

Práctica 3

Imagen médica

Por: Álvaro Gómez Haro

Ejercicio A

A continuación se muestran los resultados del algoritmo de crecimiento de regiones. Primeramente en la figura 1 la imagen a segmentar:

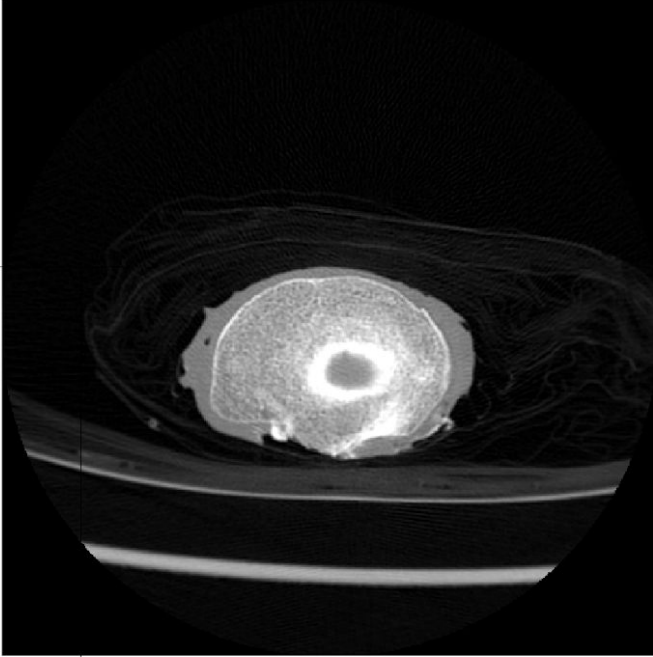
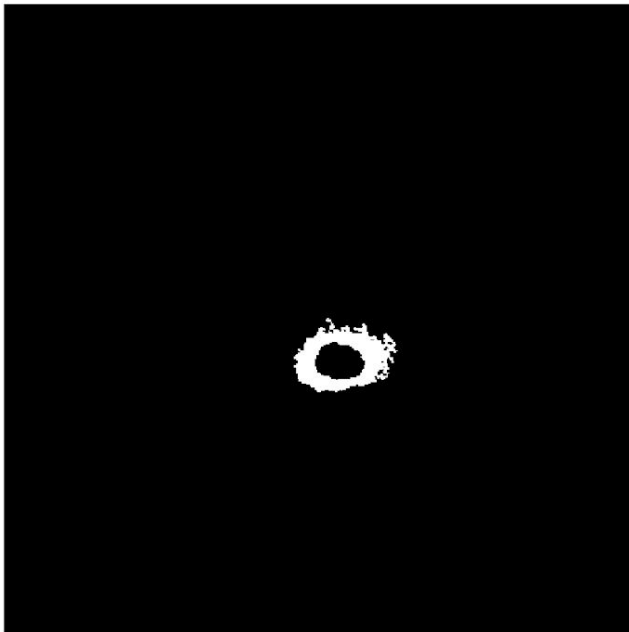
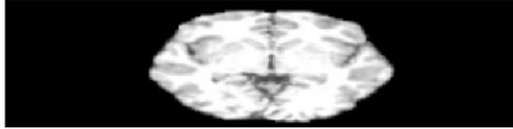


Imagen segmentada (figura 2):

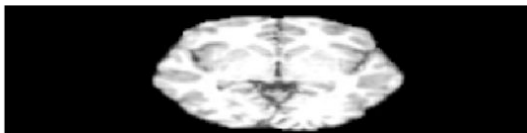


Ejercicio B

Comenzamos con la imagen original segmentada con 4 clases (figura 3):

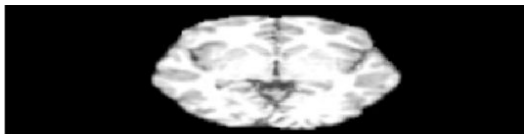


Continuamos con la segmentación con 6 clases en la figura 4:



La segmentación de 4 clases es bastante más efectiva, debido a la distribución más adecuada de las gaussianas.

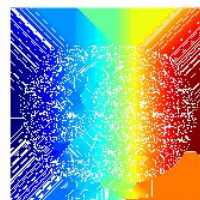
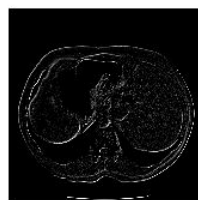
A continuación, modificamos el algoritmo para inicializarse con las gaussianas en los valores indicados, los resultados son mucho mejores como se ve en la figura 5:



Además, los tiempos se ven muy reducidos, en la configuración de 4 clases cuando utilizamos la distribución uniforme tardamos 3,42 segundos y cuando definimos la inicialización tardamos 1,87 segundos.

