versión clásica:

Colocamos las semillas presionando el botón izquierdo del ratón sobre la posición deseada en la imagen (las semillas aparecen resaltadas en color verde, en un recuadro de 2 x 2 píxeles) y configuramos la conectividad y el umbral. Para las pruebas se empleó conectividad 8-vecinos y un umbral de 55. En la pantalla a continuación se resaltaron las semillas en círculos rojos sobre la imagen. En este ejemplo se emplearon dos semillas.

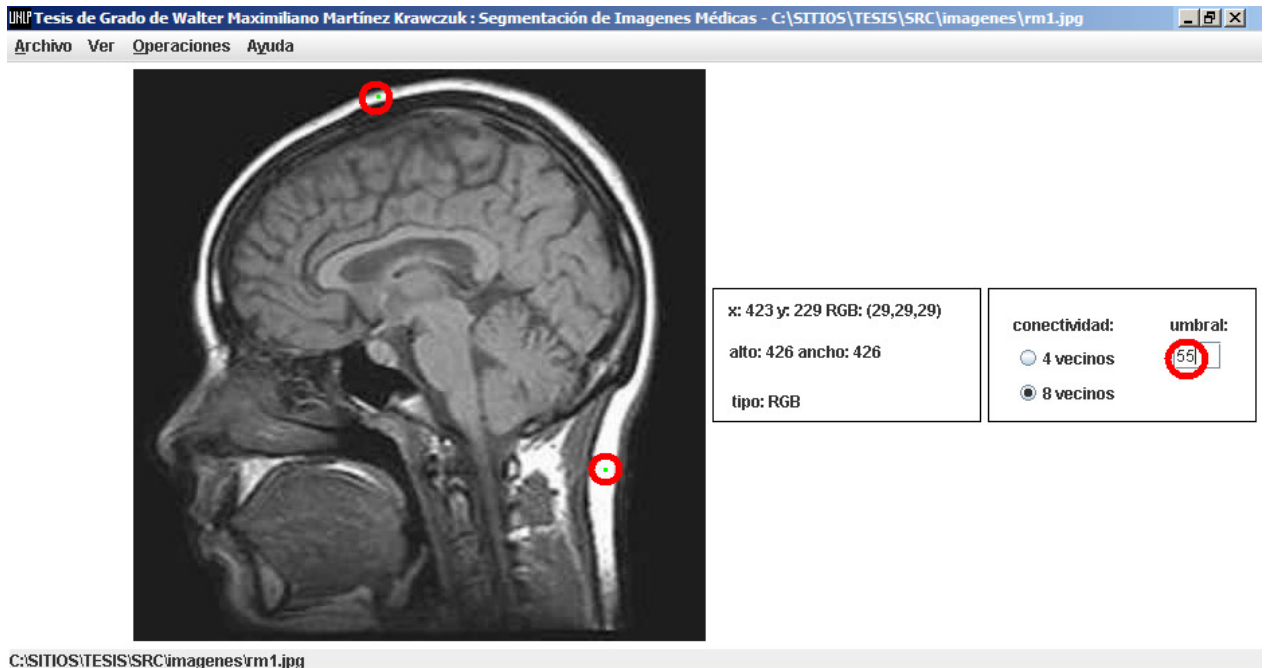


figura 40: paso 2, clásica: colocamos las semillas y configuramos la conectividad y el umbral

Paso 3, versión clásica:

Iniciamos el proceso de segmentación eligiendo la opción de menú “Operaciones->Segmentar->Segmentar”. Observamos que la imagen se va pintando de rojo en las regiones donde se colocaron semillas. Hasta colorear toda la región en rojo.

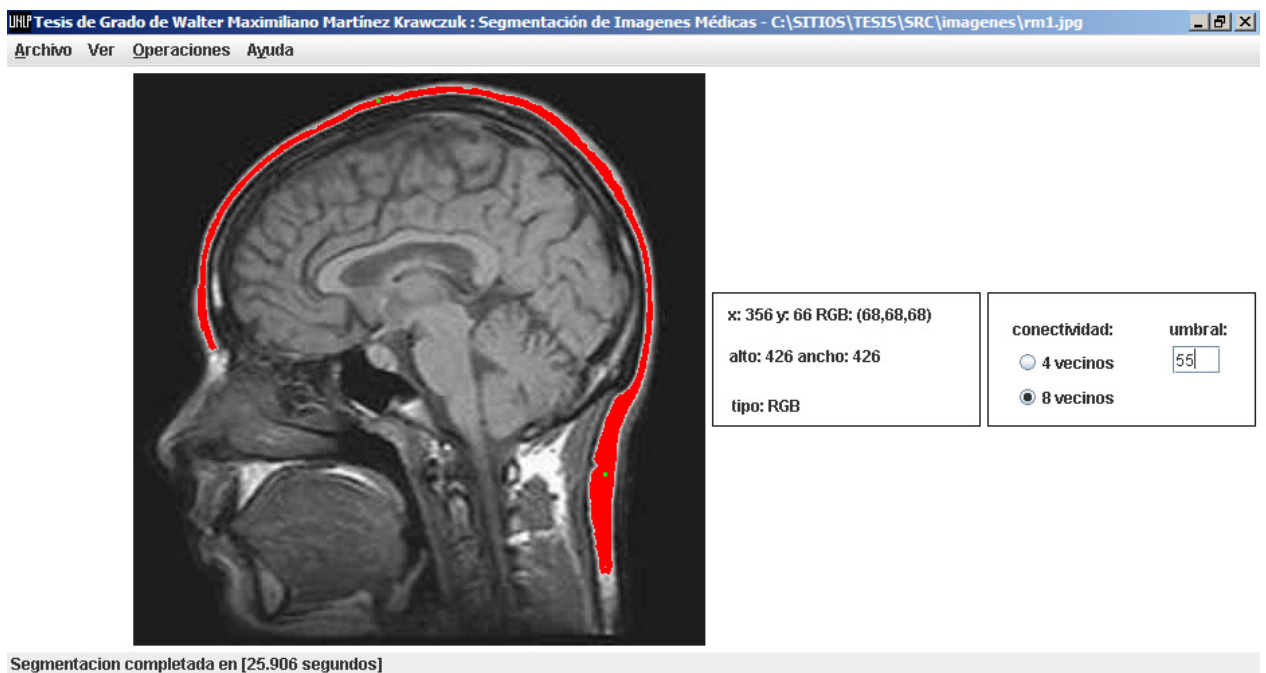


figura 41: paso 3, clásica: iniciamos el proceso seleccionando Operaciones->Segmentar->Segmentar

Notamos al finalizar que en la barra de estado, justo debajo de la imagen, aparece el tiempo que demoró el proceso de segmentación. En este ejemplo, el proceso de segmentación de la versión clásica del algoritmo demoró **25.9 segundos**.

Pasos a seguir para segmentar la imagen empleando la versión mejorada del algoritmo.

Paso 1, versión mejorada:

Iniciamos nuestro navegador web preferido y lo apuntamos al sitio web donde se encuentra alojada la versión mejorada del algoritmo de segmentación, para las pruebas se utilizó la dirección de servidor local o localhost (127.0.0.1)

Información

presione para seleccionar

alto	ancho
426	426
tipo	resolución
JPG	8 bits

Configuración

conectividad	umbral
<input checked="" type="radio"/> 4 vecinos	10
<input type="radio"/> 8 vecinos	

Ayuda

- presione "examinar" y seleccione una imagen
- configure el algoritmo usando el panel de configuración
- haga clic sobre la imagen, vea los resultados

[inicio](#)

Universidad Nacional de La Plata - Facultad de Informatica
Tesis de Grado de Walter Maximiliano Martinez Krawczuk
Segmentacion de Imagenes Medicas
versión mejorada del algoritmo de segmentación

segmentar

figura 42: paso 1, mejorada: apuntamos el navegador al sitio web donde esta alojado el sistema y luego cargamos una imagen guardada en disco.

Paso 2, versión mejorada:

Colocamos las semillas presionando el botón izquierdo del ratón sobre la posición deseada en la imagen (las semillas aparecen resaltadas en color verde, en un recuadro de 2 x 2 píxeles) y configuramos la conectividad y el umbral. Para las pruebas se empleó conectividad 8-vecinos y un umbral de 55. En la pantalla a continuación se resaltaron las semillas en círculos rojos sobre la imagen. En este ejemplo se emplearon dos semillas.

Información
presione para seleccionar

alto	ancho
426	426
tipo	resolución
JPG	8 bits

Configuración

conectividad	umbral
<input type="radio"/> 4 vecinos	<input type="text" value="55"/>
<input checked="" type="radio"/> 8 vecinos	

Ayuda

- presione "examinar" y seleccione una imagen
- configure el algoritmo usando el panel de configuración
- haga clic sobre la imagen, vea los resultados

[inicio](#)

Universidad Nacional de La Plata - Facultad de Informática
Tesis de Grado de Walter Maximiliano Martínez Krawczuk
Segmentación de Imágenes Médicas
versión mejorada del algoritmo de segmentación



figura 43: paso 2, mejorada: colocamos las semillas y seleccionamos el nivel de umbral y la conectividad.

Paso 3, versión mejorada:

Una vez colocadas las semillas presionamos el botón “segmentar”.

Universidad Nacional de La Plata - Facultad de Informatica
Tesis de Grado de Walter Maximiliano Martinez Krawczuk
Segmentacion de Imagenes Medicas
versión mejorada del algoritmo de segmentación

Información

presione para seleccionar

alto	ancho
<input type="text" value="426"/>	<input type="text" value="426"/>
tipo	resolución
<input type="text" value="JPG"/>	<input type="text" value="8 bits"/>

Configuración

conectividad umbral

4 vecinos

8 vecinos

Ayuda

- presione "examinar" y seleccione una imagen
- configure el algoritmo usando el panel de configuración
- haga clic sobre la imagen, vea los resultados

[inicio](#)

segmentar



Segmentación completada en 0.328 segundos

figura 44: paso 3: resultado de segmentar la imagen empleando la versión mejorada del algoritmo de segmentación

Al finalizar notamos debajo de la imagen una leyenda que indica el tiempo que demoró el proceso de segmentación de la versión mejorada del algoritmo. En este caso el proceso demoró **0.32 segundos**.

Al comparar los resultados obtenidos por ambos algoritmos, notamos que la versión clásica tiende a tapar o rellenar completamente las regiones mientras que la versión mejorada deja huecos. Esto se debe a la implementación de los mismos y a la configuración ingresada al momento de iniciar el proceso.

A continuación se incluyen otras comparaciones.

Comparación de versiones:

Objetivo: identificar las regiones que corresponden a hueso en la imagen.


	<ul style="list-style-type: none">• resonancia magnética de rodilla izquierda.• 250 x 281 píxeles• formato JPG (RGB)• 8 bits por píxel
---	---

Tabla 4: comparación de versiones

Empleando la versión clásica



UNIF Tesis de Grado de Walter Maximiliano Martínez Krawczuk : Segmentación de Imágenes Médicas - C:\SITIOS\TESIS\SRC\imag... _ □ ×

Archivo Ver Operaciones Ayuda

x: 249 y: 34 RGB: (7,0,8)
alto: 250 ancho: 281
tipo: RGB

conectividad: umbral:
 4 vecinos
 8 vecinos

Segmentacion completada en 21.766 segundos

figura 45: resultado de haber segmentado la imagen con la versión clásica del algoritmo empleando conectividad 4-vecinos, 2 semillas y umbral 10. Notamos que el proceso dejó afuera gran parte de la región. El tiempo que demoró fue 21.76 segundos.

