

# Capítulo

# 7

## Pruebas del método *Snake* sobre imágenes RM pre- procesadas.

El objetivo del presente capítulo, es dejar al descubierto el impacto que tiene el pre-procesamiento propuesto en el capítulo anterior sobre imágenes RM de la región de la cabeza, en el desempeño del *Snake*, tanto con el campo externo tradicional como el *GVF*.

Las pruebas se realizaron sobre las mismas imágenes médicas con las que se trabajó en los capítulos 5 y 6 (RM1 y RM2), de resonancia magnética. La metodología consiste en que cada prueba se contrastará con su equivalente del capítulo 5, y se evaluarán 3 aspectos:

1. Tiempo de procesamiento.
2. Cantidad de reajustes de los parámetros.
3. Detección del borde.

## 7.1 Transformada de contraste aplicada en la imagen RM1

Para establecer los parámetros de entrada de la transformación *imadjust* aplicada en RM1, nos basamos en la distribución del campo de fuerzas externas. Para este tipo de imagen donde la información de interés está en la región central, nuestro objetivo fue hacer homogéneo el fondo oscuro que rodea al contorno exterior, con el propósito de que el gradiente en esa región fuese nulo y no perturbara el funcionamiento del *Snake*. Haciendo un análisis de los valores de los píxeles, en el capítulo 6, se mostró que el valor máximo de intensidad de los que conforman la región del fondo es de 45, lo cual equivale a 0.0007, si consideramos que el valor 65535 (dado que se tienen 16 bits por píxel, por ser una imagen en el formato DICOM) se normaliza en 1. Para el límite superior se puede ser menos exacto; la única condición que debe de cumplir el límite superior es que incluya al nivel máximo de intensidad de la imagen. En caso de RM1 este se ubica en el valor 1289, por lo que colocarlo en 0.6 se considera un valor adecuado. La ecuación 7.1 muestra los parámetros que se utilizan en la transformación *imadjust*:

$$g = \text{imadjust}(j, [0.0007 \ 0.6], [0 \ 1], 0.9) \quad (7.1)$$

donde:

*j*: imagen de entrada.

*g*: imagen de salida.

Con el fin de completar el análisis e ilustrar el efecto de la transformación *imadjust* sobre la homogeneidad de algunas regiones, se tomaron ciertos valores alrededor de 0.0007 y los resultados se muestran a continuación.

| Rango de valores de intensidad de entrada | Argumento de la función <i>imadjust</i> (valores de intensidad de entrada divididos entre 65535). | Figura | Observaciones  |
|---|---|--------|--|
| [15 39321]                                | [0.00023 0.6]   | 7.1a   | Se aprecia de una manera más clara la forma de la información de interés, sin embargo, el ruido aún impide ver detalles de la misma.   |
| [25 39321]                                | [0.0004 0.6]  | 7.1b   | El ruido en el fondo se ha reducido pero aún quedan ciertos vectores del campo lejos del área de interés.  |
| [35 39321]                                | [0.00053 0.6]   | 7.1c   | Las líneas de campo fuera de la información de interés son mínimas.  |
| [45 39321]                                | [0.0007 0.6]  | 7.1d   | El área alrededor de la información de interés se aprecia libre de ruido y se observa poca pérdida de información, por lo tanto este intervalo es el que tomaremos como óptimo para el desarrollo de las siguientes pruebas. |
| [55 39321]                                | [0.00084 0.6]   | 7.1e   | La pérdida de información útil continúa.   |
| [65 39321]                                | [0.001 0.6]   | 7.1f   | Se aprecia pérdida de información y disminución de la densidad del campo.  |
| [75 39321]                                | [0.0011 0.6]  | 7.1g   | La pérdida de información útil, no es significativa pero produce pérdida de detalles de la imagen.   |

Tabla 7.1 Intervalos de niveles de intensidad de entrada y observaciones del resultado.

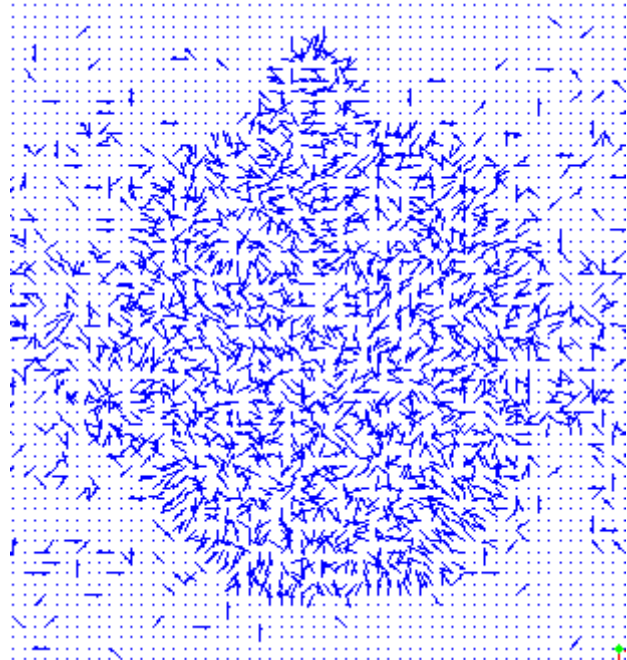
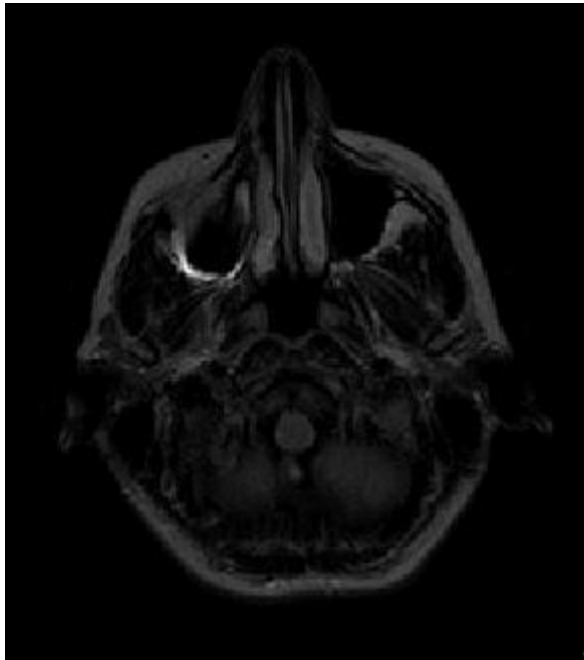


Imagen de salida.

Campo de fuerzas.

(a)  $g = imadjust(j, [0.00023 0.6], [0 1], 0.9)$ .

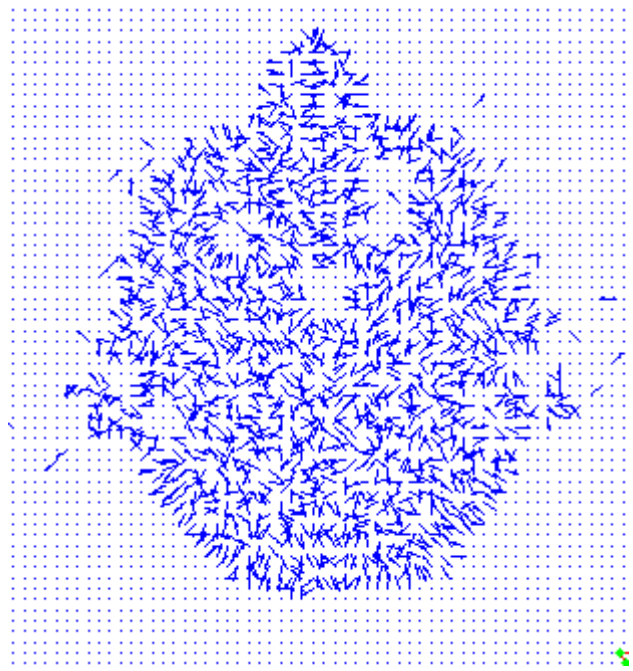
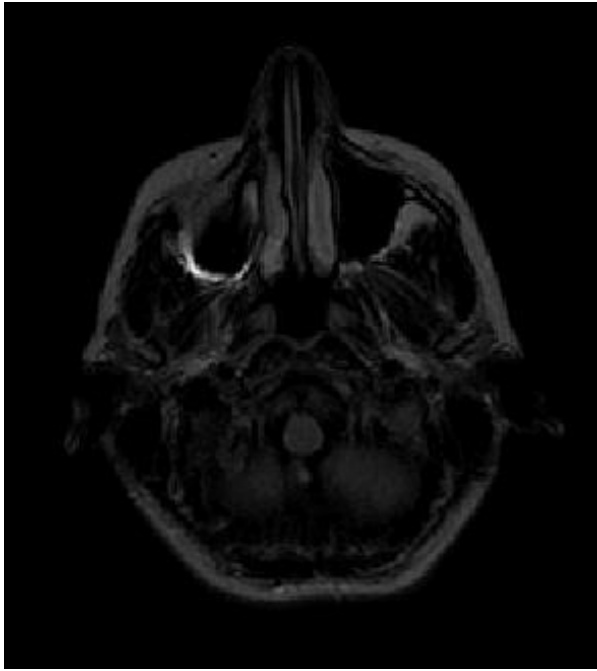


Imagen de salida.