Ejercicio C

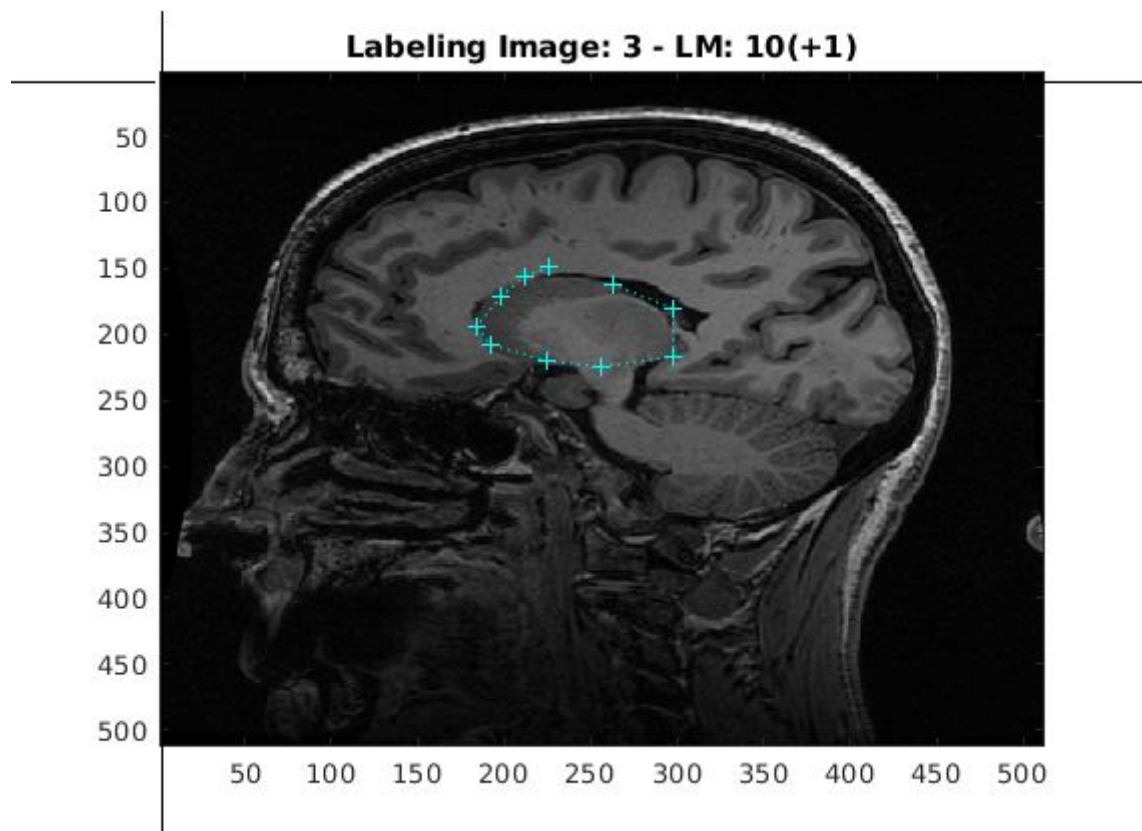
A continuación se muestra en la figura 6, en orden de izquierda a derecha, la imagen original, el gradiente de la imagen realizado con el filtro Prewitt, la imagen sobre segmentada del algoritmo watershed de Matlab y por último la segmentación creando mínimos en ciertas zonas de la imagen.



En este caso no se ha conseguido segmentar de manera adecuada el hígado de la imagen debido a que la imagen gradiente tiene mucho ruido y por tanto, aun creando mínimos en la imagen la segmentación no es satisfactoria.