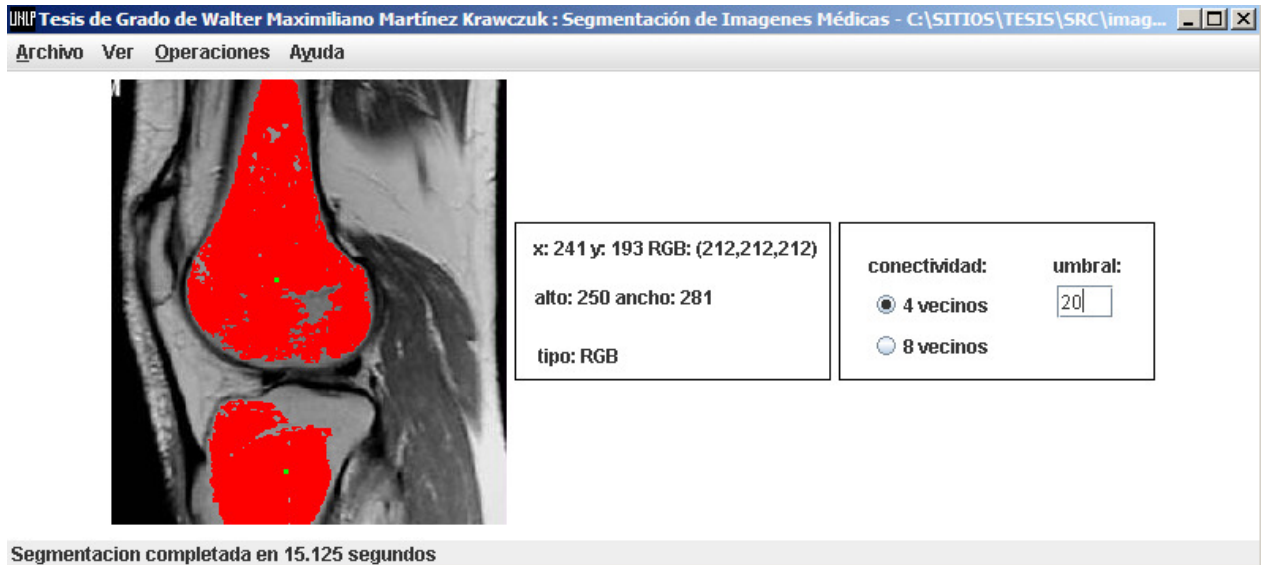


figura 46: resultado de haber segmentado la imagen con la versión clásica del algoritmo empleando conectividad 4-vecinos, 2 semillas y umbral 20. El proceso sigue sin poder cubrir toda la región. El tiempo que demoró fue 15.12 segundos.

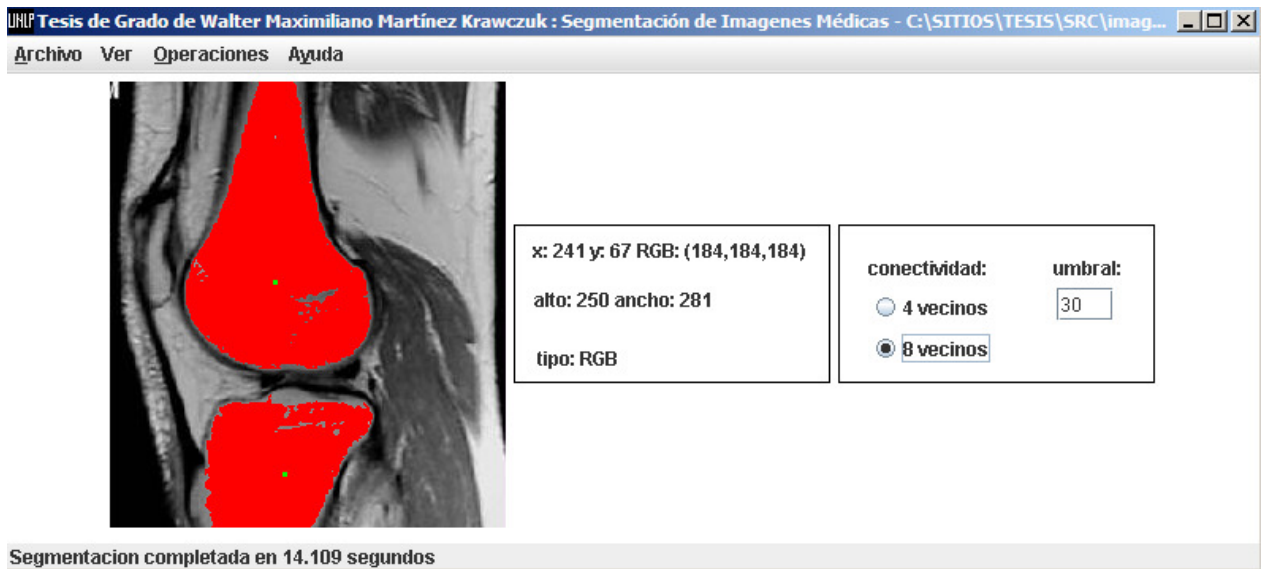


figura 47: resultado de haber segmentado la imagen con la versión clásica del algoritmo empleando conectividad 8-vecinos, 2 semillas y umbral 30. Se cubrió gran parte de la región pero continúan apareciendo "islas". El proceso demoró 14.10 segundos.

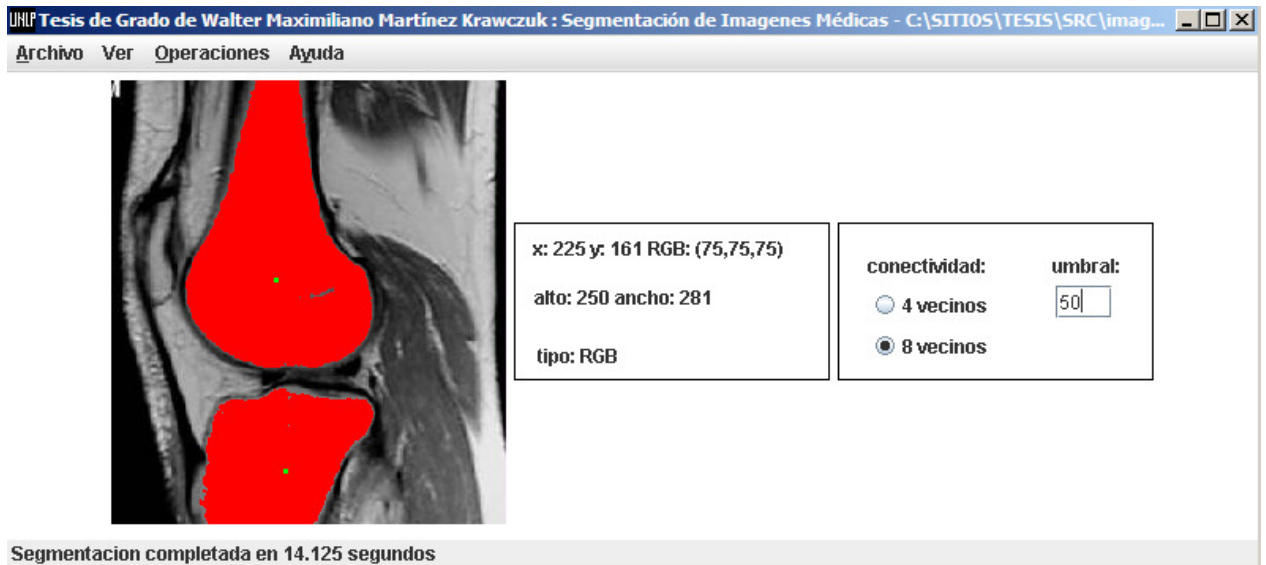


figura 48: resultado de haber segmentado la imagen con la versión clásica del algoritmo empleando conectividad 8-vecinos, 2 semillas y umbral 50. Se cubrió gran parte de la región y las “islas” prácticamente desaparecieron. El proceso demoró 14.12 segundos.

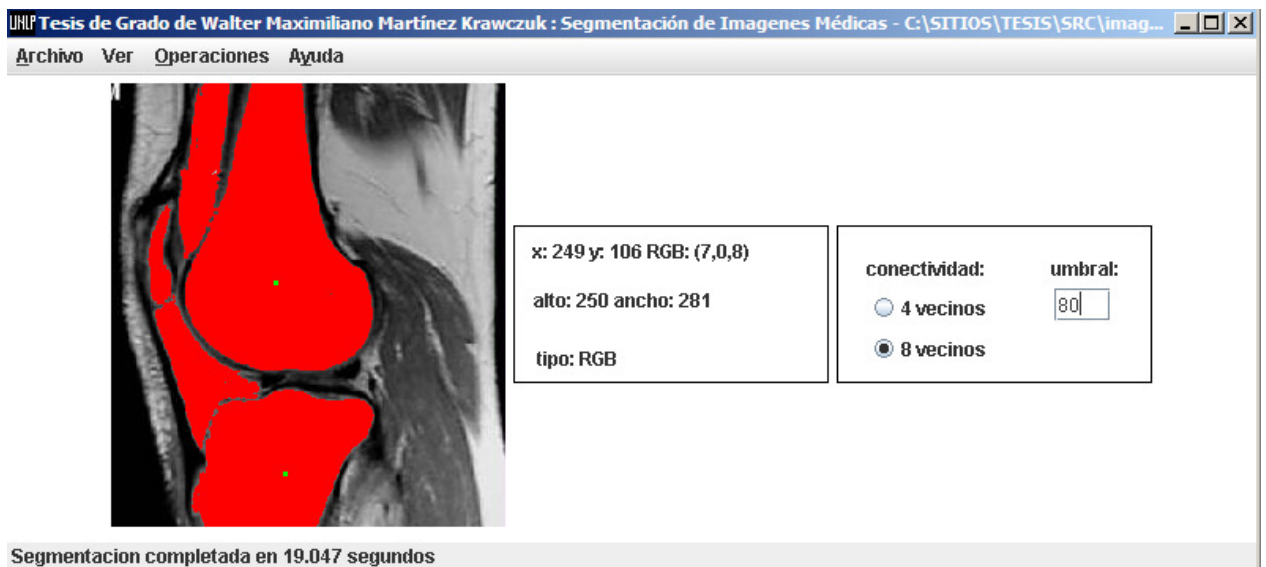


figura 49: resultado de haber segmentado la imagen con la versión clásica del algoritmo empleando conectividad 8-vecinos, 2 semillas y umbral 80. El proceso agregó información innecesaria (en la parte izquierda de la imagen). El tiempo de ejecución fue 19.04 segundos.