Campo de fuerzas.

$$(b) \ g = imadjust(j, [0.0004 \ 0.6], [0 \ 1], 0.9).$$

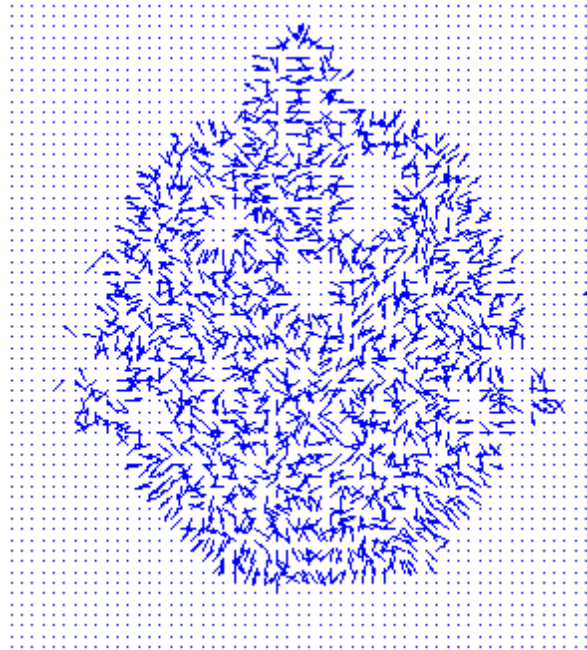
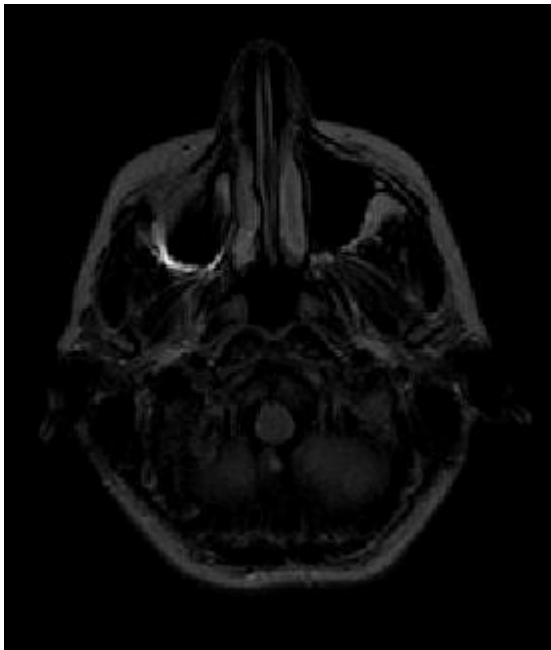


Imagen de salida.

Campo de fuerzas.

$$(c) \ g = imadjust(j, [0.00053 \ 0.6], [0 \ 1], 0.9)$$

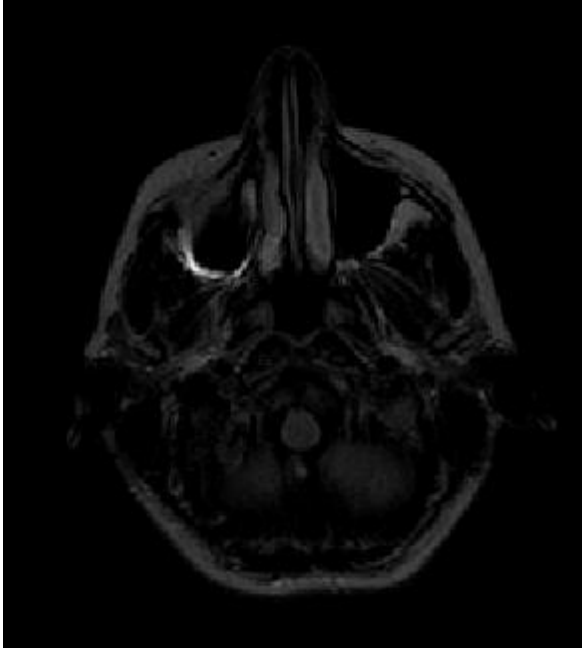
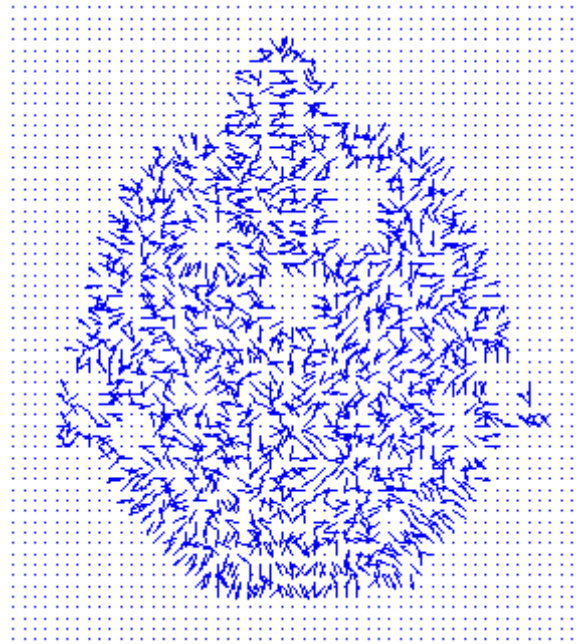


Imagen de salida.



Campo de fuerzas.

(d)  $g = imadjust(j, [0.0007 \ 0.6], [0 \ 1], 0.9)$ .

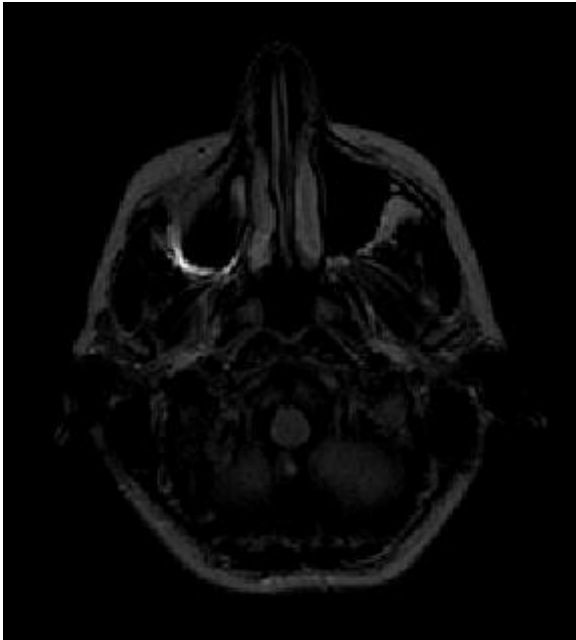
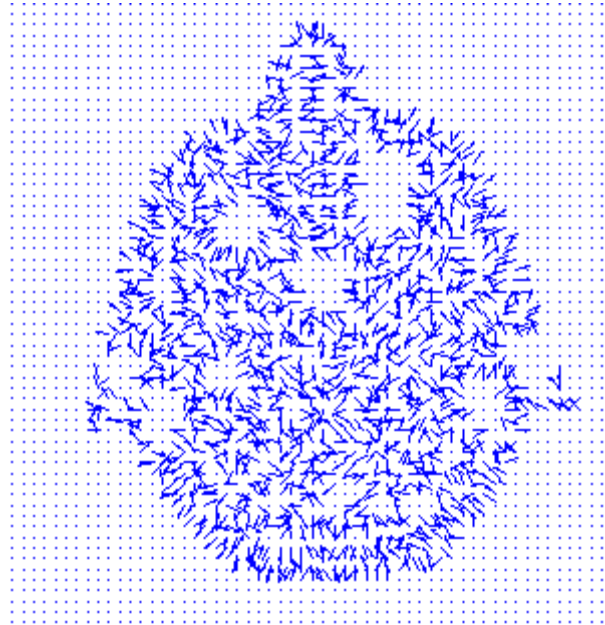


Imagen de salida.



Campo de fuerzas.

(e)  $g = imadjust(j, [0.00084 \ 0.6], [0 \ 1], 0.9)$ .

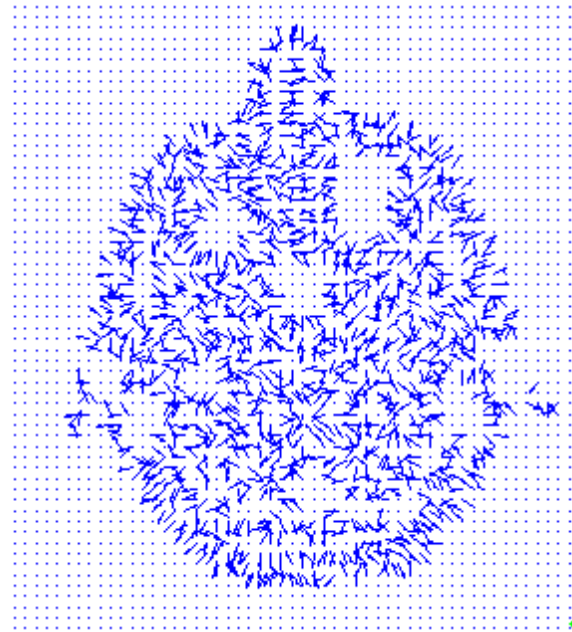


Imagen de salida.

Campo de fuerzas.

$$(f) g = imadjust(j, [0.001 \ 0.6], [0 \ 1], 0.9).$$

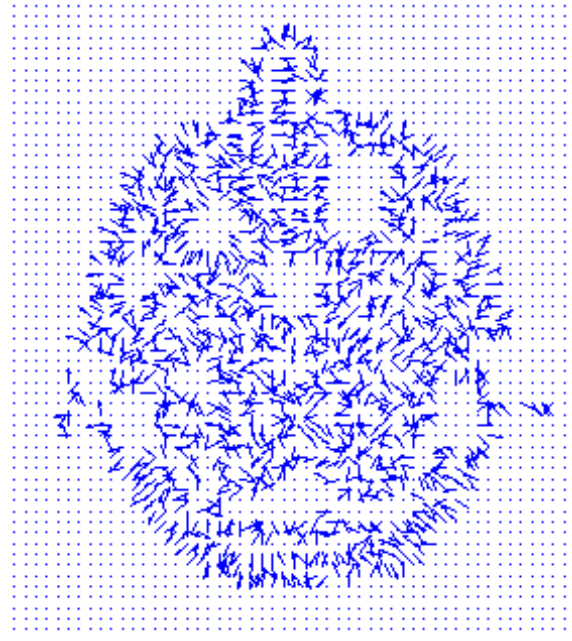
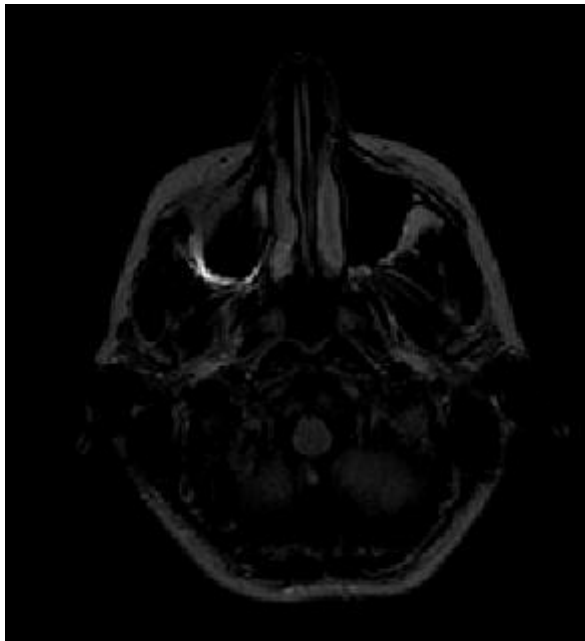


Imagen de salida.

Campo de fuerzas.

$$(g) g = imadjust(j, [0.0011 \ 0.6], [0 \ 1], 0.9).$$

Figura 7.1 Impacto de *imadjust* sobre el mapa de bordes de la imagen RM1.

Las pruebas realizadas modificando los límites de la transformación ilustran los cambios que sufre el campo de fuerzas externas al modificar el intervalo de niveles de intensidad de entrada, el ruido contenido en el fondo de la información de interés desaparece, y se forman regiones homogéneas en ciertas áreas de la imagen. Sin embargo, se puede observar que esta etapa de pre-procesamiento además impacta otras áreas, por ejemplo, ambos ojos. Como en todo método de filtrado, el costo de esta técnica es la posible pérdida de información, que de ser importante, se reflejará en las siguientes pruebas. El caso de la figura 7.1(d), fue utilizado para las pruebas del *Snake* tradicional y del *GVF*, por presentar homogeneidad sin caer en un exceso de pérdida de información de interés.

### **7.1.1 Snake tradicional sobre la imagen RM1 pre-procesada**

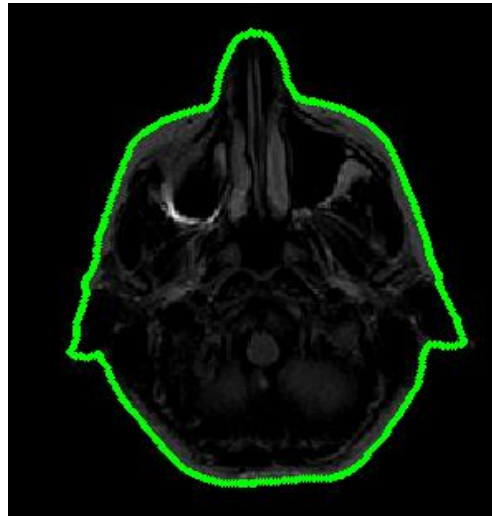
Todas las pruebas de este capítulo fueron análogas a las de la sección 5.2 en cuanto a: regiones segmentadas, posición inicial del contorno activo, parámetros iniciales y la secuencia de reajuste de los parámetros; la imagen con la que se trabajó es la mostrada en la figura 7.1(d). Cabe resaltar que esto se hizo con el fin de contar con una referencia para contrastar los resultados, y poder una evaluación integral de lo que es trabajar con imágenes pre-procesadas en base a la transformación *imadjust*.

#### **7.1.1.1 Contorno exterior**

En esta ocasión no fue necesario recurrir al reajuste de parámetros, los iniciales fueron adecuados para que el Snake detectara al borde de interés, tal y como se observa en la figura 7.2, la pérdida de información fue mínima, ya que permitió recuperar la forma del borde de interés de una manera muy bien aproximada.

El uso de una etapa de pre-procesamiento mejoró el tiempo de procesamiento, ya que se alcanzó un resultado aceptable con tan solo 500 iteraciones; redujo las ocasiones en que el usuario debió intervenir, permitió detectar las regiones de la nariz y las orejas, aunque estas últimas con cierta atenuación.





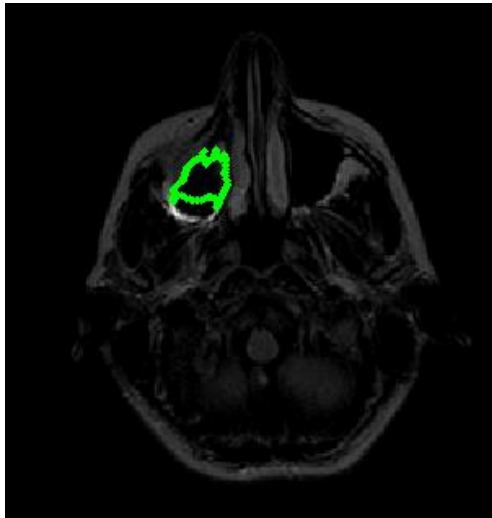
Parámetros iniciales:  
 $\alpha = 1, \beta = 0$  e iteraciones = 500.

*Figura 7.2* Segmentación del borde exterior de RM1 pre-procesada con *Snake* tradicional.

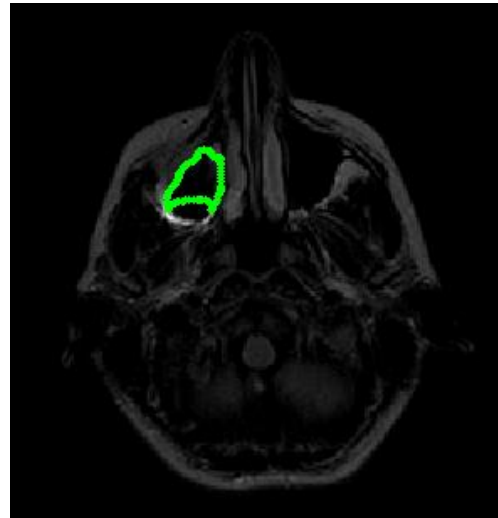
#### 7.1.1.2 Región del ojo izquierdo

En este caso se notó que el desempeño del *Snake* también evolucionó adecuadamente en cuanto al tiempo de procesamiento, ya que solo fueron necesarias 400 iteraciones para obtener el resultado final, en comparación con las 500 iteraciones utilizadas para obtener el resultado de la figura 5.18; solo en ese aspecto se alcanzó una diferencia respecto a los resultados anteriores.

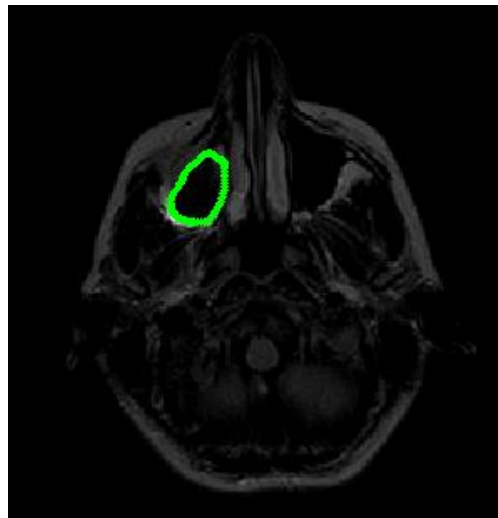




(a) Parámetros iniciales:  
 $\alpha = 0.05, \beta = 0$  e iteraciones = 100.



(b) Primer ajuste de parámetros:  
 $\alpha = 0.05, \beta = 1$  e iteraciones = 100.



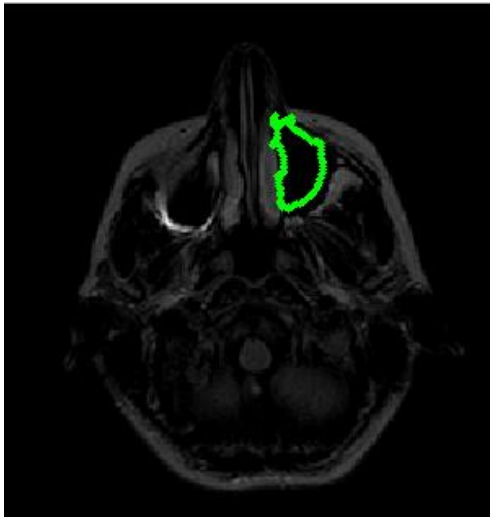
(c) Segundo ajuste de parámetros:  
 $\alpha = 1, \beta = 1$  e iteraciones = 200.

Figura 7.3 Segmentación del borde del ojo izquierdo de RM1 pre-procesada con *Snake* tradicional.

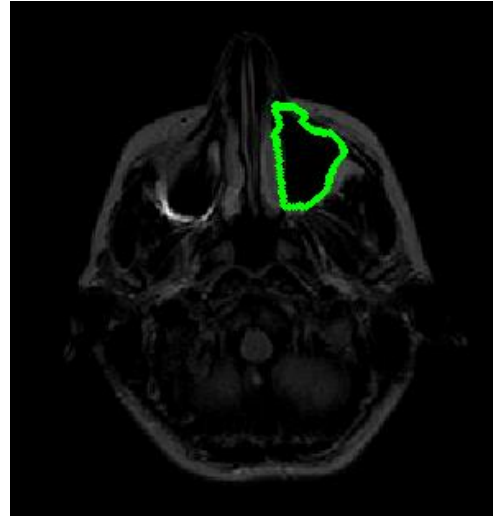
### 7.1.1.3 Región del ojo derecho

Contrariamente al caso anterior en esta ocasión se incrementó el número de iteraciones, al requerir 800 para llegar a un resultado aceptable, a diferencia de las pruebas sin pre-procesamiento que requirieron solo 500; consideramos que esto se debió a que la región rodeada por este borde se hizo homogénea, como resultado de la transformación *imadjust*, lo que provocó la desaparición de ciertas líneas de campo externo que sirvieron

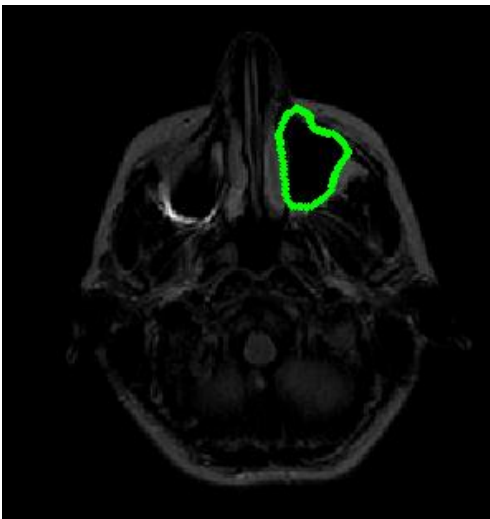
de “empuje” para lograr el resultado de la figura 5.19. Por esta razón en este experimento se incrementó el número de iteraciones, para que el *Snake* lograra censar el campo que lo guiara al borde.



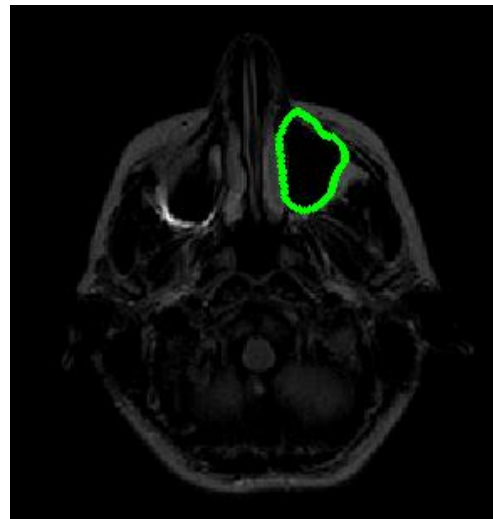
(a) Parámetros iniciales:  
 $\alpha = 0.05, \beta = 0$  e iteraciones = 100.



(b) Primer ajuste de parámetros:  
 $\alpha = 0.05, \beta = 1$  e iteraciones = 500.



(c) Segundo ajuste de parámetros:  
 $\alpha = 1, \beta = 1$  e iteraciones = 100.



(d) Tercer ajuste de parámetros:  
 $\alpha = 2, \beta = 1$  e iteraciones = 100.

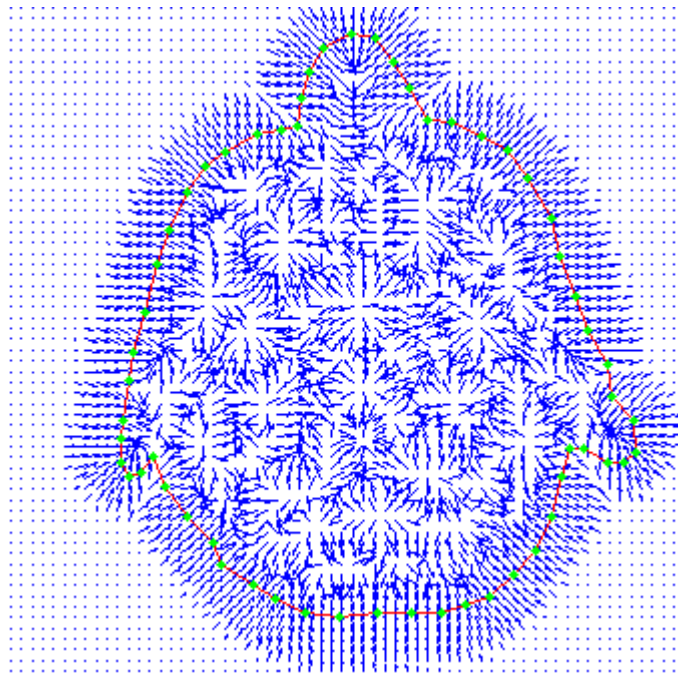
Figura 7.4 Segmentación del borde del ojo derecho de RM1 pre-procesada con *Snake* tradicional.

## 7.1.2 Snake GVF trabajando con la imagen RM1 pre-procesada

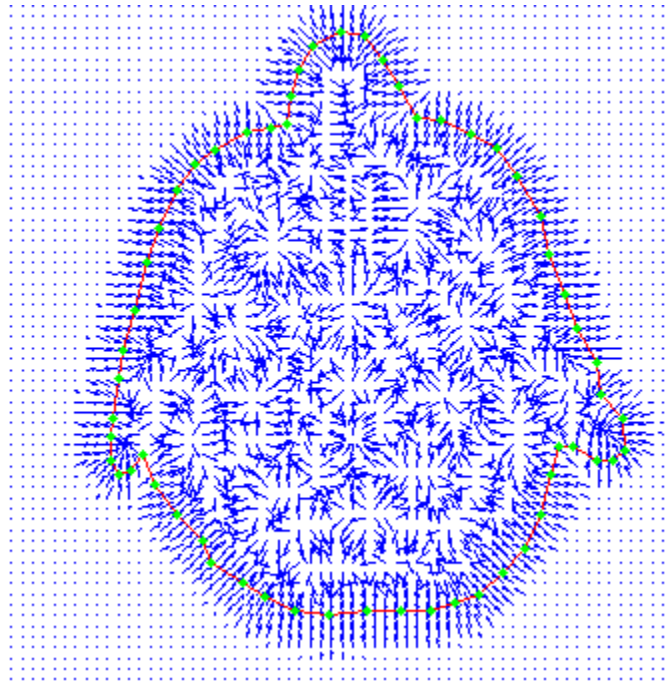
### 7.1.2.1 Contorno exterior

De la misma manera se reprodujeron las pruebas de la sección 5.2.2.1 para apreciar, tanto mejoras, como retrocesos. En la figura 7.5 se aprecia que trabajar con imágenes RM que presentan mayor homogeneidad en algunas regiones beneficia al proceso de difusión, ya que el área de cobertura del campo *GVF* se incrementa y de forma aún más uniforme respecto a la figura 5.25, ya que la mayor parte del campo apunta hacia el borde de interés.

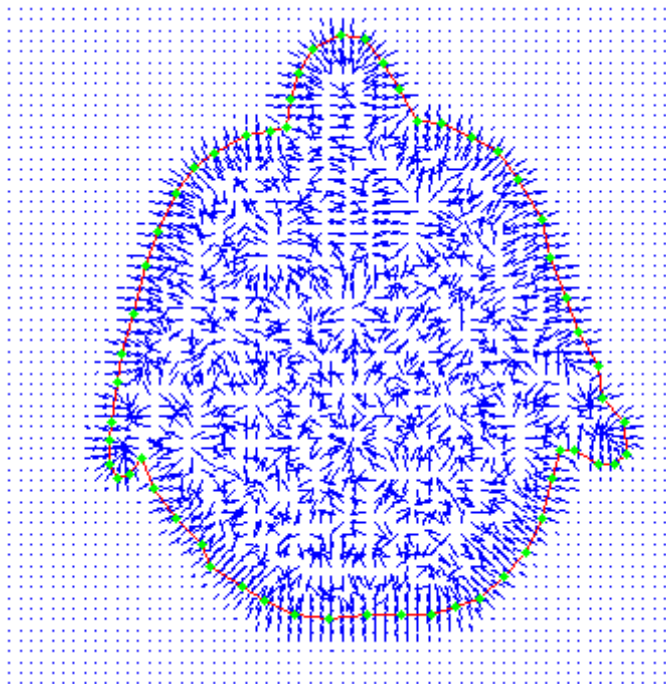
También aquí se utilizó  $\mu = 0.02$  como en el punto 5.2.2.1.1, y el resultado del *Snake* se presenta en la figura 7.6. En esta ocasión se aprecia cierta pérdida de precisión en la detección del borde, a pesar de que la nariz y las orejas fueron detectadas; además de que el número de iteraciones se incrementó. Debido a que no obtuvimos resultados satisfactorios, primero intentamos manipular los parámetros para mejorar el rendimiento, sin embargo, esta opción nos llevó nuevamente a resultados no deseados, así que optamos por ajustar otro parámetro, disminuyendo  $\mu$  hasta 0.002, el resultado se presenta en la figura 7.7.



(a)  $\mu = 0.1$  e iteraciones = 80.

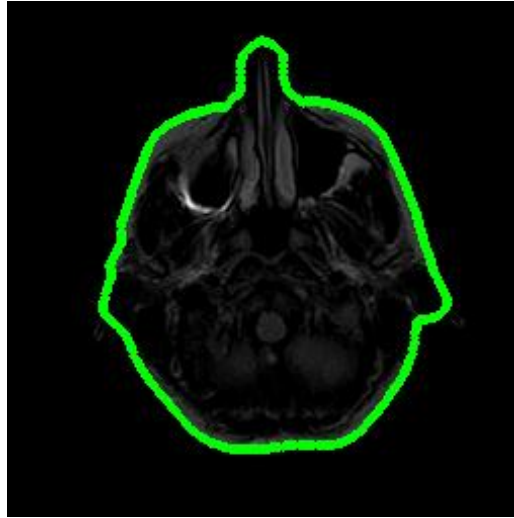


(b)  $\mu = 0.05$  e iteraciones = 80.



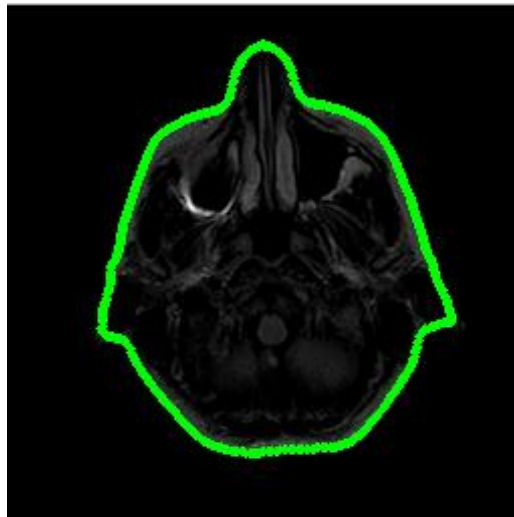
(c)  $\mu = 0.02$  e iteraciones = 80.

Figura 7.5 Manipulación de  $\mu$  para el cálculo del GVF para RM1 pre-procesada.



Parámetros iniciales:  
 $\alpha = 3, \beta = 0$  e iteraciones = 200.

Figura 7.6 Segmentación del borde exterior de RM1 pre-procesada con *Snake GVF*.



*GVF*:  $\mu = 0.002$  e iteraciones = 80.  
 $\alpha = 3, \beta = 0$  e iteraciones = 100.

Figura 7.7 Segmentación del borde exterior de RM1 pre-procesada con *Snake GVF* ( $\mu = 0.002$ ).

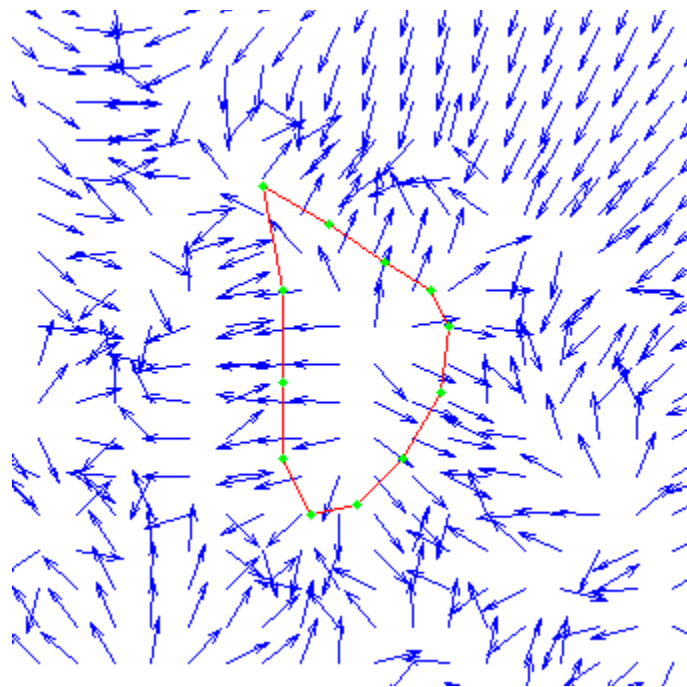
Consideramos que la segmentación alcanzada y mostrada en la figura 7.7 es similar a la obtenida en la figura 5.26. Entonces, en este caso fue necesario disminuir  $\mu$  de tal manera que se pudiese reducir el área de cobertura y densidad del campo *GVF* y conservar así más detalles del borde en estudio. Siguiendo con el análisis de los valores de los

parámetros, nos podemos preguntar ¿por qué el valor de  $\mu=0.02$  sí funcionó en la imagen sin pre-procesamiento, y en este último experimento no? Conociendo el modo de operar del método, la respuesta puede ser muy sencilla: las zonas no homogéneas por su naturaleza, también generan un flujo de campo vectorial durante el proceso de difusión, que al encontrarse con el flujo generado por un borde, frenan la propagación de este último. Por esa razón, cuando se reutilizó el mismo valor de  $\mu$  en la imagen RM1 pre-procesada que se caracteriza por poseer una mayor homogeneidad, disminuyeron las componentes de campo que frenan la propagación, por lo que se alcanzó una mayor área de cobertura, que borró algunos detalles finos del borde, principalmente en la región de la nariz.

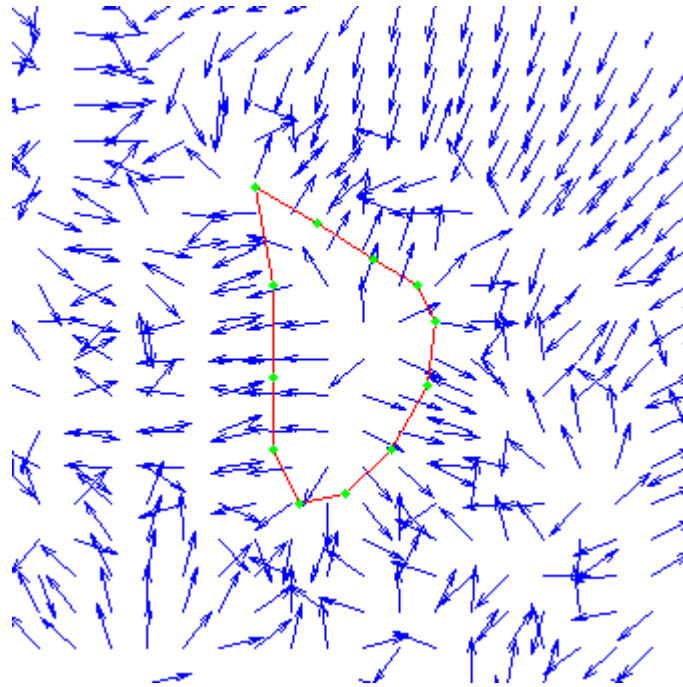
### 7.1.2.2 Región del ojo derecho

La figura 7.8 muestra la forma en la que se comportó el campo *GVF* al variar  $\mu$  en la imagen RM1 pre-procesada en la región del ojo derecho.

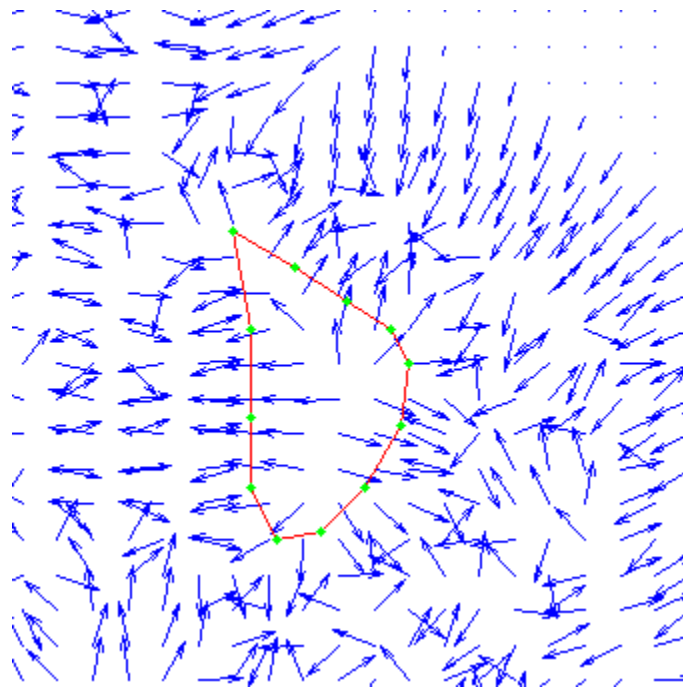
La figura 7.9 muestra el resultado obtenido con RM1 pre-procesada con  $\mu = 0.05$ , en esta ocasión, no se apreció ningún cambio tangible en cuanto a los elementos que se han manejado para comparar las dos pruebas.



(a)  $\mu = 0.1$  e iteraciones = 80.



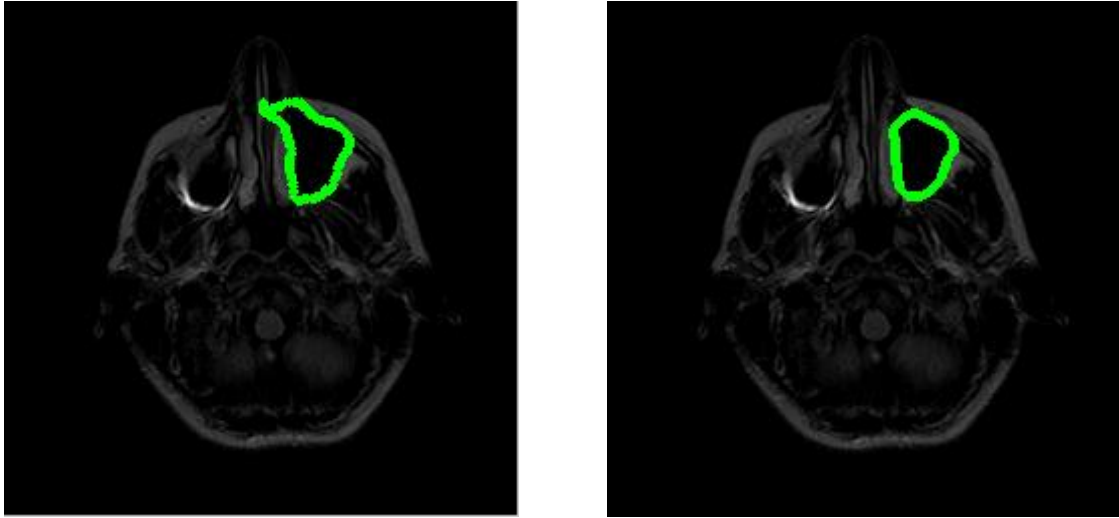
(b)  $\mu = 0.05$  e iteraciones = 80.



(c)  $\mu = 0.02$  e iteraciones = 80.

Figura 7.8 Manipulación de  $\mu$  para el cálculo del GVF para RM1 pre-procesada, centrado en la región del ojo derecho.





(a) Parámetros iniciales:  
 $\alpha = 0.05, \beta = 0$  e iteraciones = 100.

(b) Primer ajuste de parámetros:  
 $\alpha = 4, \beta = 0$  e iteraciones = 100.

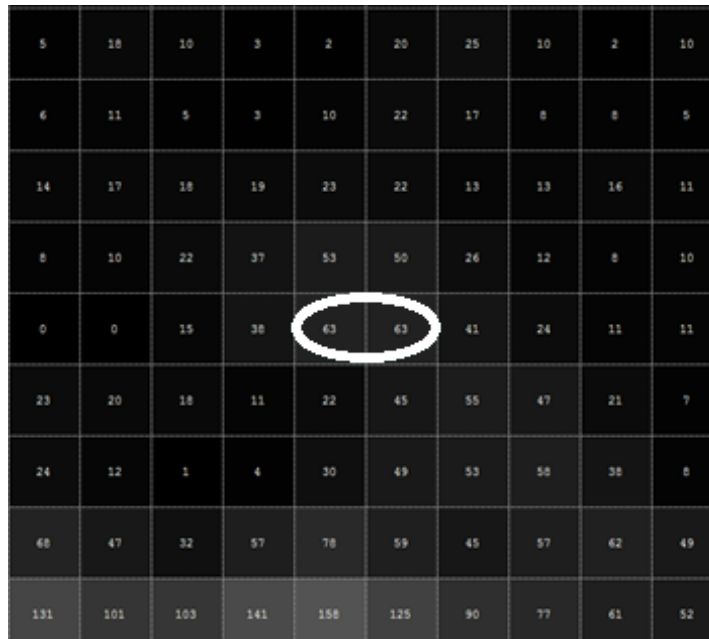
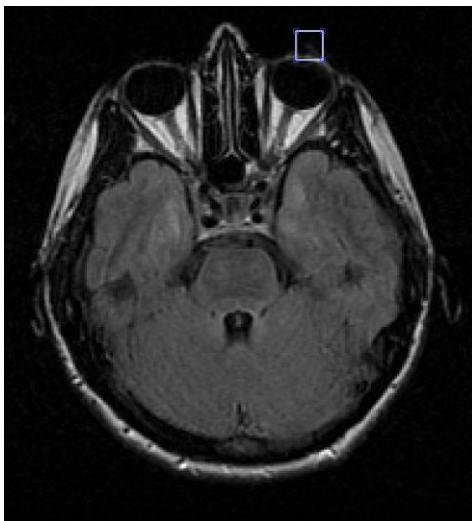
Figura 7.9 Segmentación del borde del ojo derecho de RM1 pre-procesada con *Snake GVF*.

## 7.2 Transformada de contraste aplicada en la imagen RM2

Para la imagen RM2 se aplicó una transformación de intensidad, análoga a la de la imagen RM1. Para determinar el parámetro de entrada de la transformación *imadjust*, nivel mínimo de intensidad de entrada, igualmente fue necesario buscar alrededor del contorno externo el valor más alto de intensidad, con el fin de hacer homogénea esta región. La búsqueda de píxel a píxel permitió establecer que el nivel mínimo de entrada debía fijarse en 63, lo que equivale a 0.00096. La figura 7.10 señala la región encerrada en un recuadro, donde se tiene dicho valor.

La ecuación 7.2 muestra la configuración final de la transformación *imadjust*. La figura 7.11 ilustra el impacto de la transformación de contraste sobre el comportamiento del campo externo tradicional.

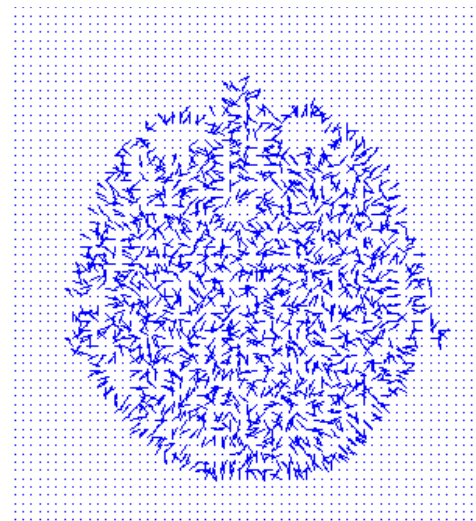
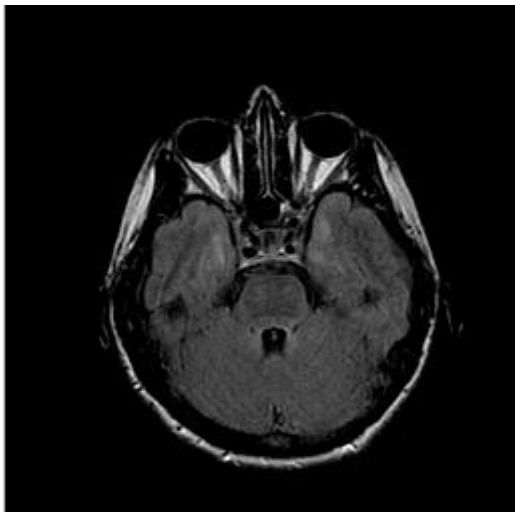
$$g = imadjust(j, [0.00096 \ 0.6], [0 \ 1], 0.9) \quad (7.2)$$



(a) Región donde se ubica el valor con máximo nivel de gris alrededor del contorno exterior.

(b) Acercamiento a la región de estudio.

Figura 7.10 Máximo nivel de gris de la imagen RM2.



(a) RM2 después de aplicar *imadjust*.

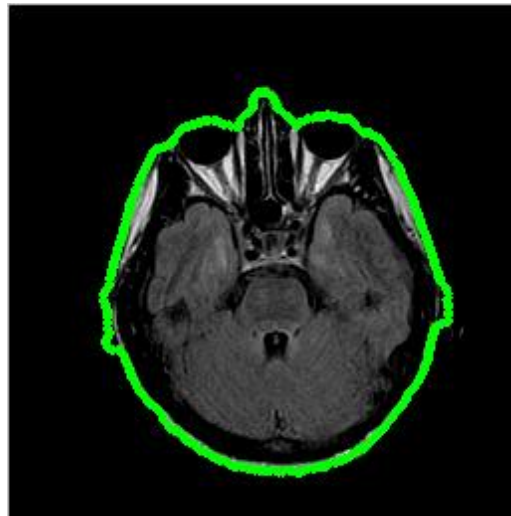
(b) Campo de fuerzas.

Figura 7.11 Impacto de *imadjust* sobre el campo externo de RM2.

## 7.2.1 *Snake* tradicional sobre la imagen RM2 pre-procesada

### 7.2.1.1 Contorno exterior

La figura 7.12 muestra que se alcanzó una mejor segmentación que la mostrada en la figura 5.22, sin embargo, aún presenta problemas en la detección de la región de las orejas, debido a la atenuación del borde. Además se tuvieron mejores resultados en el tiempo de procesamiento ya que se redujo de 3100 a 700 iteraciones; y en el número de intervenciones del usuario.

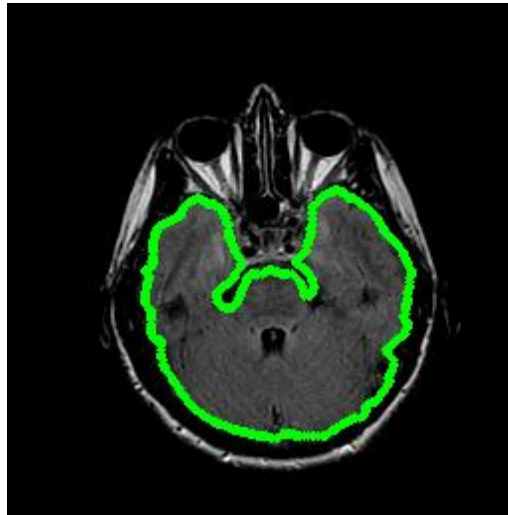


*Parámetros iniciales:*  
 $\alpha = 1, \beta = 0$  e iteraciones = 700.

*Figura 7.12* Segmentación del borde exterior de RM2 pre-procesada con *Snake* tradicional.

### 7.2.1.2 Región del cerebro

En esta prueba se redujo la cantidad de reajustes de los parámetros, el número de iteraciones y sobre todo, se obtuvo una detección más detallada del borde en estudio, el resultado se muestra en la figura 7.13.

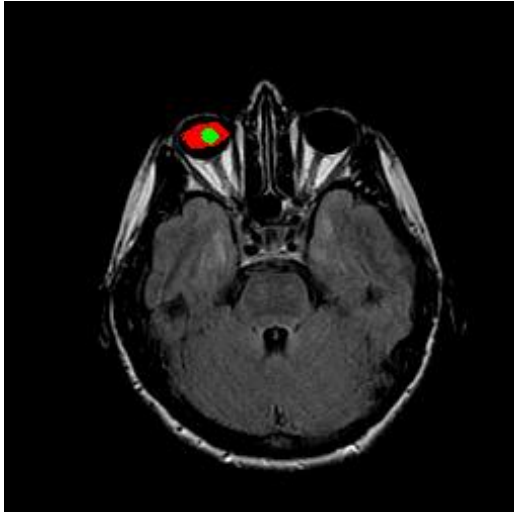


Parámetros iniciales:  
 $\alpha = 0.05, \beta = 1$  e iteraciones = 100.

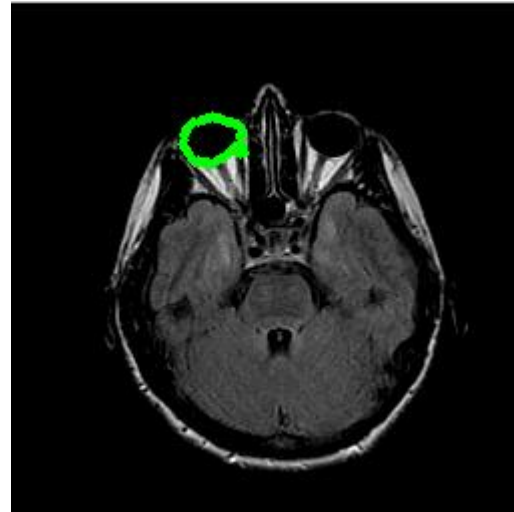
*Figura 7.13* Segmentación del borde del cerebro de RM2 pre-procesada con *Snake* tradicional.

### 7.2.1.3 Región del ojo izquierdo

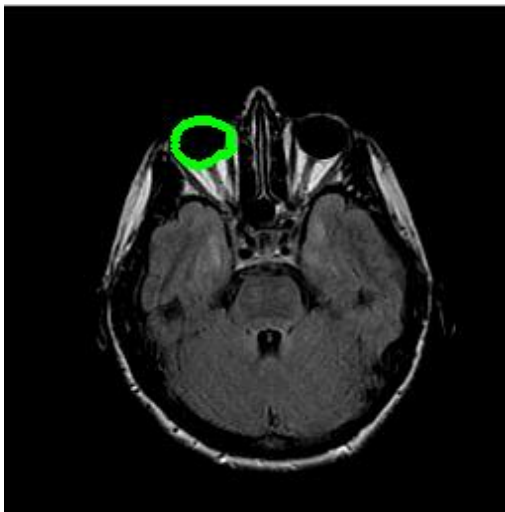
En esta prueba los parámetros iniciales resultaron inconvenientes porque se colapsó al contorno activo, figura 7.14(a). En consecuencia, fue necesario reiniciar la posición del *Snake* y reajustar los parámetros, figura 7.14(b); estos mostraron mejor respuesta, y fueron complementados con los ajustes de (c) y (d) de la misma figura. Al comparar los resultados de las figuras 5.24(c) y 7.14(d), apreciamos un avance en esta última, porque el contorno se ha ovalado, sin llegar a ser tan pronunciado como el de la figura 5.24(b). En cuanto al número de iteraciones, aumentaron al doble.



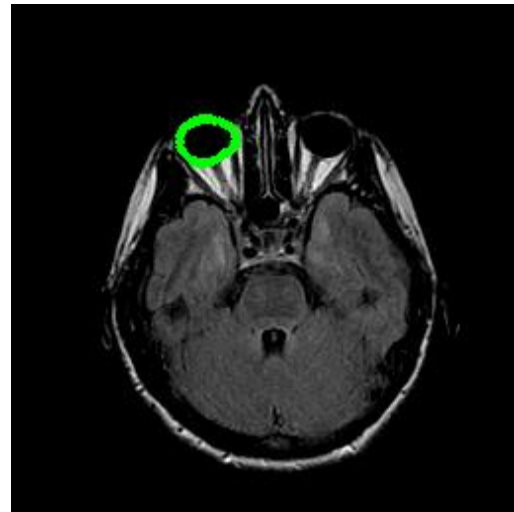
(a) Parámetros iniciales:  
 $\alpha = 1, \beta = 0$  e iteraciones = 75



(b) Reinicio de la posición del Snake con los parámetros:  
 $\alpha = 0.05, \beta = 0$  e iteraciones = 900



(c) Primer ajuste de parámetros:  
 $\alpha = 1, \beta = 1$  e iteraciones = 100



(d) Segundo ajuste de parámetros:  
 $\alpha = 2, \beta = 2$  e iteraciones = 200

Figura 7.14 Segmentación del borde del ojo izquierdo de RM2 con Snake tradicional.

## 7.2.2 Snake GVF trabajando con la imagen RM2 pre-procesada

### 7.2.2.1 Contorno exterior

Para esta prueba nuevamente se utilizó el valor de  $\mu = 0.02$  al igual que los parámetros iniciales. Se tuvo una mejora en los 3 aspectos que se han estado evaluando

durante el presente capítulo: el número de iteraciones se redujo de 420 a 40, ya no fue necesario ningún tipo de ajuste de parámetros y la segmentación en cuanto a forma, tuvo un ligero avance, ya que el *Snake* logró censurar la presencia de la región de la oreja derecha. Las figuras 7.15 y 7.16 muestran el campo *GVF* y la posición final del contorno activo respectivo.

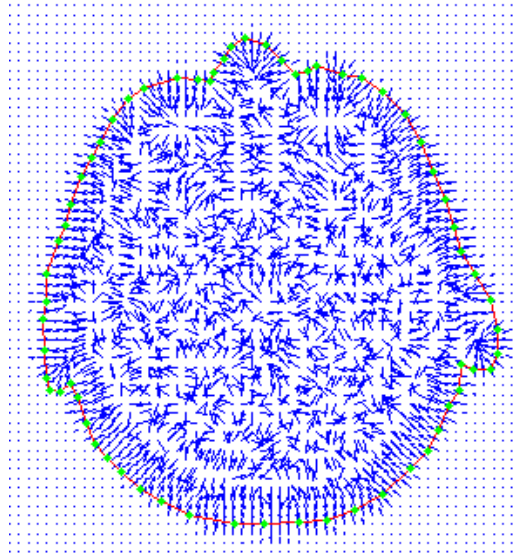
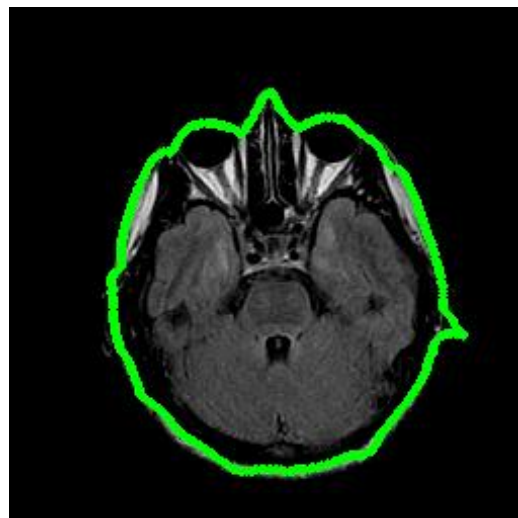


Figura 7.15 Campo *GVF* de la imagen RM2 pre-procesada con  $\mu = 0.02$ .



Parámetros iniciales:  
 $\alpha = 1, \beta = 1$  e iteraciones = 40.

Figura 7.16 Segmentación del borde exterior de RM2 con *Snake GVF*.

### 7.2.2.2 Región del cerebro

Se tuvieron dos mejoras alcanzadas con RM pre-procesada: la reducción de la cantidad de iteraciones, y la más importante, el *Snake* logró detectar el borde de una forma más fiel, ya que se conservan detalles finos del borde, figura 7.18.

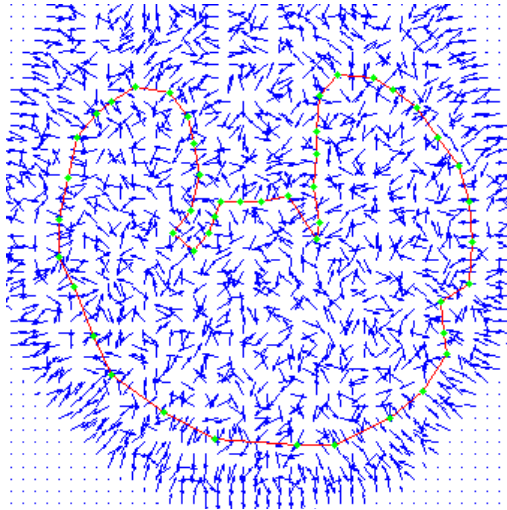
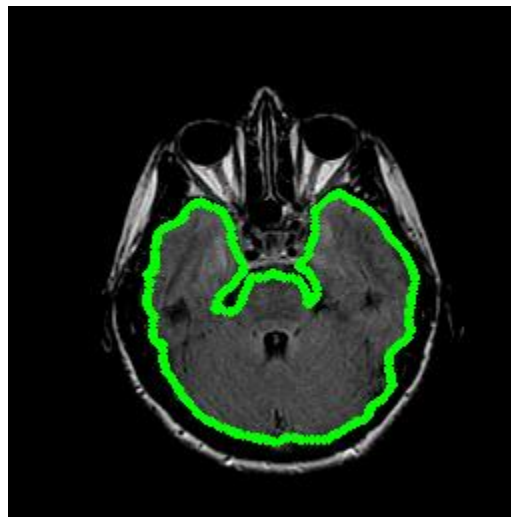


Figura 7.17 Campo GVF de la imagen RM2 con  $\mu = 0.002$ , centrado en la región del cerebro.



Parámetros iniciales:  
 $\alpha = 0.05, \beta = 1$  e iteraciones = 50.

Figura 7.18 Segmentación del borde del cerebro de RM2 pre-procesada con *Snake GVF*.