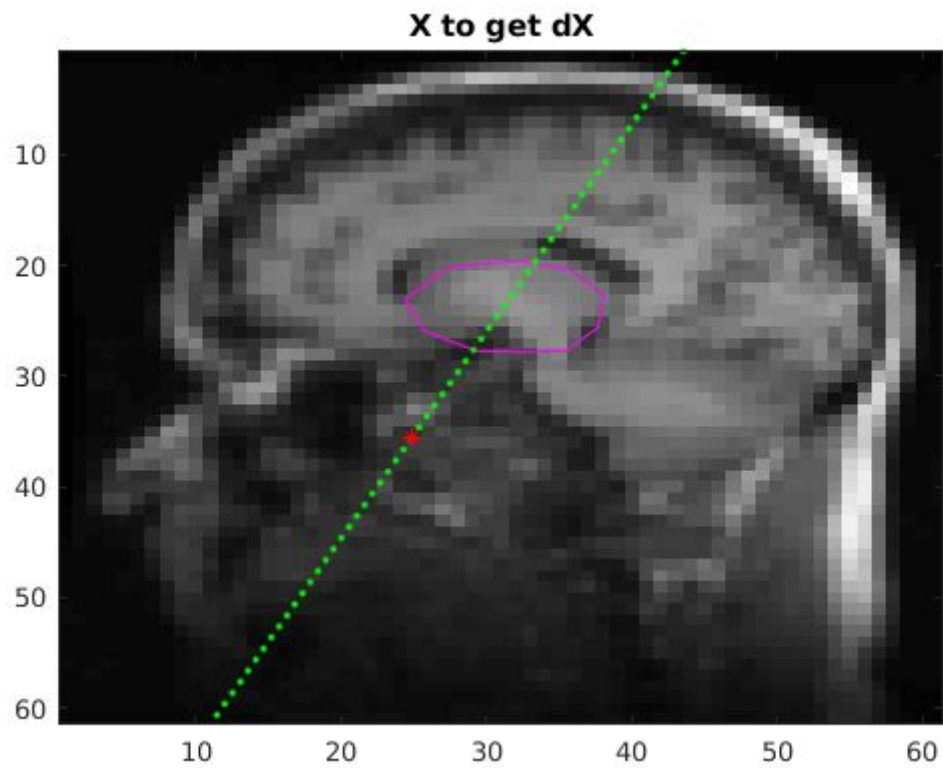
Ejercicio D

Primero seleccionan los puntos de la zona central del cerebro. Figura 10:

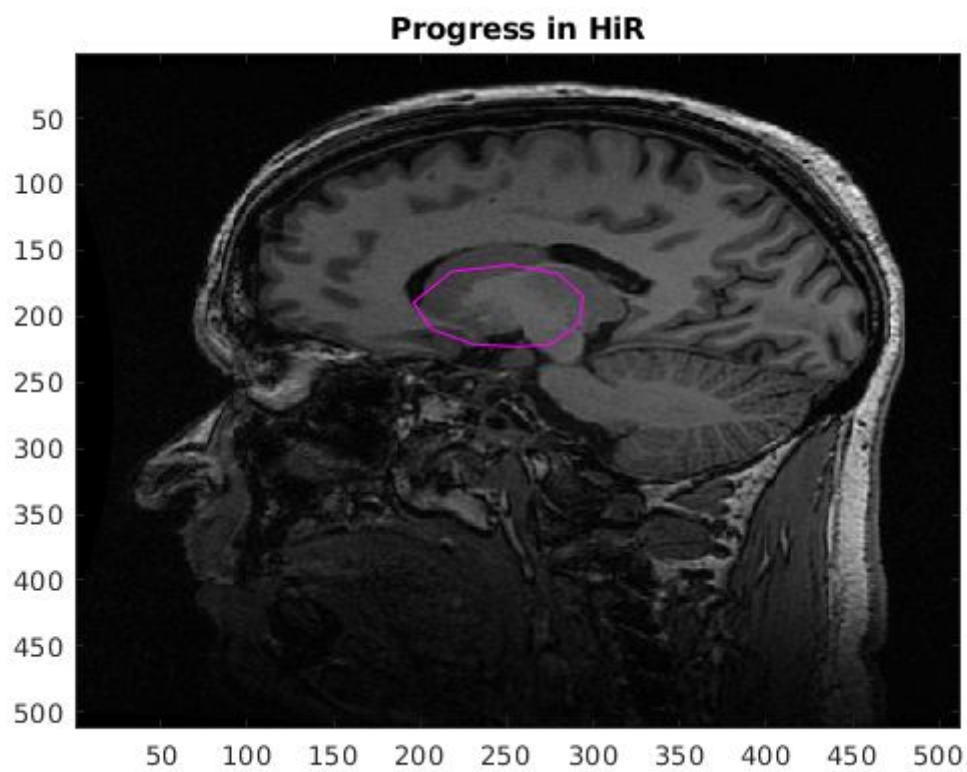


Posteriormente se entrena el modelo y muestras las formas animadas en este [link](#) o el la carpeta de figuras de la memoria (Figura 11).