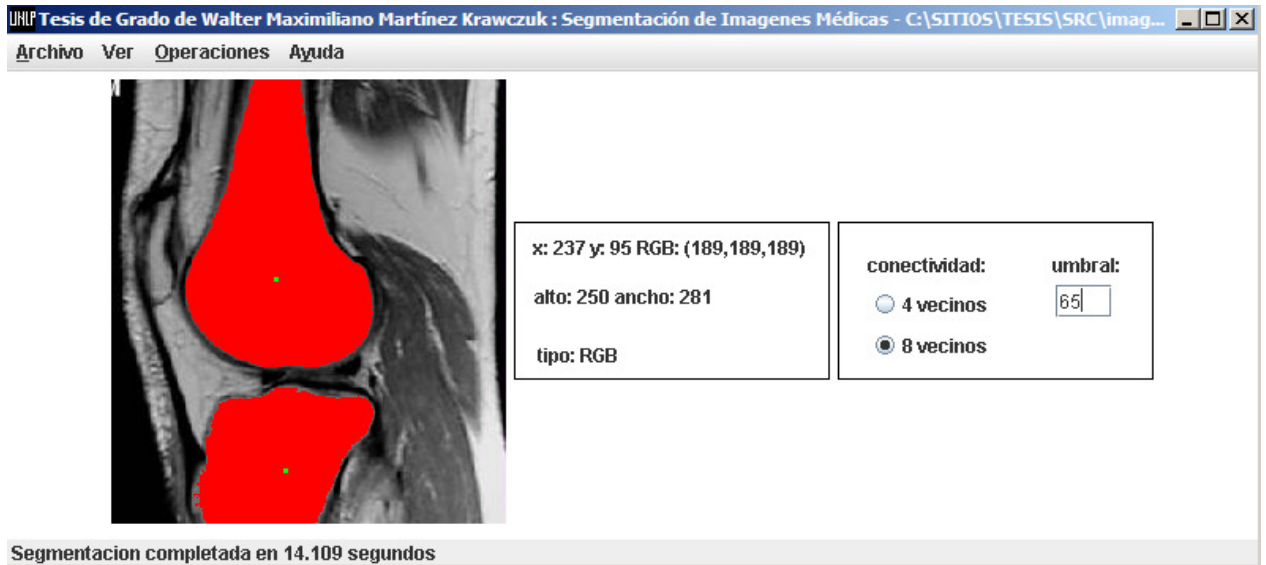


figura 50: resultado de haber segmentado la imagen con la versión clásica del algoritmo empleando conectividad 8-vecinos, 2 semillas y umbral 65. El resultado es el esperado. El proceso demoró 14.10 segundos.

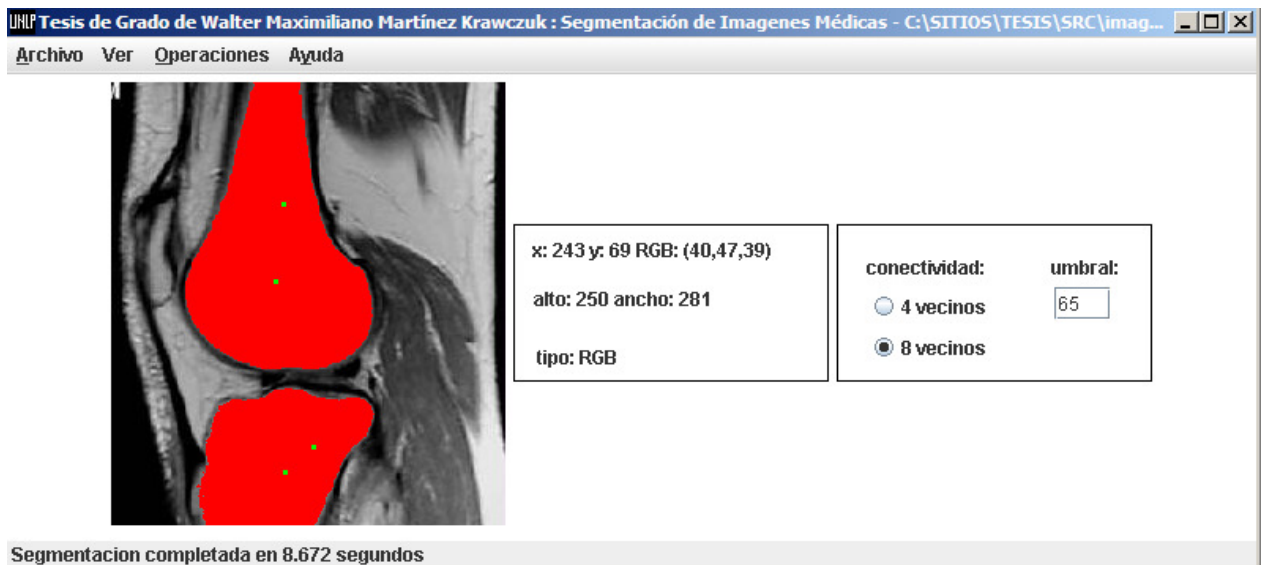


figura 51: resultado de haber segmentado la imagen con la versión clásica del algoritmo empleando conectividad 8-vecinos, 4 semillas y umbral 65. El resultado es el esperado. El proceso demoró 8.62 segundos.

Empleando la versión mejorada

Universidad Nacional de La Plata - Facultad de Informatica
Tesis de Grado de Walter Maximiliano Martinez Krawczuk
Segmentacion de Imagenes Medicas
versión mejorada del algoritmo de segmentación

Información

presione para seleccionar

alto	ancho
<input type="text" value="281"/>	<input type="text" value="281"/>
tipo	resolucion
<input type="text" value="JPG"/>	<input type="text" value="8 bits"/>


Configuración

conectividad	umbral
<input checked="" type="radio"/> 4 vecinos	<input type="text" value="10"/>
<input type="radio"/> 8 vecinos	

Ayuda

- presione "examinar" y seleccione una imagen
- configure el algoritmo usando el panel de configuración
- haga clic sobre la imagen, vea los resultados

[inicio](#)



Segmentación completada en 0.157 segundos

figura 52: resultado de haber segmentado la imagen con la versión mejorada del algoritmo empleando conectividad 4-vecinos, 2 semillas y umbral 10. Notamos que el proceso agregó información innecesaria (en la parte izquierda).

El tiempo que demoró fue 0.15 segundos.

Universidad Nacional de La Plata - Facultad de Informatica
Tesis de Grado de Walter Maximiliano Martinez Krawczuk
Segmentacion de Imagenes Medicas
versión mejorada del algoritmo de segmentación

Información

presione para seleccionar

alto	ancho
<input type="text" value="281"/>	<input type="text" value="281"/>
tipo	resolucion
<input type="text" value="JPG"/>	<input type="text" value="8 bits"/>

Configuración

conectividad	umbral
<input checked="" type="radio"/> 4 vecinos	<input type="text" value="20"/>
<input type="radio"/> 8 vecinos	

Ayuda

- presione "examinar" y seleccione una imagen
- configure el algoritmo usando el panel de configuración
- haga clic sobre la imagen, vea los resultados

[inicio](#)



Segmentación completada en 0.156 segundos

figura 53: resultado de haber segmentado la imagen con la versión mejorada del algoritmo empleando conectividad 4-vecinos, 2 semillas y umbral 20. El proceso sigue agregando información innecesaria (en la parte izquierda). El

tiempo de ejecución fue 0.15 segundos.

Información
presione para seleccionar

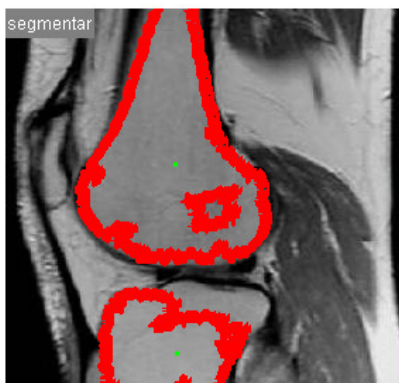
alto ancho
281 281
tipo resolución
JPG 8 bits

Configuración
conectividad umbral
 4 vecinos 30
 8 vecinos

Ayuda

- presione "examinar" y seleccione una imagen
- configure el algoritmo usando el panel de configuración
- haga clic sobre la imagen, vea los resultados

[inicio](#)



Segmentación completada en 0.156 segundos

figura 54: resultado de haber segmentado la imagen con la versión mejorada del algoritmo empleando conectividad 8-vecinos, 2 semillas y umbral 30. El proceso agregó islas o huecos dentro de la región. El tiempo de ejecución fue 0.15 segundos.

Información
presione para seleccionar

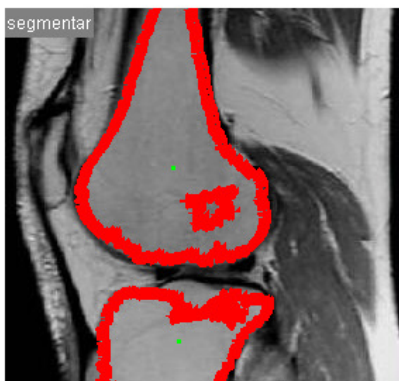
alto ancho
281 281
tipo resolución
JPG 8 bits

Configuración
conectividad umbral
 4 vecinos 40
 8 vecinos

Ayuda

- presione "examinar" y seleccione una imagen
- configure el algoritmo usando el panel de configuración
- haga clic sobre la imagen, vea los resultados

[inicio](#)



Segmentación completada en 0.156 segundos

figura 55: resultado de haber segmentado la imagen con la versión mejorada del algoritmo empleando conectividad 8-vecinos, 2 semillas y umbral 40. El proceso continúa generando "islas" dentro de la región. . El tiempo de ejecución fue 0.15 segundos.

Universidad Nacional de La Plata - Facultad de Informatica
Tesis de Grado de Walter Maximiliano Martinez Krawczuk
Segmentacion de Imagenes Medicas
versión mejorada del algoritmo de segmentación

Información	
presione para seleccionar	
Examinar...	
alto	ancho
281	281
tipo	resolucion
JPG	8 bits

Configuración	
conectividad	umbral
<input type="radio"/> 4 vecinos	80
<input checked="" type="radio"/> 8 vecinos	
ok	

Ayuda
<ul style="list-style-type: none">• presione "examinar" y seleccione una imagen• configure el algoritmo usando el panel de configuración• haga clic sobre la imagen, vea los resultados

[inicio](#)



Segmentación completada en 0.14 segundos

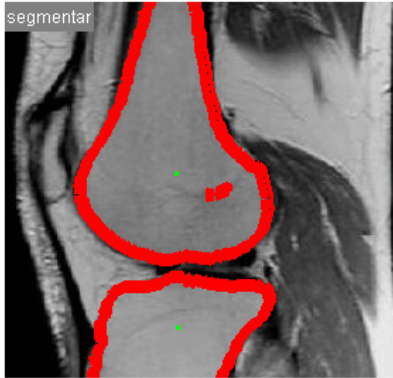
figura 56: resultado de haber segmentado la imagen con la versión mejorada del algoritmo empleando conectividad 8-vecinos, 2 semillas y umbral 80. La mayoría de las "islas" han desaparecido pero el resultado no es óptimo. El proceso demoró 0.14 segundos.