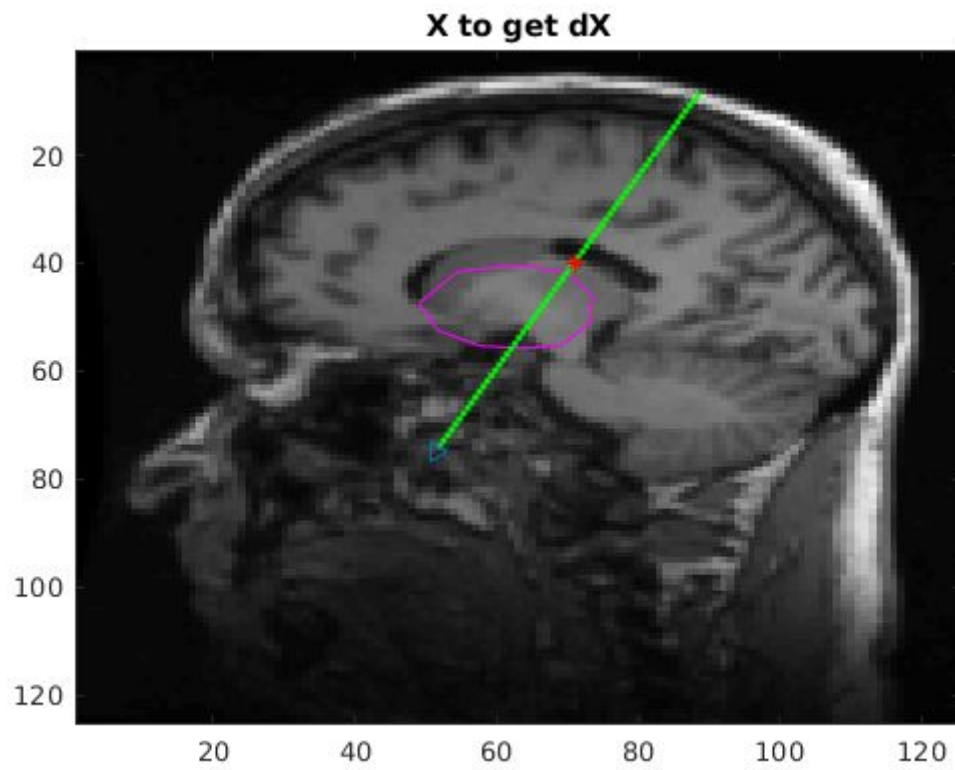
Ahora seleccionamos el centrado del objeto y comienza el algoritmo empezando varias iteraciones con poca resolución. (Figura 12)



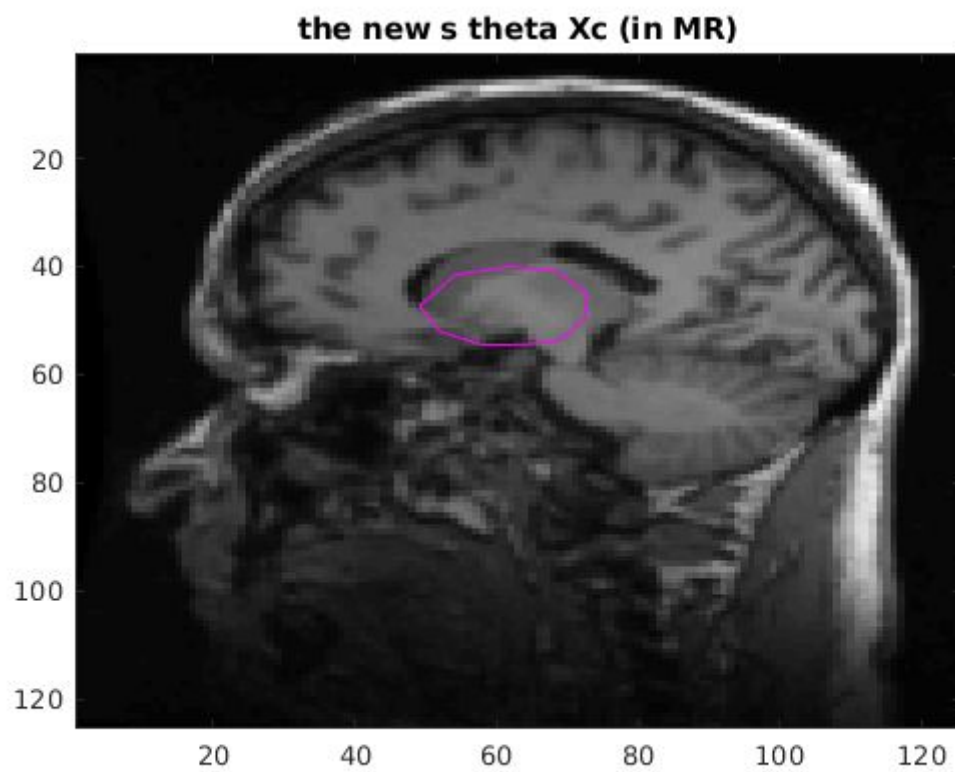
Y posteriormente mostrando un primer resultado. Figura 13