Información	
presione para seleccionar	
Examinar...	
alto	ancho
281	281
tipo	resolucion
JPG	8 bits

Configuración	
conectividad	umbral
<input type="radio"/> 4 vecinos	100
<input checked="" type="radio"/> 8 vecinos	
ok	

Ayuda
<ul style="list-style-type: none">• presione "examinar" y seleccione una imagen• configure el algoritmo usando el panel de configuración• haga clic sobre la imagen, vea los resultados

[inicio](#)



Segmentación completada en 0.156 segundos

figura 57: resultado de haber segmentado la imagen con la versión mejorada del algoritmo empleando conectividad 8-vecinos, 2 semillas y umbral 100. La mayoría de las islas han desaparecido pero el resultado sigue sin ser óptimo. El proceso demoró 0.15 segundos.

Universidad Nacional de La Plata - Facultad de Informatica
Tesis de Grado de Walter Maximiliano Martinez Krawczuk
Segmentacion de Imagenes Medicas
versión mejorada del algoritmo de segmentación

Información
presione para seleccionar

alto ancho

tipo resolución

Configuración
conectividad umbral
 4 vecinos
 8 vecinos

Ayuda

- presione "examinar" y seleccione una imagen
- configure el algoritmo usando el panel de configuración
- haga clic sobre la imagen, vea los resultados

[inicio](#)

segmentar



Segmentación completada en 0.156 segundos

figura 58: resultado de haber segmentado la imagen con la versión mejorada del algoritmo empleando conectividad 8-vecinos, 2 semillas y umbral 120. Vemos que la posición de las semillas hace variar el resultado final. El proceso demoró 0.15 segundos.

Información
presione para seleccionar

alto ancho

tipo resolución

Configuración
conectividad umbral
 4 vecinos
 8 vecinos

Ayuda

- presione "examinar" y seleccione una imagen
- configure el algoritmo usando el panel de configuración
- haga clic sobre la imagen, vea los resultados

[inicio](#)

segmentar



Segmentación completada en 0.156 segundos

figura 59: resultado de haber segmentado la imagen con la versión mejorada del algoritmo empleando conectividad 8-vecinos, 2 semillas y umbral 120. Las islas han desaparecido, el resultado es el esperado. El proceso demoró 0.15 segundos.

Universidad Nacional de La Plata - Facultad de Informática
Tesis de Grado de Walter Maximiliano Martínez Krawczuk
Segmentación de Imágenes Médicas
versión mejorada del algoritmo de segmentación

Información

presione para seleccionar

Examinar...

alto: ancho:

tipo: resolución:

Configuración

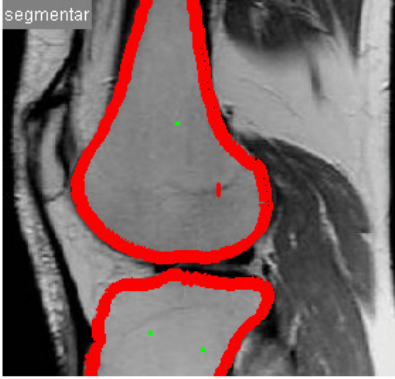
conectividad: 4 vecinos 8 vecinos

umbral:

Ayuda

- presione "examinar" y seleccione una imagen
- configure el algoritmo usando el panel de configuración
- haga clic sobre la imagen, vea los resultados

[inicio](#)



Segmentación completada en 0.234 segundos

figura 60: resultado de haber segmentado la imagen con la versión mejorada del algoritmo empleando conectividad 8-vecinos, 3 semillas y umbral 120. Las islas han desaparecido, el resultado es el esperado. El proceso demoró 0.23 segundos.

Imagen	Versión Clásica	Versión Mejorada
Resonancia Magnética Hemicara: 426 x 426 píxeles (RGB), 8 bits por píxel, formato PNG.	Demoró 25.9 segundos empleando conectividad 8-vecinos, nivel de umbral 55 y 2 semillas.	Demoró 0.32 segundos empleando conectividad 8-vecinos, nivel de umbral 55 y 2 semillas.
Resonancia Magnética rodilla: 237 x 95 píxeles (RGB), 8 bits por píxel, formato JPG.	Demoró 14.10 segundos empleando conectividad 8-vecinos, nivel de umbral 65 y 2 semillas.	Demoró 0.15 segundos empleando conectividad 8-vecinos, nivel de umbral 120 y 2 semillas.
	Demoró 8.62 segundos empleando conectividad 8-vecinos, nivel de umbral 65 y 4 semillas.	Demoró 0.23 segundos empleando conectividad 8-vecinos, nivel de umbral 120 y 3 semillas.

Tabla 5: Tabla comparativa entre versión clásica y mejorada del algoritmo de segmentación de imágenes por crecimiento de regiones

CAPÍTULO SEXTO: Conclusiones

17 ¿Porqué el método de segmentación propuesto es mejor que el original?

Antes de explicar por qué la performance de un algoritmo es mejor que la del otro explicaremos a qué nos referimos cuando decimos “performance”.

Como se expuso anteriormente, no hay una forma estándar de comparar algoritmos de segmentación de imágenes médicas. En general lo que se tiene en cuenta para poder decir que un algoritmo de segmentación es mejor que otro es el factor tiempo y la calidad de la segmentación. Entonces basaremos las comparaciones en dos aspectos: *tiempo* y *calidad*.

Con *tiempo* nos referimos a la duración del cómputo, medido en segundos.

Con *calidad* nos referimos a lograr cubrir la totalidad de la región segmentada sin dejar huecos.

A la hora de comparar las dos versiones del algoritmo debemos tener en cuenta las variables en juego:

1. cantidad de semillas
2. conectividad empleada
3. nivel del umbral
4. ubicación de las semillas sobre la imagen a segmentar

Entonces las preguntas que surgen naturalmente son:

- 1) ¿Cómo influye la cantidad de semillas en el resultado final?
- 2) ¿Qué efecto tiene la conectividad empleada en el resultado final?
- 3) ¿A mayor nivel de umbral mejor segmentación?
- 4) ¿Cómo influye la posición de las semillas en el resultado final?

Para lograr responder a estas preguntas primero vamos a presentar tablas de resultados obtenidos a partir segmentar la imagen que aparece a continuación empleado ambas versiones del algoritmo. Lo que se intentó hacer en el experimento fue extraer la región correspondiente a hueso craneal.



figura 61: tomografía computada del cerebro

Versión Clásica:

Nota: se resaltaron en color los mejores resultados

semillas	umbral	conectividad	tiempo	notas
1	10	4	42.61	hay huecos
1	20	4	42.35	hay huecos
1	30	4	42.36	hay huecos
1	40	4	42.31	hay huecos
1	50	4	42.31	ok
1	60	4	42.40	ok
1	70	4	42.18	ok
1	80	4	42.15	ok
1	90	4	42.10	sobre segmenta
2	10	4	21.23	hay huecos
2	20	4	21.60	hay huecos
2	30	4	21.59	hay huecos
2	40	4	21.15	hay huecos
2	50	4	21.04	ok
2	60	4	21.15	ok
2	70	4	21.10	ok
2	80	4	21.15	ok
2	90	4	21.10	sobre segmenta
3	10	4	21.21	hay huecos
3	20	4	21.64	hay huecos
3	30	4	21.60	hay huecos
3	40	4	21.15	hay huecos
3	50	4	20.93	ok
3	60	4	21.09	ok
3	70	4	21.0	ok
3	80	4	21.0	ok

3	90	4	21.04	sobre segmenta
4	10	4	12.15	hay huecos
4	20	4	13.15	hay huecos
4	30	4	13.14	hay huecos
4	40	4	11.07	hay huecos
4	50	4	11.12	ok
4	60	4	10.90	ok
4	70	4	10.90	ok
4	80	4	10.90	ok
4	90	4	12.59	sobre segmenta
5	10	4	12.11	hay huecos
5	20	4	13.04	hay huecos
5	30	4	13.14	hay huecos
5	40	4	11.12	hay huecos
5	50	4	11.12	ok
5	60	4	10.95	ok
5	70	4	10.90	ok
5	80	4	10.90	ok
5	90	4	12.54	sobre segmenta
6	10	4	12.10	hay huecos
6	20	4	13.14	hay huecos
6	30	4	13.09	hay huecos
6	40	4	11.12	hay huecos
6	50	4	11.12	ok
6	60	4	10.25	ok
6	70	4	10.20	ok
6	80	4	10.20	ok
6	90	4	10.20	sobre segmenta
7	10	4	12.11	hay huecos
7	20	4	13.14	hay huecos
7	30	4	13.09	hay huecos
7	40	4	11.03	hay huecos
7	50	4	11.12	ok
7	60	4	10.14	ok
7	70	4	10.09	ok
7	80	4	10.09	ok
7	90	4	10.25	sobre segmenta
8	10	4	11.03	hay huecos
8	20	4	10.57	hay huecos
8	30	4	10.57	hay huecos
8	40	4	10.14	hay huecos
8	50	4	9.92	ok
8	60	4	8.62	ok
8	70	4	8.67	ok
8	80	4	8.67	ok
8	90	4	8.67	sobre segmenta
9	10	4	11.01	hay huecos
9	20	4	10.57	hay huecos
9	30	4	10.57	hay huecos

9	40	4	10.14	hay huecos
9	50	4	9.87	ok
9	60	4	8.67	ok
9	70	4	8.71	ok
9	80	4	8.67	ok
9	90	4	8.67	sobre segmenta
10	10	4	8.93	hay huecos
10	20	4	10.57	hay huecos
10	30	4	10.57	hay huecos
10	40	4	10.18	hay huecos
10	50	4	9.92	ok
10	60	4	8.67	ok
10	70	4	8.67	ok
10	80	4	8.62	ok
10	90	4	8.62	sobre segmenta
11	10	4	8.84	hay huecos
11	20	4	10.48	hay huecos
11	30	4	10.53	hay huecos
11	40	4	10.09	hay huecos
11	50	4	9.92	ok
11	60	4	8.62	ok
11	70	4	8.67	ok
11	80	4	8.67	ok
11	90	4	8.67	sobre segmenta
12	10	4	8.89	hay huecos
12	20	4	10.57	hay huecos
12	30	4	10.53	hay huecos
12	40	4	10.09	hay huecos
12	50	4	9.92	ok
12	60	4	8.67	ok
12	70	4	8.67	ok
12	80	4	8.67	ok
12	90	4	8.67	sobre segmenta
13	10	4	8.89	hay huecos
13	20	4	10.53	hay huecos
13	30	4	10.53	hay huecos
13	40	4	10.14	hay huecos
13	50	4	9.92	ok
13	60	4	8.62	ok
13	70	4	8.71	ok
13	80	4	8.67	ok
13	90	4	8.67	sobre segmenta
14	10	4	8.89	hay huecos
14	20	4	10.53	hay huecos
14	30	4	10.53	hay huecos
14	40	4	10.09	hay huecos
14	50	4	9.87	ok
14	60	4	8.67	ok
14	70	4	8.67	ok

14	80	4	8.67	ok
14	90	4	8.67	sobre segmenta
15	10	4	7.31	hay huecos
15	20	4	8.51	hay huecos
15	30	4	8.51	hay huecos
15	40	4	7.03	hay huecos
15	50	4	7.03	ok
15	60	4	6.65	ok
15	70	4	6.70	ok
15	80	4	6.76	ok
15	90	4	6.81	sobre segmenta
16	10	4	7.31	hay huecos
16	20	4	8.51	hay huecos
16	30	4	8.51	hay huecos
16	40	4	6.98	hay huecos
16	50	4	6.98	ok
16	60	4	5.34	ok
16	70	4	5.29	ok
16	80	4	5.29	ok
16	90	4	5.29	sobre segmenta
17	10	4	7.32	hay huecos
17	20	4	8.56	hay huecos
17	30	4	8.56	hay huecos
17	40	4	6.98	hay huecos
17	50	4	6.98	ok
17	60	4	5.34	ok
17	70	4	5.34	ok
17	80	4	5.34	ok
17	90	4	5.34	sobre segmenta
18	10	4	7.35	hay huecos
18	20	4	8.56	hay huecos
18	30	4	8.56	hay huecos
18	40	4	7.03	hay huecos
18	50	4	6.98	ok
18	60	4	5.29	ok
18	70	4	5.29	ok
18	80	4	5.29	ok
18	90	4	5.29	sobre segmenta
1	10	8	30.84	hay huecos
1	20	8	30.71	hay huecos
1	30	8	30.76	hay huecos
1	40	8	30.73	hay huecos
1	50	8	30.67	ok
1	60	8	30.5	ok
1	70	8	30.5	ok
1	80	8	30.53	ok
1	90	8	30.5	sobre segmenta
2	10	8	15.60	hay huecos
2	20	8	15.60	hay huecos

2	30	8	15.61	hay huecos
2	40	8	15.65	hay huecos
2	50	8	15.54	ok
2	60	8	15.56	ok
2	70	8	15.54	ok
2	80	8	15.5	ok
2	90	8	15.54	sobre segmenta
3	10	8	15.28	hay huecos
3	20	8	15.21	hay huecos
3	30	8	15.17	hay huecos
3	40	8	15.17	hay huecos
3	50	8	15.17	ok
3	60	8	14.98	ok
3	70	8	15.01	ok
3	80	8	15.0	ok
3	90	8	14.97	sobre segmenta
4	10	8	13.53	hay huecos
4	20	8	12.5	hay huecos
4	30	8	12.43	hay huecos
4	40	8	12.39	hay huecos
4	50	8	12.61	ok
4	60	8	12.45	ok
4	70	8	12.43	ok
4	80	8	12.43	ok
4	90	8	13.14	sobre segmenta
5	10	8	13.5	hay huecos
5	20	8	12.5	hay huecos
5	30	8	12.39	hay huecos
5	40	8	12.43	hay huecos
5	50	8	12.43	ok
5	60	8	12.32	ok
5	70	8	12.34	ok
5	80	8	12.28	ok
5	90	8	13.14	sobre segmenta
6	10	8	9.21	hay huecos
6	20	8	8.45	hay huecos
6	30	8	8.45	hay huecos
6	40	8	8.40	hay huecos
6	50	8	8.40	ok
6	60	8	8.45	ok
6	70	8	8.45	ok
6	80	8	8.45	ok
6	90	8	8.45	sobre segmenta
7	10	8	9.21	hay huecos
7	20	8	8.23	hay huecos
7	30	8	8.18	hay huecos
7	40	8	7.90	hay huecos
7	50	8	7.85	ok
7	60	8	7.90	ok

7	70	8	7.90	ok
7	80	8	7.86	ok
7	90	8	7.86	sobre segmenta
8	10	8	7.20	hay huecos
8	20	8	6.32	hay huecos
8	30	8	6.21	hay huecos
8	40	8	6.21	hay huecos
8	50	8	6.21	ok
8	60	8	6.21	ok
8	70	8	6.21	ok
8	80	8	6.21	ok
8	90	8	6.21	sobre segmenta
9	10	8	7.23	hay huecos
9	20	8	6.32	hay huecos
9	30	8	6.17	hay huecos
9	40	8	6.21	hay huecos
9	50	8	6.21	ok
9	60	8	6.21	ok
9	70	8	6.21	ok
9	80	8	6.21	ok
9	90	8	6.21	sobre segmenta
10	10	8	4.85	hay huecos
10	20	8	4.85	hay huecos
10	30	8	4.85	hay huecos
10	40	8	4.85	hay huecos
10	50	8	4.86	ok
10	60	8	4.85	ok
10	70	8	4.90	ok
10	80	8	4.90	ok
10	90	8	4.87	sobre segmenta
11	10	8	4.75	hay huecos
11	20	8	4.79	hay huecos
11	30	8	4.79	hay huecos
11	40	8	4.75	hay huecos
11	50	8	4.75	ok
11	60	8	4.79	ok
11	70	8	4.79	ok
11	80	8	4.75	ok
11	90	8	4.75	sobre segmenta
12	10	8	4.64	hay huecos
12	20	8	3.76	hay huecos
12	30	8	3.76	hay huecos
12	40	8	3.76	hay huecos
12	50	8	3.76	ok
12	60	8	3.76	ok
12	70	8	3.81	ok
12	80	8	3.76	ok
12	90	8	8.81	sobre segmenta

Tabla 6: conjunto de resultados obtenidos segmentando la imagen que aparece en la figura 61 haciendo variar la cantidad de semillas, posición de las semillas, nivel de umbral y conectividad empleando la versión clásica del algoritmo. Los mejores resultados fueron resaltados.

Versión Mejorada:

Nota: Se resaltaron en color los mejores resultados

semillas	umbral	conectividad	tiempo	notas
1	10	4	0.09	no cubre toda la región
1	20	4	0.15	ok
1	30	4	0.11	ok
1	40	4	0.09	ok
1	50	4	0.1	hay huecos
1	60	4	0.11	hay huecos
1	70	4	0.10	hay huecos
1	80	4	0.09	ok
1	90	4	0.10	ok
1	100	4	0.10	ok
1	110	4	0.10	ok
1	120	4	0.15	ok
1	130	4	0.09	ok
1	140	4	0.11	ok
1	150	4	0.09	ok
1	160	4	0.10	ok
1	170	4	0.10	ok
1	180	4	0.11	sobre segmenta
2	10	4	0.18	no cubre toda la región
2	20	4	0.21	hay huecos
2	30	4	0.25	hay huecos
2	40	4	0.18	hay huecos
2	50	4	0.18	hay huecos
2	60	4	0.18	hay huecos
2	70	4	0.18	hay huecos
2	80	4	0.21	ok
2	90	4	0.20	ok
2	100	4	0.21	ok
2	110	4	0.20	ok
2	120	4	0.18	ok
2	130	4	0.26	ok
2	140	4	0.21	ok
2	150	4	0.20	ok
2	160	4	0.18	sobre segmenta
3	10	4	0.23	no cubre toda la región
3	20	4	0.29	hay huecos
3	30	4	0.29	hay huecos
3	40	4	0.29	hay huecos
3	50	4	0.31	hay huecos

3	60	4	0.32	hay huecos
3	70	4	0.34	hay huecos
3	80	4	0.28	ok
3	90	4	0.28	ok
3	100	4	0.29	ok
3	110	4	0.28	ok
3	120	4	0.29	ok
3	130	4	0.28	ok
3	140	4	0.28	ok
3	150	4	0.32	ok
3	160	4	0.29	sobre segmenta
4	10	4	0.40	hay huecos
4	20	4	0.46	hay huecos
4	30	4	0.40	hay huecos
4	40	4	0.37	hay huecos
4	50	4	0.37	hay huecos
4	60	4	0.37	hay huecos
4	70	4	0.40	hay huecos
4	80	4	0.40	ok
4	90	4	0.39	ok
4	100	4	0.45	ok
4	110	4	0.42	ok
4	120	4	0.45	ok
4	130	4	0.39	ok
4	140	4	0.40	ok
4	150	4	0.40	ok
4	160	4	0.40	sobre segmenta
1	10	8	0.09	no cubre toda la región
1	20	8	0.10	ok
1	30	8	0.10	ok
1	40	8	0.10	ok
1	50	8	0.10	ok
1	60	8	0.11	ok
1	70	8	0.11	ok
1	80	8	0.10	ok
1	90	8	0.10	ok
1	100	8	0.10	ok
1	110	8	0.10	ok
1	120	8	0.09	ok
1	130	8	0.09	ok
1	140	8	0.11	ok
1	150	8	0.09	ok
1	160	8	0.10	ok
1	170	8	0.10	ok
1	180	8	0.10	ok
1	190	8	0.10	sobre segmenta
2	10	8	0.25	ok
2	20	8	0.21	ok
2	30	8	0.20	ok

2	40	8	0.20	ok
2	50	8	0.20	ok
2	60	8	0.21	ok
2	70	8	0.21	ok
2	80	8	0.21	ok
2	90	8	0.21	ok
2	100	8	0.21	ok
2	110	8	0.26	ok
2	120	8	0.20	ok
2	130	8	0.20	ok
2	140	8	0.21	ok
2	150	8	0.20	ok
2	160	8	0.21	ok
2	170	8	0.21	ok
2	180	8	0.12	sobre segmenta
3	10	8	0.26	no cubre toda la región
3	20	8	0.29	ok
3	30	8	0.31	ok
3	40	8	0.34	ok
3	50	8	0.35	ok
3	60	8	0.31	ok
3	70	8	0.32	ok
3	80	8	0.32	ok
3	90	8	0.31	ok
3	100	8	0.32	ok
3	110	8	0.32	ok
3	120	8	0.31	ok
3	130	8	0.32	ok
3	140	8	0.31	ok
3	150	8	0.37	ok
3	160	8	0.32	ok
3	170	8	0.31	sobre segmenta

Tabla 7: conjunto de resultados obtenidos segmentando la imagen que aparece en la figura 61 haciendo variar la cantidad de semillas, posición de las semillas, nivel de umbral y conectividad empleando la versión mejorada del algoritmo. Los mejores resultados fueron resaltados.

Resumen de los mejores resultados obtenidos:

Versión Clásica:

semillas	umbral	conectividad	tiempo	notas
1	80	4	42.15	ok
2	50	4	21.04	ok
3	70	4	21.0	ok
4	60	4	10.90	ok
5	70	4	10.90	ok
6	70	4	10.20	ok
7	70	4	10.09	ok

8	60	4	8.62	ok
9	60	4	8.67	ok
10	80	4	8.62	ok
11	60	4	8.62	ok
12	60	4	8.67	ok
13	60	4	8.62	ok
14	60	4	8.67	ok
15	60	4	6.65	ok
16	70	4	5.29	ok
17	60	4	5.34	ok
18	60	4	5.29	ok
1	60	8	30.5	ok
2	80	8	15.5	ok
3	60	8	14.98	ok
4	70	8	12.43	ok
5	80	8	12.28	ok
6	50	8	8.40	ok
7	80	8	7.86	ok
8	50	8	6.21	ok
9	50	8	6.21	ok
10	60	8	4.85	ok
11	50	8	4.75	ok
12	50	8	3.76	ok

Tabla 8: subconjunto de los mejores resultados obtenidos en la tabla 6, empleado la versión clásica del algoritmo.

Versión Mejorada:

semillas	umbral	conectividad	tiempo	notas
1	40	4	0.09	ok
2	120	4	0.18	ok
3	80	4	0.28	ok
4	80	4	0.40	ok
1	120	8	0.09	ok
2	30	8	0.20	ok
3	20	8	0.29	ok

Tabla 9: subconjunto de los mejores resultados obtenidos en la tabla 6, empleado la versión clásica del algoritmo.