Para continuar con varias iteraciones de mayor resolución. Figura 14

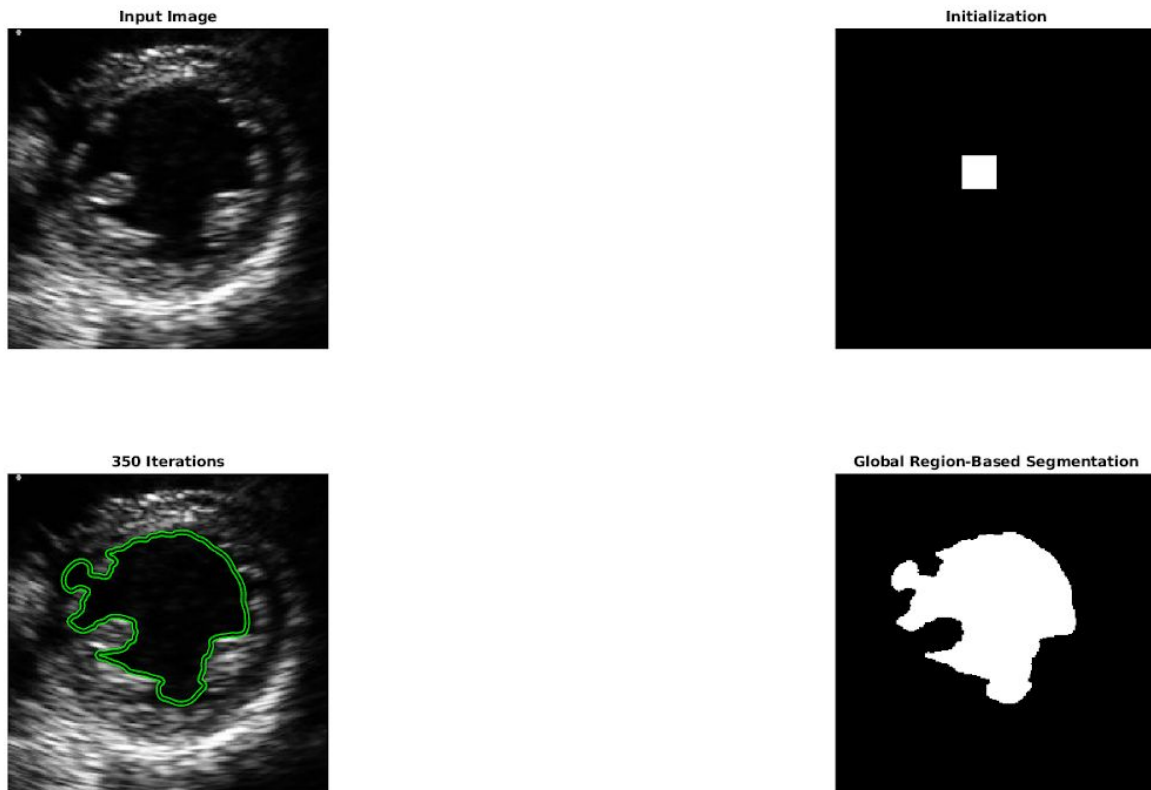


Y por último el resultado final en esta resolución. Figura 15.