Notas:

- recordar que la versión mejorada NO rellena todo el interior de la región, sólo colorea los bordes
- si las semillas están muy juntas no se obtienen mejoras notorias
- a igualdad de tiempo y calidad, es mejor aquel que emplee menos cantidad de semillas

Ahora procederemos a responder a cada una de las interrogantes planteadas

1) ¿Cómo influye la cantidad de semillas en el resultado final?

Esta pregunta está relacionada al factor tiempo.

Versión Clásica:

- A mayor cantidad de semillas menor tiempo
- La posición de las semillas influye en el resultado final
- Para la misma cantidad y posición de semillas, en general la 8-conectividad demora menos tiempo de cómputo
- Se requieren muchas mas semillas empleado 4-conectividad, para obtener resultados similares a los obtenidos empleando 8-conectividad con menor número de semillas
- En teoría, se podrían obtener resultados similares a los de la versión mejorada empleando 8-conectividad y un elevado número de semillas. Para las pruebas, se encontró que empleando 8-conectividad y cerca de 50 semillas se obtienen resultados similares a los obtenidos con la versión mejorada (en cuanto al tiempo)

Versión Mejorada:

- Se obtienen mejores resultados empleando pocas semillas
- A mayor cantidad de semillas mayor tiempo de cómputo
- Mas semillas no implican necesariamente mejor calidad de segmentación (no es verdad que más semillas logren cubrir mejor la región)

Luego, vemos que es mejor la versión mejorada porque con menos cantidad de semillas se obtienen mejores resultados.

2) ¿Qué efecto tiene la conectividad empleada en el resultado final?

Esta pregunta está relacionada tanto al factor tiempo como al factor calidad.

Versión Clásica:

- la 8-conectividad logra cubrir mejor la región
- la 4-conectividad demora más tiempo

Versión Mejorada:

- la 8-conectividad logra cubrir mejor la región
- la 4-conectividad demora más tiempo

En este caso no podemos concluir que un algoritmo sea mejor que el otro.

3) ¿A mayor nivel de umbral mejor segmentación?

Esta pregunta viene asociada al factor calidad.

Versión Clásica:

- En general a mayor nivel de umbral se logran mejores resultados en el sentido de lograr cubrir mejor la región que nos interesa evitando huecos. Pero esto también agrega tiempo de retardo
- En base a las pruebas efectuadas encontramos que los mejores resultados se obtuvieron haciendo variar el nivel de umbral entre 50 y 80. Independientemente de la cantidad de semillas empleadas

Posible mejora propuesta para la versión Clásica:

Una posible mejora surge de la idea de auto-detectar el mejor nivel de umbral para la imagen a segmentar.

Sería posible hacer esto si el algoritmo fuese "entrenado" por el operador. Con entrenar nos referimos a que una vez cargada la imagen en el programa y colocadas las semillas, el sistema podría proveer un conjunto de resultados al operador haciendo variar el nivel de umbral. El operador simplemente tendría que aceptar o rechazar el resultado. De esta manera el programa "aprendería" qué rango de nivel de umbral es aceptable.

En base a las pruebas realizadas, se concluye que si el nivel de umbral se mantiene dentro de ese rango los resultados serán buenos independientemente de la cantidad de semillas empleadas.

De esta forma el operador no tendría que seleccionar el nivel de umbral.

Pero como también podemos concluir que la 8-conectividad arroja mejores resultados que la 4-conectividad, podemos eliminar este parámetro y asumir siempre 8-conectividad.

Finalmente, el operador solo tendría que preocuparse por colocar las semillas sobre la imagen. De esta forma el programa sería más simple de utilizar y se obtendrían buenos resultados siempre. Este “entrenamiento” debería ser realizado antes de segmentar un conjunto de imágenes obtenidas de una misma fuente y con características similares (ejemplo: una serie de tomografías computadas). Debido a que la calidad de la segmentación depende del nivel de umbral empleado y como las imágenes tendrán características similares (contraste, luz) sería factible esta mejora propuesta.

Versión Mejorada:

- No necesariamente a mayor nivel de umbral mejor segmentación. La calidad depende de la posición de las semillas y de su cantidad mas que del nivel de umbral
- Es cierto que el umbral no debería estar muy cerca de los valores límites para obtener buenos resultados (0 y 255)

Concluimos que la versión mejorada es mejor por ser más independiente del nivel de umbral (se obtienen buenos resultados sin tener tanto en cuenta el umbral usado). Además, en la versión clásica a mayor nivel de umbral mayor retardo y esto no pasa en la versión mejorada.

4) ¿Cómo influye la posición de las semillas en el resultado final?

Es una pregunta relacionada con el factor calidad.

Versión Clásica:

- La posición de las semillas es muy importante para lograr segmentar correctamente la imagen
- No se podrían obtener buenos resultados si se colocaran mal las semillas

Versión Mejorada:

- La posición de las semillas es muy importante para lograr segmentar correctamente la imagen
- No se podrían obtener buenos resultados si se colocaran mal las semillas

Notamos que ambas versiones son necesariamente interactivas: no podrían funcionar si el operador no colocara las semillas sobre la imagen. No podemos concluir en este caso que un algoritmo sea mejor que el otro.

Conclusiones finales

Basándonos en las comparaciones efectuadas anteriormente podemos concluir lo siguiente:

- La versión mejorada es notoriamente más veloz que la versión clásica, empleado además menor cantidad de semillas
- La versión clásica tiene mas retardo a mayor nivel de umbral y esto no ocurre en la versión mejorada
- La versión mejorada, al no colorear internamente la región segmentada, demora menos tiempo. Se concentra en detectar los bordes de la región, los que colorea
- Visualmente, el efecto de pintar internamente la región segmentada (no solo sus bordes) puede resultar más cómodo para el operador. Esto es una ventaja de la versión clásica
- Seria posible modificar la versión clásica del algoritmo para facilitar su uso según se explicó
- Además, la versión mejorada fue implementada en plataforma web y pensada para poder ser empleada en una red como un servicio de segmentación compartido, disponible para todos los usuarios de esa red. Esto significa que cualquier usuario conectado a la red podrá emplear el sistema, incluso podría emplearse desde otras redes por medio de Internet. Es una solución mucho mas escalable que la solución clásica que corre como aplicación de escritorio, no requiere instalar software ni hardware especial y soporta imágenes en formato DICOM, JPEG, PNG, GIF, etc.

18 Trabajos futuros

La versión mejorada puede seguir evolucionando si se le incorporan mejoras al algoritmo como por ejemplo soporte para imágenes en 3 dimensiones, clasificación de regiones y soporte para otros formatos de imágenes.

El soporte para otros formatos de imágenes puede ser implementado en lenguaje de scripting del lado del servidor (en la implementación mejorada se empleó PHP) esto implica que no necesitaremos modificar el algoritmo de segmentación (que fue implementado como un Applet en Java) para incorporarle soporte para otros formatos de imágenes, lo cual es muy bueno porque nos permite abstraernos de la implementación interna del algoritmo de segmentación.

Otra variante interesante surge de la idea de poder clasificar las regiones una vez segmentadas. La clasificación consistiría en asignar un color distinto a cada región identificada. Esto sería un proceso automático que requeriría de una configuración mínima por parte del operador (por ejemplo seleccionar el pool de colores a emplear.).

ANEXOS:

Anexo A: Clasificación de Imágenes Médicas

La segmentación de imágenes es definida como la partición de una imagen en regiones no solapadas, las cuales son homogéneas con respecto a alguna característica como ser: intensidad o una textura o color. Si el dominio de la imagen está dado por I , entonces el problema de segmentación radica en encontrar el conjunto S_k cuya unión es la imagen I completa. Por lo tanto, el conjunto que forma la segmentación deberá satisfacer la fórmula siguiente:

$$I = \bigcup_{k=1}^K S_k$$

donde $S_k \cap S_j = \emptyset$ para $k \neq j$ y cada S_k está conectado. Idealmente, un método de segmentación encontrará aquellos conjuntos que corresponden a distintas regiones o estructuras anatómicas de interés dentro de la imagen.

Si eliminamos la restricción de que las regiones estén conectadas, al proceso de determinar los conjuntos S_k lo llamamos *clasificación de píxel* y a los conjuntos los llamamos *clases*.

La clasificación de píxeles es, frecuentemente, un objetivo deseable en el tratamiento de

imágenes médicas. En especial cuando se necesita clasificar regiones desconectadas que pertenecen a un mismo tejido.

Determinar el número K de clases en la clasificación de píxeles puede ser un problema complejo, por lo que *generalmente lo asumimos conocido*, basado en conocimientos previos de la anatomía en estudio.

Los métodos clasificadores son técnicas de reconocimiento de patrones, que buscan dividir un espacio característico derivado de una imagen empleando datos con etiquetas conocidas. Llamamos *espacio característico* a un rango espacial de cualquier función de la imagen, siendo las intensidades de la imagen el más común de los espacios característicos.

Los clasificadores son conocidos como *métodos supervisados* debido a que requieren datos de entrenamiento que deben ser suministrados en forma manual, para ser luego empleados en la segmentación automática de nuevos datos. Hay muchas formas de aplicar los datos de entrenamiento en los métodos de clasificación. Un clasificador simple es el clasificador del vecino más cercano, donde cada píxel es clasificado dentro de la misma clase que el dato de entrenamiento con la intensidad más cercana.

Los clasificadores estándar requieren que la estructura a ser segmentada cuente con características cuantificables distintas. Debido a que los datos de entrenamiento pueden ser etiquetados, los clasificadores pueden transferir estas etiquetas a los nuevos datos siempre que el espacio característico distinga cada etiqueta lo suficiente. No son iterativos, por lo que son relativamente eficientes computacionalmente y pueden ser aplicados a imágenes multicanal. Una desventaja de los clasificadores es que no obedecen a ningún modelo espacial. Otra desventaja es la necesidad de la interacción manual para obtener los datos de entrenamiento. Los conjuntos de entrenamiento pueden ser adquiridos para cada imagen que requiera segmentación, pero esto puede tomar mucho tiempo y esfuerzo. Adicionalmente, el uso de un mismo conjunto de entrenamiento para gran número de imágenes puede llevar a resultados confusos, ya que no tienen en cuenta las variaciones fisiológicas y anatómicas entre diferentes sujetos.

Anexo B: Otras definiciones

- **Adquirir** Este término se refiere a un comando de menú que se encuentra en aplicaciones de edición de imágenes como Adobe Photoshop que se usa para cambiar el control a un dispositivo periférico como un scanner para el programa interface del usuario. El interface del usuario se llama un driver Plug- In o TWAIN.
- **Colores aditivos** Se refiere a los colores de la rueda de colores primarios Rojo, Verde y Azul, donde cualquier color puede componerse a partir de estos tres colores primarios.
- **ADF (Alimentador Automático de Documentos)** Un aparato que puede proporcionar al scanner una hoja por vez de un montón de documentos. Por lo tanto, el usuario puede escanear diferentes documentos a la vez.
- **Aplicación** Un paquete de software diseñado para fines específicos.
- **Razón de Aspecto** Un número que describe la relación proporcional entre la altura y la anchura de un rectángulo o un cuadrado.
- **Auto Densidad** Un término de procesado de imagen que describe el ajuste de los niveles de luces y sombras en las imágenes a su realzado dinámico de rango máximo (libre de distorsión).
- **Bit** La unidad más pequeña de memoria. 8 bits hacen 1 byte. Un bit puede representarse por 1 o 0 imágenes a pluma p. e. 1 para negro, 0 para blanco.
- **BMP (Archivo de Mapa de Bits)** La extensión del nombre que se da a archivos de imágenes gráficas de mapa de bit.
- **Brillo** Determina las intensidades de los elementos sombreados de la imagen. Una imagen de blanco puro tiene el máximo de brillo, de manera que una imagen negra pura tiene el mínimo de brillo. El contraste, por otro lado, mide el grado del sombreado en un elemento.