Ejercicio E

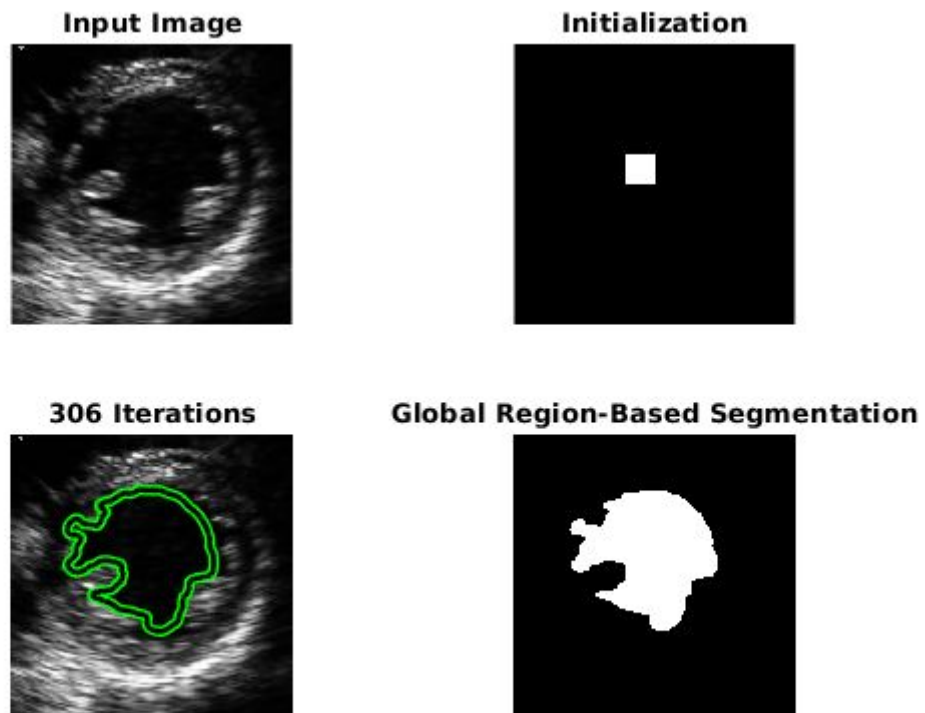
Para mejorar la la primera segmentación obtenida se han reducido el número de iteraciones a 350 y también se ha reducido el suavizado a 0.4 para marcar mejor el contorno deseado. Los resultados son estos (figura 7):



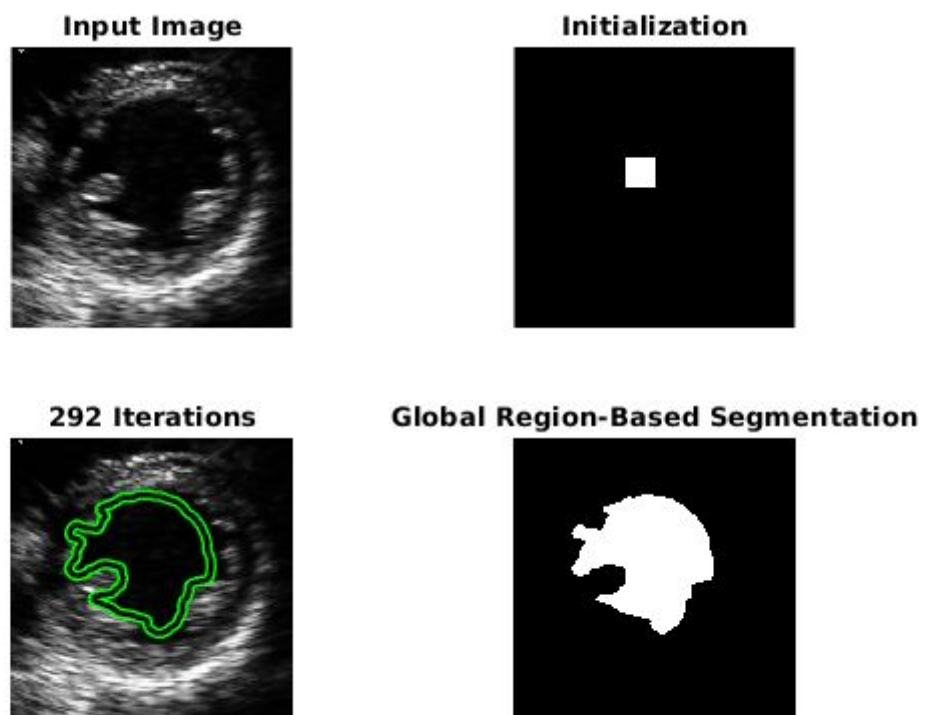
Se han añadido dos nuevos criterios de parada. El primero `max_mean_diff`, el cual establece un parada automática cuando la diferencia de la media de los valores entre la región exterior y la interior del modelo el más alta que el valor definido.

También se ha añadido un punto de parada cuando el número de puntos dentro del contorno interior es mayor al tamaño indicado (`max_interior_points`).