- Calibrar Dice se del proceso de ajustar la entrada o la salida de un dispositivo con respecto a una muestra de referencia
- CCD Charged Coupled Device. Un dispositivo semiconductor fotosensitivo que convierte la intensidad de la luz en una señal eléctrica.
- CGM Computer Graphics Metafile.
- Canal Se refiere a cada uno de los tres porciones (rojo, azul, verde) de los datos de una imagen de color en un archivo de imagen a color.
- Cromático Este término se refiere al matiz relativo de un color.
- CMYK CMYK (Cían, Magenta, Amarillo, y Negro) es el espacio de color que utilizan la mayoría de los impresores.
- Método de Calibración del Color Se refiere al algoritmo que se usa para calibrar un dispositivo de entrada y/ o salida para una reproducción exacta de imágenes que se producen/ muestran en dispositivos de salida/ entrada.
- Canal de Color Se refiere a cada uno de los componentes de color Rojo, Verde y Azul y al Componente Master.
- Calibración de Color Se refiere al proceso de mantener los colores fieles en imágenes gráficas al ser pasadas de los documentos originales de dispositivos de entrada (p. e. scanner) a muestras de salida o ámbito de Color. Los ámbitos de color son tres modelos dimensionales que se usan para describir el color. RVA (Rojo, Verde, Azul) es el ámbito de color para la mayoría de monitores. CMYK (Cían, Magenta, Amarillo y Negro) es el ámbito de color usado por la mayoría de impresiones.

- **Contraste** Se refiere al rango de brillo o sombras de grises en una imagen. Una imagen de mayor contraste es una imagen en blanco y negro sin sombras de grises. Una imagen de bajo contraste muestra sombras de grises en un rango medio.
- **Defecto** Se refiere a los valores de fábrica predeterminados del programa de software o hardware para la configuración de los parámetros.
- **Destramar** Se refiere a una de las Funciones de filtrado. Esta función puede reducir los efectos de patrones no deseados.
- **Cuadro de diálogo** Una ventana en la pantalla que proporciona al usuario de la computadora una elección entre múltiples opciones en diferentes cursos de acción.
- **Dpi (Puntos por pulgada)** Una medida de la resolución de un dispositivo de entrada (p. e. scanner) o salida (p. e. pantalla del monitor, impresora). Para la presentación en la pantalla del monitor, se refiere al número de píxeles que se pueden mostrar en una pulgada lineal. Para una impresora, se refiere al número de puntos por pulgada que se pueden imprimir para una pulgada lineal.
- **Rango Dinámico** Se refiere a la extensión en que imágenes en escala de grises o a color explotan el rango total de posibles niveles de intensidad (brillo o saturación.)
- **EPS Encapsulated Postscript Files.** Este es un método de almacenamiento de una imagen mediante series de descripciones que se pueden ser interpretados por una impresora para una reproducción posterior. Estos archivos de imágenes se usan con impresoras PostScript.
- **Filtrado** Esta opción es útil para suavizar o enfocar una imagen escaneada. MagicScan tiene dos filtros: uno para desenfocar y el otro para enfocar la imagen. La opción de Desenfocar de MagicScan le permite definir contornos de los bordes o de color.

- Corrección de Gama Esto ajusta los valores Gama de los píxeles en su trama para compensar las diferencias entre el entremado de color original de la imagen y el que se muestra en su monitor.
- Escala de grises Las sombras de grises entre blanco y negro, que se encuentran en una imagen.
- Claridad Se refiere a las áreas más claras de la imagen.
- Histograma Se refiere a una representación en un gráfico que muestra el total de píxeles para cada uno de los 256 niveles de color o de sombras de gris.
- Matiz La calidad que distingue un color de otro.
- Software de Edición de Imágenes Software que se utiliza para crear o cambiar imágenes para una reproducción posterior. Adobe Photoshop es un ejemplo de ello.
- Image Setter Un dispositivo de salida que se usa para documentos de alta resolución en película fotográfica o papel.
- Interpolación Este término se refiere al método que se usa para compensar el vacío de la resolución óptica. El software de procesamiento de la imagen calcula la resolución que se necesita e introduce el número de píxeles o puntos para la imagen.
- Eslabones El efecto como de eslabones que aparece al aumentar una imagen. En lugar de verse una línea desenfocada, se ven unos eslabones. La interpolación se usa para eliminarlos.
- JPEG (Joint Photographic Experts Group) Se refiere a un estándar de compresión de archivos de imagen que comprime archivos de imagen en una fracción del tamaño original con algunas pérdidas en la calidad de la imagen.

- Nivel En los datos de imágenes en escala de grises, este término se refiere al valor de los datos de brillo para cada píxel. En datos de una imagen de color, se refiere a los valores de los datos de saturación para cada uno de los píxeles de colores primarios (canales). En los dos casos, siempre es un valor entre 0 y 255.
- Pluma Una imagen que consista en valores puramente blancos y negros sin ningún valor intermedio de grises o color. Ejemplos de esto son clipart blanco y negro, dibujos en líneas y documentos de texto. Por lo tanto, en las imágenes se almacenan como un formato de un bit o un píxel.
- Aumento (o Zoom) en una área seleccionada para mostrar más detalles en esa área.
- Memoria La cantidad de espacio para almacenar, medido en bytes, de su sistema.
- Gama Partes de una imagen con gradaciones de tonos con valores intermedios, normalmente entre el 25% y 75%.
- Moiré Un efecto no deseado que resulta de patrones de interaplados de luz y sombra alternadas. Normalmente resulta de reescanear una imagen en medios tonos o escanear una imagen de una publicación que usa una impresión de medios tonos. Si esto ocurre intente utilizar la fotografía original o la transparencia.
- OCR Optical Character Recognition. Es un software que tiene la habilidad de reconocer la mayoría de formatos de caracteres. Por lo tanto, un documento escaneado puede leerse y editarse en su procesador de textos.
- Resolución Óptica Esta es la resolución verdadera del scanner. Es una medida del sistema óptico para distinguir imágenes situadas muy cerca. El otro tipo de resolución es la resolución interpolada donde los detalles que faltan de una imagen se infieren.
- PCT La extensión de archivo que se utiliza para formatos de archivo PICT.

- PCX Este es un formato de archivo en mapa de bits con capacidades para escala de grises. Se usa de manera menos frecuente que otros formatos de archivo ae tales como TIFF.
- PICT (PICTure) El formato de una imagen gráfica que se puede codificar en mapa de bits y imágenes gráficas orientadas a un objeto. Utilizado originalmente en aplicaciones Apple Macintosh, ahora también se usa en algunas aplicaciones IBM PC.
- Píxel (Picture Element) Se refiere al elemento de la imagen más pequeño (punto) que define una imagen en una pantalla de ordenador.
- Profundidad de Píxel Este término se refiere al grado en que un color puede ser traducido por un sólo píxel. Depende de cómo sea la imagen. Millones de colores RVA (24- bit), miles de colores 64K (16- bit), 256 colores (8- bit), pluma (1- bit) -blanco y negro.
- Plug-Ins Los módulos Plug-In son aplicaciones o programas que ha creado Macintosh o una tercera persona para funcionar con la aplicación objetivo dentro de un ambiente Macintosh.
- Preprocesado Procesado de la imagen que se lleva a cabo al mismo tiempo que la operación de escaneo.
- Preescaneo El acto de escanear una imagen sin guardarla en el disco.
- Impresión Envía la imagen escaneada o manipulada a la impresora para la salida.
- Raster Término que se da para una cantidad de espacio que se sitúa entre los elementos en una línea horizontal en un monitor.
- Resolución Es una medida de la habilidad de un sistema óptico para distinguir imágenes situadas de manera muy cercana. Las unidades de medida son puntos por pulgada (dpi), líneas por pulgada (lpi) o píxeles por pulgada.

- RVA Una manera de definir cualquier color posible como una combinación de Rojo, Verde, Azul. Estos tres colores forman el modelo de color aditivo.
- Saturación Un término de imagen que se utiliza en conexión con imágenes de color, refiriéndose a la medida de la intensidad de cualquier color en una imagen de color
- Escalado Este término se refiere a la reducción o el aumento de la imagen al tiempo que se mantienen las proporciones de los lados.
- Scanner Un dispositivo que captura una imagen y la convierte en un mapa de bits que puede almacenarse en el disco A.
- SCSI Small Computer Systems Interface - Un interface con algo de inteligencia, usado por muchos scanners y otros dispositivos, incluyendo discos duros.
- Sombrear/ sombreados Un término general para referirse a las variaciones de tono, como son las variaciones de claridad en una imagen en escala de grises, o variaciones de color y saturación en una imagen a color.
- Sombra La área más oscura en una imagen.
- Colores Substractivos Los colores substractivos están representados por los tres colores primarios Cian, Magenta y Amarillo . Los colores substractivos absorben parte del espectro de la luz reflejada y vuelven a emitir otros. Por ejemplo, Cian absorbe la luz roja mientras que refleja la luz azul y verde; el amarillo sólo absorbe la luz azul, mientras que refleja luz roja y verde. Por lo tanto, al mezclar Cian y Amarillo totalmente saturados da verde
- TGA Un formato de archivo en mapa de bits para imágenes gráficas que fue desarrollado originalmente por pantallas Truevision Targa.

- TIFF (Tagged Image File Format) Un formato de archivo de imagen gráfica que está apoyado universalmente por todos los dispositivos de escaneo de imagen. Este formato de archivo fue lanzado al mercado conjuntamente por Aldus y Microsoft.
- Tonos Un término de imagen que se refiere a colores y mezclas de colores y tiene aproximadamente el mismo significado que el término general sombreado.
- TWAIN (Technology Without An Independent Name) Un driver independiente de la aplicación que le permite controlar una aplicación de Ventanas desde dentro de otra aplicación de Ventanas,

Anexo C: Figuras y Tablas

Figura o Tabla	Página
figura 1: 8 vecinos	9
figura 2: diagrama de cromaticidad según el estándar de la CIE. La curva representa los colores visibles saturados al 100%. W representa el color blanco. Los números en la curva representan longitudes de onda.	14
Figura 3: tetraedro de color RGB. Los puntos a lo largo de la diagonal principal tienen valores de gris, desde el negro en el origen del punto (1,1,1)	15
Tabla 1: distintas representaciones de color	20
figura 4: estructura básica de un archivo de imagen	25
figura 5: representación de la transformación de dos imágenes en una	31
figura 6: (a) imagen original sintética inicial; (b) imagen original sintética en un instante posterior, donde el cuadrado se ha acercado a la cámara; (c) diferencia de las dos imágenes. El movimiento se detecta por el contorno negro	34
figura 7: disposición en forma de rejilla de los píxeles de una imagen	37
figura 8: ante cualquier transformación geométrica, los píxeles de la rejilla transformada (línea punteada) no tienen por qué coincidir con los de la rejilla destino o final	38
figura 9: esquema gráfico de la interpolación	39
figura 10: dilatación	44
figura 11: (a) triángulo de color HSI; (b) color e intensidad en el modelo HSI	47
figura 12: diagrama de bloques para la transformación de una imagen de grises a pseudocolor. $R(x,y)$, $G(x,y)$ y $B(x,y)$ son las salidas de las respectivas componentes RGB de color.	51
figura 13: representación de histograma	52
figura 14: histogramas de intensidad; parte superior: umbral único y parte inferior: umbrales múltiples	58
figura 15: histograma de intensidad como suma de dos funciones de densidad de probabilidad	60
figura 16: (a) una escena con seis regiones distintas, (b) el grafo de adyacencia de la escena	65
figura 17: matriz de adyacencia	65
figura 18: parte superior: imagen dividida; parte inferior: árbol cuaternario correspondiente	67
figura 19: algoritmo de fusión y división	69
figura 20: tomografía computada	73
figura 21: resonancia magnética	74
Tabla 2: tomografía computada VS resonancia magnética	77

figura 22: umbralización empleado un nivel de umbral de 255	82
figura 23: umbralización empleado un nivel de umbral de 192	83
figura 24: umbralización empleado un nivel de umbral de 128	84
figura 25: umbralización empleado un nivel de umbral de 64	85
figura 26: umbralización empleado un nivel de umbral de 3	86
figura 27: crecimiento de regiones mediante adición de píxeles	89
figura 28: crecimiento de regiones mediante adición de píxeles empleando umbral 0	90
figura 29: crecimiento de regiones mediante adición de píxeles empleando umbral 64	91
figura 30: crecimiento de regiones mediante adición de píxeles empleando umbral 128	92
figura 31: crecimiento de regiones mediante adición de píxeles empleando umbral 192	93
figura 32: crecimiento de regiones mediante adición de píxeles empleando umbral 255	94
Tabla 3: versión clásica VS versión mejorada	99
figura 33: imagen a segmentar	101
figura 34: colocando las semillas	102
figura 35: segmentando con conectividad 8-vecinos, el tiempo de ejecución fue de 7.5 segundos	102
figura 36: configurando el programa para emplear 4-conectividad en lugar de 8-conectividad	103
figura 36: el tiempo de ejecución fue de 4.7 segundos	103
figura 37: resonancia magnética	104
figura 38: resonancia magnética de hemicara derecha: imagen empleada para la comparación de los algoritmos. En este ejemplo se pretende segmentar la imagen separando la región correspondiente a hueso craneal del resto de la imagen.	106
figura 39: paso 1, clásica: seleccionamos la imagen a segmentar	107
figura 40: paso 2, clásica: colocamos las semillas y configuramos la conectividad y el umbral	108
figura 41: paso 3, clásica: iniciamos el proceso seleccionando Operaciones->Segmentar->Segmentar	108
figura 42: paso 1, mejorada: apuntamos el navegador al sitio web donde esta alojado el sistema y luego cargamos una imagen guardada en disco.	109
figura 43: paso 2, mejorada: colocamos las semillas y seleccionamos el nivel de umbral y la conectividad.	110
figura 44: paso 3: resultado de segmentar la imagen empleando la versión mejorada del algoritmo de segmentación	111
Tabla 4: comparación de versiones	112
figura 45: resultado de haber segmentado la imagen con la versión clásica del algoritmo empleando conectividad 4-vecinos, 2 semillas y umbral 10. Notamos que el proceso dejó afuera gran parte de la región. El tiempo que demoró fue 21.76 segundos	112
figura 46: resultado de haber segmentado la imagen con la versión clásica del algoritmo empleando conectividad 4-vecinos, 2 semillas y umbral 20. El proceso sigue sin poder cubrir toda la región. El tiempo que demoró fue 15.12 segundos	113
figura 47: resultado de haber segmentado la imagen con la versión clásica del algoritmo empleando conectividad 8-vecinos, 2 semillas y umbral 30. Se cubrió gran parte de la región pero continúan apareciendo "islas". El proceso demoró 14.10 segundos	113
figura 48: resultado de haber segmentado la imagen con la versión clásica del algoritmo empleando conectividad 8-vecinos, 2 semillas y umbral 50. Se cubrió gran parte de la región y las "islas" prácticamente desaparecieron. El proceso demoró 14.12 segundos.	114
figura 49: resultado de haber segmentado la imagen con la versión clásica del algoritmo empleando conectividad 8-vecinos, 2 semillas y umbral 80. El proceso agregó información innecesaria (en la parte izquierda de la imagen). El tiempo de ejecución fue 19.04 segundos.	114
figura 50: resultado de haber segmentado la imagen con la versión clásica del algoritmo empleando conectividad 8-vecinos, 2 semillas y umbral 65. El resultado es el esperado. El proceso demoró 14.10 segundos	115
figura 51: resultado de haber segmentado la imagen con la versión clásica del algoritmo empleando conectividad 8-vecinos, 4 semillas y umbral 65. El resultado es el esperado. El proceso demoró 8.62 segundos.	115
figura 52: resultado de haber segmentado la imagen con la versión mejorada del algoritmo empleando conectividad 4-vecinos, 2 semillas y umbral 10. Notamos que el proceso agregó información innecesaria (en la parte izquierda). El tiempo que demoró fue 0.15 segundos	116

figura 53: resultado de haber segmentado la imagen con la versión mejorada del algoritmo empleando conectividad 4-vecinos, 2 semillas y umbral 20. El proceso sigue agregando información innecesaria (en la parte izquierda). El tiempo de ejecución fue 0.15 segundos.	116
figura 54: resultado de haber segmentado la imagen con la versión mejorada del algoritmo empleando conectividad 8-vecinos, 2 semillas y umbral 30. El proceso agregó islas o huecos dentro de la región. El tiempo de ejecución fue 0.15 segundos.	117
figura 55: resultado de haber segmentado la imagen con la versión mejorada del algoritmo empleando conectividad 8-vecinos, 2 semillas y umbral 40. El proceso continúa generando "islas" dentro de la región	117
figura 56: resultado de haber segmentado la imagen con la versión mejorada del algoritmo empleando conectividad 8-vecinos, 2 semillas y umbral 80. La mayoría de las "islas" han desaparecido pero el resultado no es óptimo. El proceso demoró 0.14 segundos	118
figura 57: resultado de haber segmentado la imagen con la versión mejorada del algoritmo empleando conectividad 8-vecinos, 2 semillas y umbral 100. La mayoría de las islas han desaparecido pero el resultado sigue sin ser óptimo. El proceso demoró 0.15 segundos	118
figura 58: resultado de haber segmentado la imagen con la versión mejorada del algoritmo empleando conectividad 8-vecinos, 2 semillas y umbral 120. Vemos que la posición de las semillas hace variar el resultado final. El proceso demoró 0.15 segundo	119
figura 59: resultado de haber segmentado la imagen con la versión mejorada del algoritmo empleando conectividad 8-vecinos, 2 semillas y umbral 120. Las islas han desaparecido, el resultado es el esperado. El proceso demoró 0.15 segundos	119
figura 60: resultado de haber segmentado la imagen con la versión mejorada del algoritmo empleando conectividad 8-vecinos, 3 semillas y umbral 120. Las islas han desaparecido, el resultado es el esperado. El proceso demoró 0.23 segundos.	120
Tabla 5: Tabla comparativa entre versión clásica y mejorada del algoritmo de segmentación de imágenes por crecimiento de regiones	120
figura 61: tomografía computada del cerebro	122
Tabla 6: conjunto de resultados obtenidos segmentando la imagen que aparece en la figura 61 haciendo variar la cantidad de semillas, posición de las semillas, nivel de umbral y conectividad empleando la versión clásica del algoritmo. Los mejores resultados fueron resaltados.	122
Tabla 7: conjunto de resultados obtenidos segmentando la imagen que aparece en la figura 61 haciendo variar la cantidad de semillas, posición de las semillas, nivel de umbral y conectividad empleando la versión mejorada del algoritmo. Los mejores resultados fueron resaltados.	128
Tabla 8: subconjunto de los mejores resultados obtenidos en la tabla 6, empleado la versión clásica del algoritmo.	131
Tabla 9: subconjunto de los mejores resultados obtenidos en la tabla 6, empleado la versión clásica del algoritmo.	131

BIBLIOGRAFÍA:

- Image processing – Principles and Applications – Tinku Acharya and Ajoy Ray – Wiley Interscience.
- Digital Image Processing – a practical introduction using java – Nick Efford – Pearson (2000).
- Digital Image Processing: a practical introduction using Java – Nick Efford – Pearson.
- Estadística Elemental – Mario M. Triola – 7ma edición – Pearson.

- Atlas de bolsillo de cortes anatómicos de Tomografía Computada y Resonancia Magnética – Torsten B. Möller y Emil Reif – Vol. 1: cabeza, cuello, columna vertebral y articulaciones – 2da edición Editorial Médica Panamericana.
- Revistas “Medical Imaging International” ISSN 1068-1779.
- Internet:
 - <http://dac.escet.urjc.es/docencia/IM/>
 - http://www.msct.info/CT_Quality_Criteria.htm#Download%20the%202004%20CT%20Quality%20Criteria
 - <http://www.intramed.net/log.asp?retorno=/home.asp>
 - http://www.dfmf.uned.es/actividades/no_reglada/laboratorio/segmentacion1.pdf
 - <http://www.pacsone.net/>
 - <http://www.k-pacs.net/>
 - <http://medical.nema.org/> (dicom home page)
 - <http://imaging.apteryx.fr/dicom/>
 - <http://www.psychology.nottingham.ac.uk/staff/cr1/dicom.html> (Links to free DICOM viewers)
 - <http://www.psychology.nottingham.ac.uk/staff/cr1/index.html>
 - <http://www.divinev.com/radscaper/> (applet viewer)
 - <http://www.tiani.com/JDicom/> (server client)
 - http://www.dfmf.uned.es/actividades/no_reglada/imagen_medica/
- Papers:
 - [1] Fast marching methods for the continuous traveling salesman problem
June Andrews, and J. A. Sethian
doi:10.1073/pnas.0609910104
PNAS 2007;104;1118-1123; originally published online Jan 12, 2007
 - [2] Level set method for pulmonary vessels extraction
Zhang hongmei Bian zhengzhong Jiang dazong Yuan zejian Ye min
Sch. of Life Sci. & Technol., Xi'an Jiaotong Univ., China;
This paper appears in: Image Processing, 2003. ICIP 2003. Proceedings. 2003 International Conference on
Publication Date: 14-17 Sept. 2003

Volume: 2, On page(s): II- 1105-8 vol.3

ISSN: 1522-4880

ISBN: 0-7803-7750-8

INSPEC Accession Number: 7978720

Digital Object Identifier: 10.1109/ICIP.2003.1246879

Posted online: 2003-11-24 09:36:36.0

- [3] IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 10, NO. 2, FEBRUARY 2001 Active Contours Without Edges Tony F. Chan, Member, IEEE, and Luminita A. Vese
- [4] Zhang, Y.J.: A survey on evaluation methods for image segmentation. Pattern Recognition 29 (1996) 1335–1346