Para la misma iteración anterior poniendo el max_mean_diff el algoritmo para a las 306 iteraciones, figura 8:

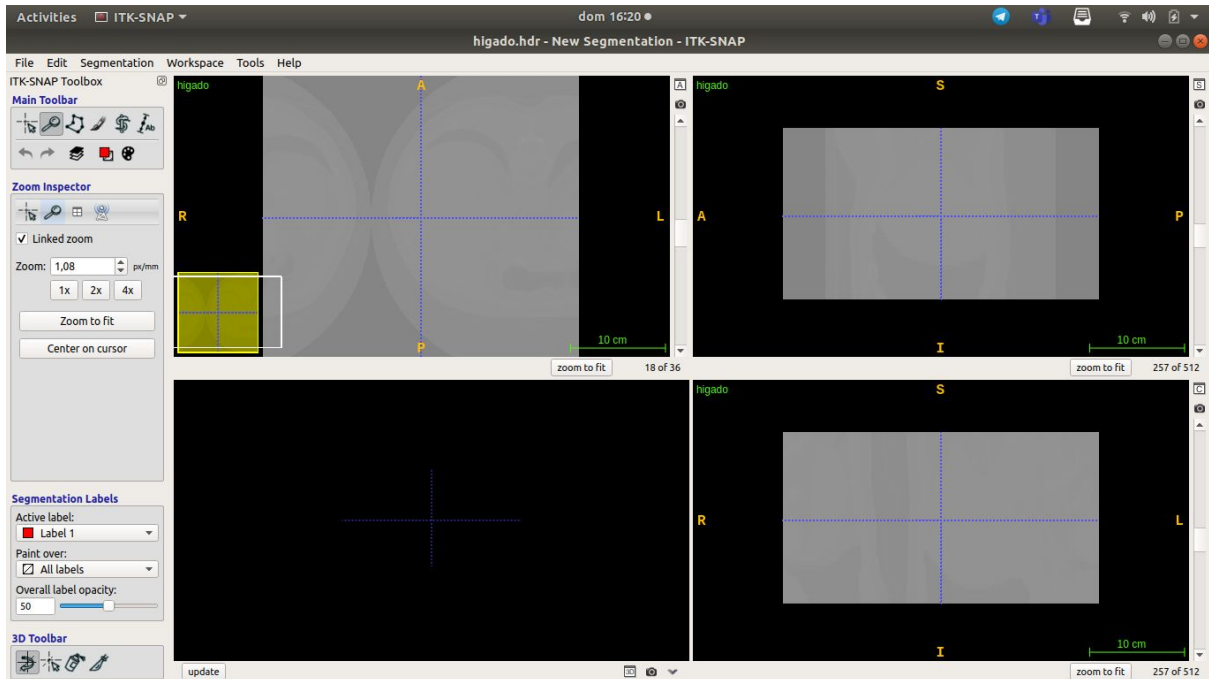


Y si añadimos la parada con 9500 puntos interiores para con 292 iteraciones. (Figura 9):

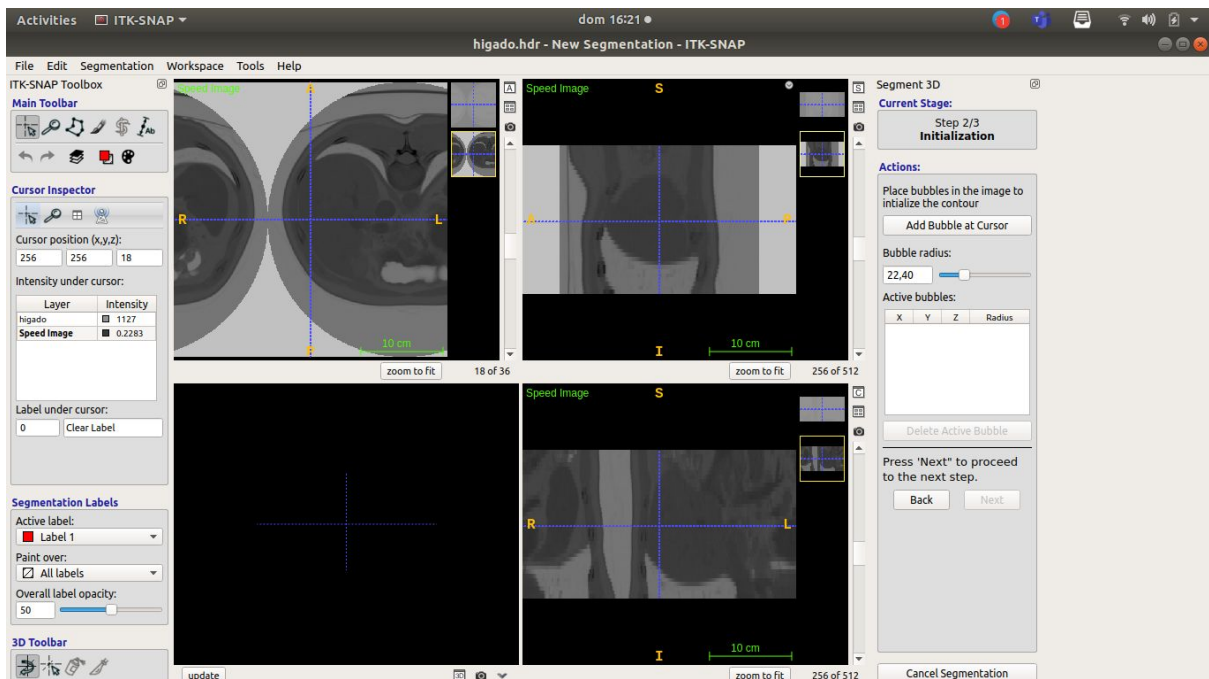


Ejercicio F

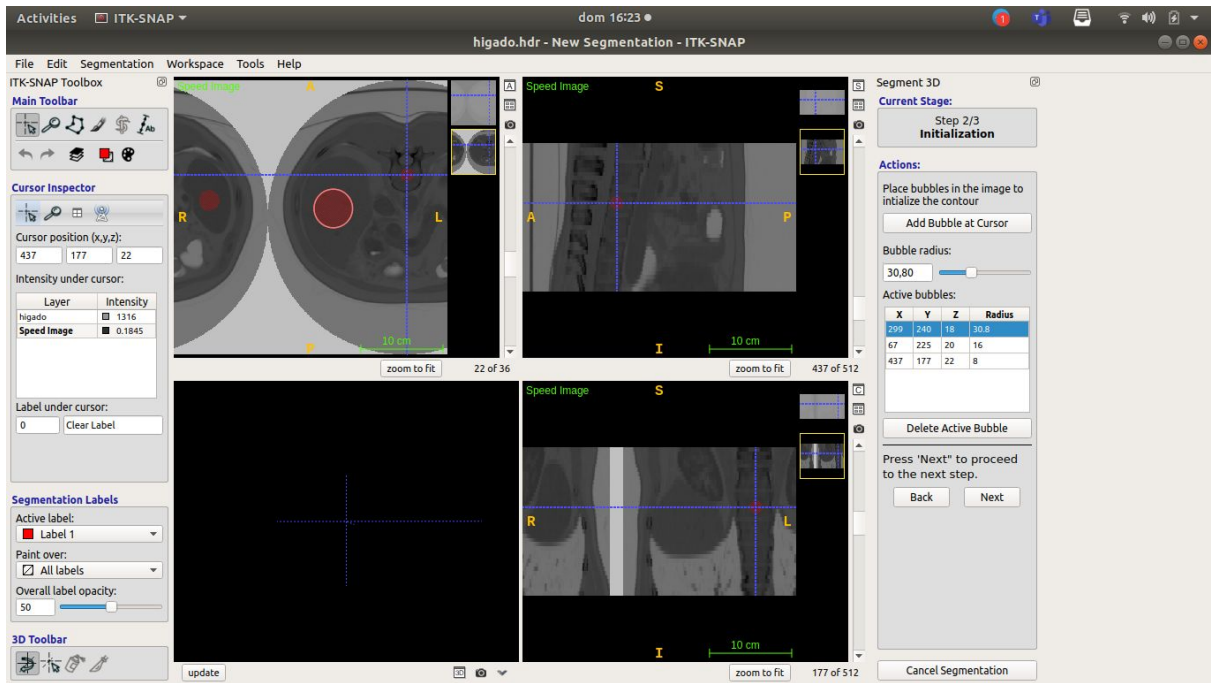
Se inicializa itk-snap y se selecciona el modo de segmentación. EN este caso se realiza con la imagen completa. Figura 16:



A continuación se realiza un threshold de la imagen para conseguir mejor rango dinámico y más diferenciación. Figura 17:



Para continuar, creamos algunas áreas de inicialización, en este caso seleccionamos los dos grupos laterales y columna vertebral. Figura 18:



Por último, se ejecuta el algoritmo y se obtienen los resultados. Tras varios intentos con diferentes configuraciones no se ha conseguido un resultado aceptable. Figura 19:

