



Desarrollo de una herramienta de segmentación de imágenes médicas mediante trazado de polígonos

Juan Manuel Rivera López
jm.rivera@uniandes.edu.co

Asesores:

Marcela Hernández Hoyos
José Miguel Suárez Lopera

Universidad de los Andes
Departamento de Ingeniería de Sistemas y Computación
Bogotá, Colombia

Resumen

La segmentación de imágenes biomédicas es una herramienta valiosa para el diagnóstico de diferentes enfermedades, ya que le permite a un experto definir un área de interés y tomar medidas en esta para desarrollar tratamientos estandarizados. Por esto, se creó una herramienta para realizar segmentación manual por contornos en el visualizador de imágenes médicas Atix. Esta herramienta permitirá realizar anotaciones de imágenes biomédicas y corregir anotaciones realizadas previamente. Se espera que esta herramienta permita incorporar algoritmos de segmentación automática por contornos a Atix como una herramienta adicional a la segmentación por máscaras que se ha desarrollado en el pasado.

Abstract

Segmentation of biomedical images is a valuable tool for the diagnosis of different diseases because an expert can define a region of interest and measure it to develop standardized treatments. Thus, a manual contour segmentation tool was developed into the online medical image viewer Atix. This tool will allow experts to annotate images and adjust previous segmentations. We expect this tool will enable automatic contour algorithms to be included in Atix as a complementary tool to mask segmentation.

Contenido

Resumen	i
Abstract.....	i
Introducción	1
1. Contexto médico.....	4
1.1. Accidentes cerebrovasculares	4
1.2. Circa.....	5
2. Trabajo previo.....	6
2.1. Atix	6
2.2. Seguimiento de tumores cerebrales.....	6
3. Estado del arte	8
3.1. Cornerstone.....	8
3.2. Insight Toolkit (ITK)	9
4. Metodología.....	10
4.1. Definición y trazado de polígonos	10
4.2. Intersección de líneas	11
4.3. Edición de polígonos.....	14
4.4. Área e intensidad de los polígonos	14
5. Arquitectura	16
5.1. Dicom	16
5.2. Dicom-tools	17
5.3. Dicom-thumbnails.....	17
5.4. Dicom-layout	18
5.5. Dicom-plugin-menu	18
5.6. Dicom-sticky-tools	18
5.7. Dicom-serie-loader	19
5.8. Dicom-serie	19
5.9. Dicom-viewer.....	19
5.10. Dicom-painter	19
5.11. Dicom-measurements.....	19
5.12. Dicom-instance-thumbnail	20

6. Resultados.....	21
7. Conclusión y trabajo futuro.....	23
8. Referencias	24

Lista de figuras

<i>Figura 1. Ejemplo de un ROI definida por una máscara (izquierda) y por un polígono (derecha).....</i>	<i>3</i>
<i>Figura 2. Herramienta para comparación de tumores</i>	<i>7</i>
<i>Figura 3. Creación de una segmentación por contornos usando la librería Cornerstone. Imagen tomada de https://tools.cornerstonejs.org/examples/tools/freehand-mouse.html....</i>	<i>8</i>
<i>Figura 4. Creación de una segmentación como polígono (izquierda) o como curva suave (derecha).....</i>	<i>9</i>
<i>Figura 5. Polígono donde no es claro qué áreas son de interés para el usuario.....</i>	<i>10</i>
<i>Figura 6. Visualización de un caso donde la nueva arista (en rojo) se compara con las líneas que se superponen en el eje vertical (en azul).</i>	<i>11</i>
<i>Figura 7. Ilustración de diferentes posibles posiciones para dos segmentos.....</i>	<i>14</i>
<i>Figura 8. Arreglo de pixeles recorridos fila por fila determinando cuáles están dentro del polígono (en naranja).....</i>	<i>15</i>
<i>Figura 9. Diagrama de componentes de Atix. Adaptado del original</i>	<i>16</i>
<i>Figura 10. Renderización de componentes dentro del componente dicom</i>	<i>17</i>
<i>Figura 11. Menú de herramientas del módulo dicom-sticky-tools</i>	<i>18</i>
<i>Figura 12. Detalle del componente measurements.</i>	<i>20</i>
<i>Figura 13. Componente dicom-instance-thumbnail al realizar zoom a una imagen.....</i>	<i>20</i>
<i>Figura 14. Segmentación por contornos realizada dentro de Atix</i>	<i>21</i>
<i>Figura 14. Creación de un contorno</i>	<i>22</i>
<i>Figura 15. Polígono siendo movido.</i>	<i>22</i>

Introducción

Las imágenes médicas son una herramienta ampliamente usada por profesionales de la salud como insumo para emitir diagnósticos y tomar decisiones. Tradicionalmente, el análisis de estas imágenes es realizado por un experto en radiología, el cual busca anomalías y emite un concepto basado en su experticia. Esto implica que el análisis de cada imagen requiere tiempo y puede haber discrepancias entre expertos. Así, cuando el tiempo es una variable decisiva al momento de iniciar un tratamiento estas limitaciones pueden impactar negativamente en el tratamiento.

Una imagen está formada por matrices numéricas, donde cada valor representa una intensidad. Si la matriz es de 2 dimensiones cada posición es un píxel, y si es de 3 dimensiones es un vóxel. Además, varias matrices pueden unirse para codificar una misma imagen, de forma que cada una representa información en un canal o color diferente. Las imágenes biomédicas son almacenadas siguiendo el estándar DICOM (The Medical Imaging Technology Association, 2023), el cual permite almacenar información complementaria junto con las imágenes. Para esto, el archivo DICOM está formado por un conjunto de elementos, donde cada uno cuenta con un identificador estandarizado, la longitud del valor y el valor como tal. De esta forma, algunos de estos elementos se destinan para las intensidades de las imágenes, otros para datos demográficos del paciente y otros para información sobre cómo fue tomada la imagen (como la posición en que se encontraba la persona o el equipo que se usó).

La información que está codificada en una imagen no está estructurada, es decir, las coordenadas de cada píxel dentro de la matriz no están directamente relacionada con su significado. Por esto, las tareas que un humano puede realizar mediante la visión no son sencillas de automatizar en una máquina. Así, para poder desarrollar algoritmos que puedan trabajar con imágenes se ha dividido el problema de la “visión” en problemas más pequeños. Uno de estos es la segmentación, que consiste en separar píxeles que pertenecen a un grupo con características similares de otras áreas.

Los algoritmos tradicionales para segmentación de imágenes se basan en dos propiedades de los grupos: la discontinuidad y la semejanza (González & Woods, 2018). Por un lado, los algoritmos basados en discontinuidad parten de que en el límite entre un grupo y otro hay una diferencia, por lo que es posible identificar un borde que separa cada uno.

Por otra parte, los algoritmos basados en semejanza asumen que los miembros que pertenecen a un mismo grupo tienen características similares entre sí, por lo que si se compara una vecindad de píxeles se podrán agrupar aquellos que pertenecen a un mismo grupo.

También se han usado algoritmos de la familia del aprendizaje supervisado para segmentar imágenes. Estos algoritmos necesitan entrenarse con un conjunto de datos etiquetados para que puedan alcanzar un desempeño óptimo (González & Woods, 2018). En resumen, este entrenamiento consiste en una serie de iteraciones donde el algoritmo predice la etiqueta de cada dato y luego compara su predicción con la etiqueta real. Así, se define una función de error para cuantificar esta diferencia, y se busca minimizarla modificando un conjunto de variables propias de cada algoritmo; esto vuelve el entrenamiento es un problema de optimización. Finalmente, se prueba el rendimiento del algoritmo con un conjunto de datos distinto a los que se usaron durante el entrenamiento, para estimar el rendimiento que tendrá una vez se despliegue. Si bien estos algoritmos permiten identificar patrones dentro de las imágenes y generar mejores contornos, están limitados por la necesidad de tener un conjunto grande de datos etiquetados para poder realizar el entrenamiento, especialmente si se desean usar redes neuronales. Como alternativa, se han propuesto algoritmos que necesitan una menor cantidad de datos etiquetados o que reentrenan modelos (transferencia de aprendizaje), y pueden obtener resultados similares a los que se entrenan desde cero (Trimpl et al., 2022).

Así, las herramientas que permiten realizar anotaciones sobre imágenes son usadas por expertos para construir conjuntos de datos segmentados, que luego pueden ser usados como insumo para entrenar algoritmos de aprendizaje supervisado.

Como alternativa a los métodos de segmentación automática se han propuesto métodos semiautomáticos, los cuáles requieren de una interacción inicial por parte del usuario, tras la cual aplican un algoritmo de segmentación automático en la zona indicada (Trimpl et al., 2022). Así, el usuario tiene más control sobre la segmentación a medida que se crea y requiere de menos datos para obtener un desempeño óptimo.

Sin embargo, tanto las segmentaciones automáticas como las semiautomáticas son susceptibles a errores, por lo que en ambos casos es necesario que el usuario pueda corregir y ajustar los contornos que se creen.

El uso de algoritmos de segmentación en imágenes biomédicas ha sido usado en radiómica, un campo que busca complementar los análisis médicos con el análisis de grandes volúmenes de información extraídos de imágenes (van Timmeren et al., 2020). Por ejemplo, estos análisis son usados para analizar el efecto que un tratamiento tiene en estudios longitudinales. En resumen, estos análisis delimitan una región de interés (ROI, por sus siglas en inglés) la cuál caracterizan mediante diferentes variables con el objetivo de entrenar un algoritmo supervisado. Así, los algoritmos de segmentación disminuyen la variabilidad de las ROI, lo cual da mayor robustez a los análisis que se realizan.

Las ROI pueden almacenarse como una máscara o como un polígono (*Figura 1*). Por un lado, una máscara consiste en una matriz del mismo tamaño de la imagen, que cuando se superpone con la imagen original permite clasificar cada píxel como parte del ROI o como parte del fondo. Por otro lado, los polígonos son un conjunto de puntos y líneas que delimitan una región, y aquellos pixeles cuyo centro estén dentro del polígono serán considerados como parte del ROI. Así, las ROI pueden codificarse como una matriz binaria o como un conjunto de puntos (x,y,z) que delimitan un polígono o un poliedro (Zwanenburg et al., 2016). Ambas representaciones pueden ser almacenadas en formato DICOM junto con la imagen original.

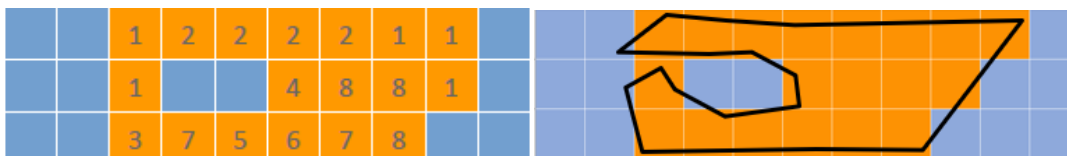


Figura 1. Ejemplo de un ROI definida por una máscara (izquierda) y por un polígono (derecha)

Teniendo esto en cuenta, se desarrolló una herramienta para trazar y editar segmentaciones por contornos dentro del visualizador web de imágenes médicas Atix. Se espera que esta herramienta permita a profesionales de la salud obtener datos cuantitativos de las imágenes biomédicas, realizar etiquetado de ROIs en diferentes imágenes y ajustar segmentaciones que hayan sido creadas previamente.

1. Contexto médico

1.1. Accidentes cerebrovasculares

Un accidente cerebrovascular (ACV) se define como un episodio agudo de disfunción focalizada del cerebro, la retina o la médula espinal, el cual da lugar a una serie de síntomas asociados al área que se haya afectado (Hankey, 2017). La etiología de esta disfunción está asociada con un cambio la perfusión sanguínea del área afectada, de forma que puede tratarse de una hemorragia intracerebral o subaracnoidea (accidente cerebrovascular hemorrágico), o del bloqueo total de una arteria debido a un trombo (accidente cerebrovascular isquémico). Parte del diagnóstico incluye realizar una tomografía axial computarizada (TAC) sin medio de contraste o una resonancia magnética de difusión, para determinar de qué tipo se trata.

El tratamiento de los ACV isquémicos consisten en aplicar medicamentos que disuelvan el coágulo, como la alteplasa, o realizar una trombectomía endovascular, la cual consiste en retirar el coágulo con un dispositivo mecánico. Sin embargo, sólo el 10% de los pacientes que sufren un ACV son candidatos para que se les realice una trombectomía (Hankey, 2017). Para determinar si un paciente es candidato o no, se analiza la extensión del área afectada por el ACV en una tomografía axial computarizada. Así, si la arteria ocluida es la cerebral media, se utiliza el puntaje ASPECTS para decidir si el paciente es candidato para una trombectomía (Lassalle et al., 2016).

El puntaje ASPECTS delimita 10 áreas de interés en cada hemisferio: 7 áreas corticales y 3 áreas en los ganglios de la base (Pexman et al., 2001). Luego, realiza una comparación de la intensidad de cada área con su contraparte contralateral y, si hay una diferencia significativa entre ambas áreas, resta 1 al puntaje. Así, un paciente sano tendrá un ASPECTS de 10 mientras que uno con un ACV isquémico tendrá un puntaje menor.

Por otro lado, el tratamiento de los ACV hemorrágicos depende de la etiología del sangrado. Algunos de los posibles tratamientos incluyen disminución de la presión arterial, administrar factores de coagulación o plaquetas, o realizar cirugías que permitan drenar la hemorragia (Hankey, 2017). Una de las escalas propuestas para evaluar el prospecto de un ACV hemorrágico es el puntaje ICH, el cuál evalúa el puntaje en la Escala de Coma de Glasgow, si el paciente es mayor de 80 años, si el origen de la hemorragia es infratentorial, si hay hemorragia intraventricular y si el volumen de la hemorragia es mayor a 30 cm³.

1.2. Circa

El proyecto Circa busca crear una aplicación que permita agilizar la atención médica a pacientes que hayan sufrido de un accidente cerebrovascular (ACV). Para esto, ofrecerá diferentes herramientas a los profesionales de la salud para apoyar el tratamiento desde el momento en que se atiende una emergencia hasta que el paciente es atendido en un centro de salud.

Una de las herramientas que se ofrecerá será el visualizador web de imágenes médicas Atix (Vicerrectoría de Investigación, 2020), con el objetivo de que las imágenes diagnósticas que se tomen en un centro de salud estén disponibles si el paciente debe ser trasladado. Además, se añadirá la funcionalidad de calcular el puntaje ASPECTS (Pexman et al., 2001) para los accidentes cerebrovasculares isquémicos. Este puntaje permite estimar la extensión del telencéfalo que fue afectada por el ACV a partir de tomografías axiales computarizadas, como se mencionó anteriormente. Para esto, se generarán segmentaciones de las regiones de interés en cada hemisferio mediante un plugin, y se permitirá la edición de estos contornos mediante el trabajo desarrollado en este documento.

2. Trabajo previo

2.1. Atix

Atix es un visualizador web de imágenes diagnósticas que se desarrolló como una alternativa para no depender de CDs ni de descargar software especializado al momento de visualizar estas imágenes (Vicerrectoría de Investigación, 2020). Este visualizador permite la conexión a un servidor donde se almacenan las imágenes en formato DICOM, y da al usuario herramientas para realizar mediciones sencillas sobre las imágenes. Además, se han desarrollado plugins para poder realizar modificaciones complejas, tales como aplicar algoritmos de segmentación por máscaras, de forma que puedan visualizarse y editarse en Atix (Camacho Daza et al., 2021; Rodríguez Castiblanco et al., 2022).

2.2. Seguimiento de tumores cerebrales

Un uso potencial para la segmentación por contornos de imágenes médicas es el seguimiento a tumores cerebrales a través del tiempo. Para esto, se debe cuantificar el área del tumor y determinar cómo esta ha variado en diferentes momentos. Así, se desarrolló una aplicación web (Cano & Hernández Hoyos, 2021; Meneses Castillo et al., 2020) que permite comparar dos imágenes en paralelo y trazar un contorno simultáneamente en ambas (*Figura 2*). De esta forma, se puede comparar el tamaño de un tumor delimitado por una ROI en dos momentos y caracterizarlo en términos de su intensidad y su área. Además, permite ver una distribución de las intensidades presentes mediante un histograma.

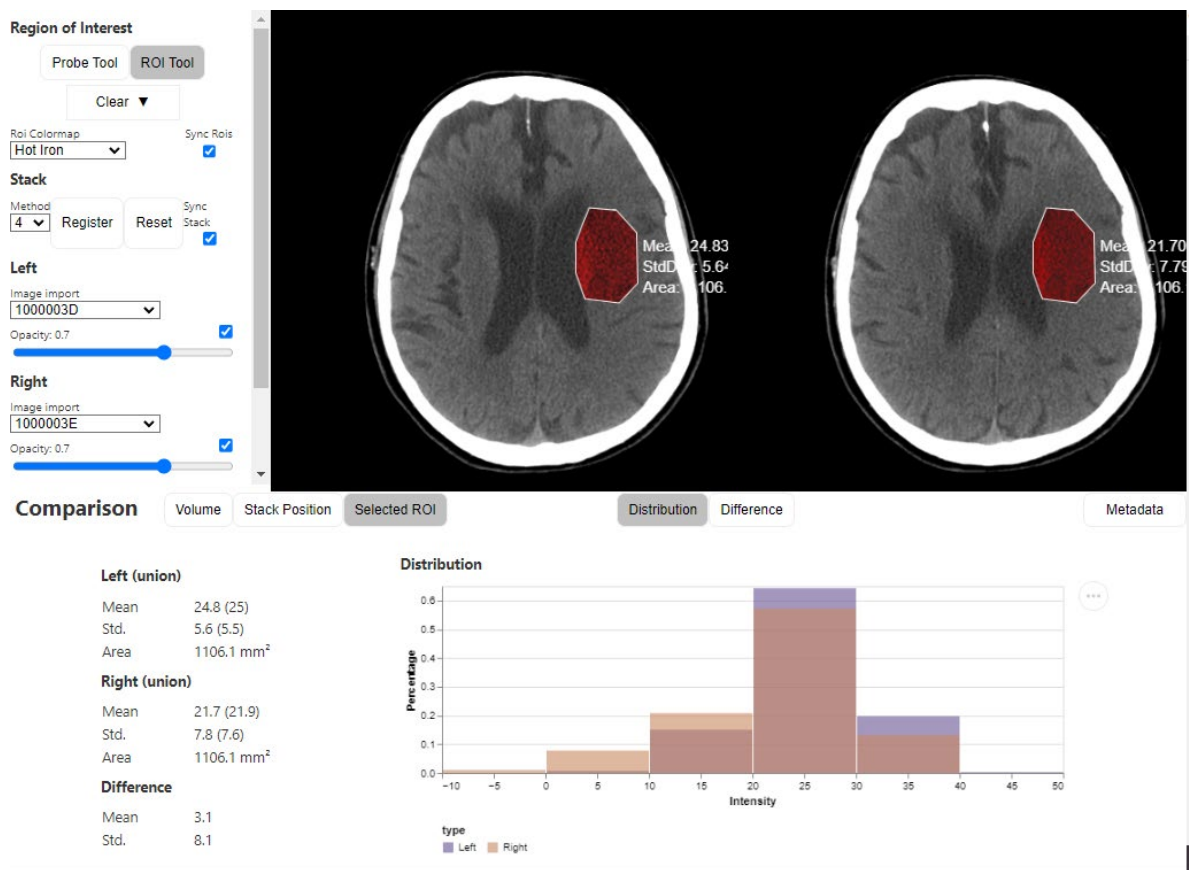


Figura 2. Herramienta para comparación de tumores

3. Estado del arte

Con el objetivo de identificar buenas prácticas al momento de desarrollar una herramienta para realizar segmentación por contornos, se revisó cómo otros programas ofrecen esta funcionalidad. Para esto, se analizó cómo es el proceso para trazar y modificar contornos, si se tiene acceso a las características del área delimitada y si es posible exportar los contornos una vez han sido creados.

3.1. Cornerstone

La Open Health Imaging Foundation creó una librería para visualizar imágenes médicas en navegadores web conocida como Cornerstone (Open Health Imaging Foundation, 2023). Esta librería permite crear visualizadores de imágenes en formato DICOM en un navegador web, los cuáles son compatibles con la librería de herramientas Cornerstone-tools. Una de estas herramientas es FreehandRoi, una herramienta que permite trazar contornos mediante puntos sobre una imagen. Si durante la creación, al momento de añadir un nuevo punto dos o más líneas se cruzan, la herramienta se lo indica al usuario cambiando el color del borde del polígono (*Figura 3*). Así, una vez creado un polígono se pueden mover los puntos que lo conforman, pero para eliminarlo o exportarlo se debe interactuar con el `imageIdSpecificToolStateManager`. Adicionalmente, se puede obtener el área y la intensidad de la región a través de los métodos de la librería de herramientas. Una de las ventajas de esta librería es que si se tiene un visualizador basado en Cornerstone, se tiene acceso a una amplia gama de herramienta con la que es compatible, pero se debe tener en cuenta que estas herramientas solo funcionarán con este visualizador.

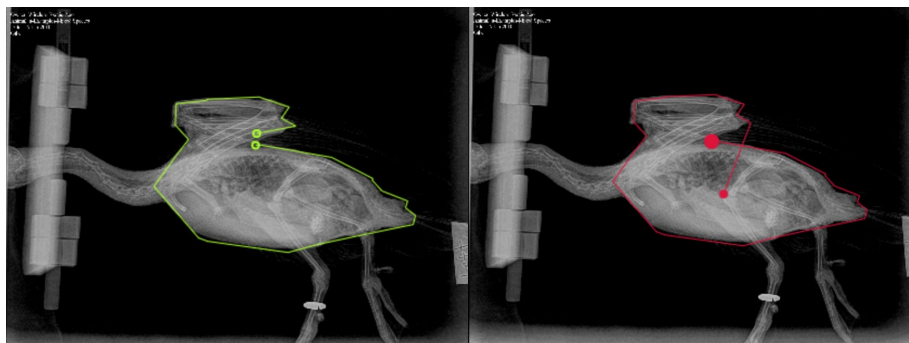


Figura 3. Creación de una segmentación por contornos usando la librería Cornerstone. Imagen tomada de <https://tools.cornerstonejs.org/examples/tools/freehand-mouse.html>

3.2. Insight Toolkit (ITK)

El Insight Toolkit (ITK) es una librería de código abierto que permite usar herramientas para análisis de imágenes en formato DICOM (Yushkevich et al., 2006). Una de las herramientas ofrecidas es el trazado de polígonos (Polygon Mode), donde se puede crear una segmentación ya sea dibujando puntos con cada click o dibujando una curva manteniendo el mouse oprimido (*Figura 4*).

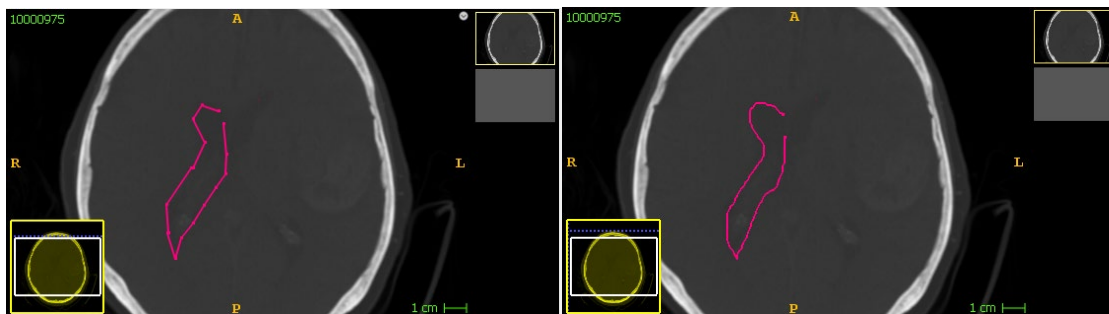


Figura 4. Creación de una segmentación como polígono (izquierda) o como curva suave (derecha)

Se pueden elegir diferentes etiquetas para anotar el área que encierra el polígono, y durante la creación se puede cerrar automáticamente el contorno, eliminar el último punto que se añadió o eliminar el polígono que se estaba trazando. Una vez trazado un contorno se asigna el valor de la etiqueta como una máscara a los píxeles que estén dentro del contorno, por lo que no es posible editar la posición de los puntos una vez creado. Además, el programa no ofrece métodos para cuantificar el área dentro, pero permite exportar las ediciones en formato GIPL, analyze, Metalmage, NiFTI, NRRD, VoxBo CUB, VTK image o el formato genérico de ITK.. Esto puede deberse a que la funcionalidad de polígonos de ITK-SNAP está pensada para realizar etiquetado de imágenes, mas no análisis de estas.

A partir de estos ejemplos se puede ver que la edición de los contornos, una vez creados, es más sencilla si se realiza editando los puntos del polígono que modificando el área de una máscara. Además, la opción de eliminar el último punto a medida que se crea el contorno es útil. Por otra parte, visualizar la línea que se forma entre el punto previo y el nuevo punto es útil si se quiere evitar un corte entre aristas, pero si la línea es muy gruesa puede entorpecer el trazado del contorno. Finalmente, se observa que proveer instrucciones claras para crear, editar y eliminar los polígonos es clave para garantizar una mejor experiencia de usuario.

4. Metodología

Para implementar el algoritmo de trazado de polígonos, primero se definieron las restricciones que debía cumplir un polígono. Posteriormente, se ajustaron algoritmos y métodos que permiten revisar y cumplir estas restricciones. Luego, se implementaron las funciones de edición de los polígonos. Finalmente, se creó un método para calcular el área y promediar la intensidad encerrada por el polígono.

4.1. Definición y trazado de polígonos

Cada polígono está compuesto por un conjunto ordenado de puntos P , donde cada punto tiene una coordenada (x,y) . Así, las aristas son las líneas que unen dos puntos consecutivos del conjunto y la línea que une el primer y el último punto. Se determinó que un polígono debe cumplir con 3 características: (1) debe tener mínimo 3 puntos, (2) ningún par de puntos pueden tener las mismas coordenadas y (3) cada arista puede cruzarse únicamente con las dos aristas con las que comparte un extremo, y sólo pueden cruzarse en un punto.

Las restricciones (1) y (2) evitan que un polígono tenga área cero, mientras que la restricción (3) evita que se presenten escenarios donde hay ambigüedad en cuanto a qué área debería considerarse como parte del interior del contorno (*Figura 5*).

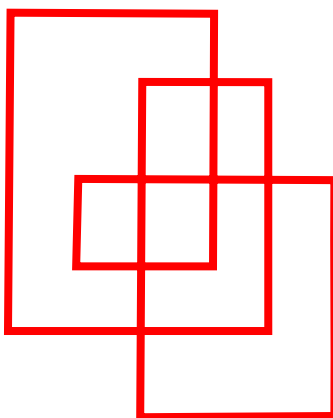


Figura 5. Polígono donde no es claro qué áreas son de interés para el usuario

Para validar la condición (3) es necesario identificar los puntos donde dos aristas se cruzan, lo cual se mostrará en detalle en la siguiente sección. Sin embargo, mientras se construye el polígono esta comparación puede simplificarse y evaluar únicamente los

puntos donde la nueva arista se cruza con las aristas ya existentes, ya que los puntos previos deben haber cumplido con la condición (3).

Se realizó una adaptación de un algoritmo optimizado para buscar intersecciones en un conjunto de segmentos rectos (de Berg et al., 2008). En resumen, este algoritmo ordena los puntos en función de su posición y los recorre de arriba a abajo, y aquellos que estén alineados horizontalmente los recorre de izquierda a derecha. A medida que revisa cada punto, analiza si hay intersecciones entre las aristas que estén a la misma altura horizontalmente. Esto permite reducir el número de comparaciones entre aristas, pues solo se comparan las que se superponen en el eje horizontal (*Figura 6*). Adicionalmente, guarda las aristas en un árbol binario, de forma que cada arista la compara únicamente con aquellas que sean sus vecinas a izquierda y derecha, reduciendo el número de comparaciones a las que son vecinas en el espacio.

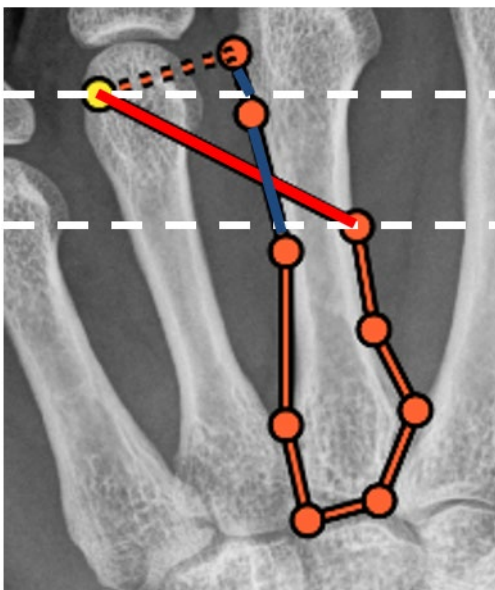


Figura 6. Visualización de un caso donde la nueva arista (en rojo) se compara con las líneas que se superponen en el eje vertical (en azul).

4.2. Intersección de líneas

Para determinar si dos aristas de un polígono se cruzan, se busca el punto donde ambas líneas tendrían el mismo valor si tuvieran una longitud infinita (Rajendra Chaudhari, 2020). Para esto, cada línea se definió como:

$$B * y - A * x = C \quad (1)$$

Donde A, B y C son constantes, que pueden calcularse a partir de las coordenadas de dos puntos dados:

$$A = inicio.Y - final.Y \quad (2)$$

$$B = inicio.X - final.X \quad (3)$$

$$C = B * inicio.Y - A * inicio.X \quad (4)$$

Luego, calcula la intersección entre las dos líneas de la siguiente forma:

$$intersección.X = \frac{B_1 C_2 - B_2 C_1}{B_2 A_1 - B_1 A_2} \quad (5)$$

$$intersección.Y = \frac{A_1 C_2 - A_2 C_1}{B_2 A_1 - B_1 A_2} \quad (6)$$

Donde los subíndices indican si pertenece a la línea 1 o a la línea 2. Nótese que si dos líneas tienen el mismo valor de C ($C_1 = C_2$) o si son paralelas, el denominador será 0 en las ecuaciones (5) y (6), por lo que estos casos se analizan de forma diferente.

En caso de que $C_1 \neq C_2$ y el denominador no sea 0, se analiza si el punto donde se intersecan ambas líneas está en el rango de cada segmento

$$inicio_1.Y > intersección.Y > final_1.Y \wedge inicio_2.Y > intersección.Y > final_2.Y \quad (7)$$

$$inicio_1.X > intersección.X > final_1.X \wedge inicio_2.X > intersección.X > final_2.X \quad (8)$$

Si se cumplen (7) y (8), la intersección forma parte de ambos segmentos, y se incumplirían las restricciones antes descritas. Si no, el punto donde se intersecan no haría parte de los segmentos (*Figura 7.A*)

Por otra parte, dos líneas pueden tocarse en los segmentos es un extremo de ambos, caso donde se cumplirían con las restricciones definidas en la sección anterior.

$$[(inicio_1.Y = intersección.Y) \vee (final_1.Y = intersección.Y)] \wedge [(inicio_2.Y = intersección.Y) \vee (final_2.Y = intersección.Y)] \quad (9)$$

$$[(inicio_1.X = intersección.X) \vee (final_1.X = intersección.X)] \wedge [(inicio_2.X = intersección.X) \vee (final_2.X = intersección.X)] \quad (10)$$

Si se cumplen (9) y (10), el punto donde las dos líneas se intersecan es uno de los extremos de ambas.

En caso de que $C_1 \neq C_2$ y el denominador de (5) y (6) sea 0, las líneas son paralelas. En este caso no se intersecan en ningún punto y no se incumple ninguna restricción (*Figura 7.D*).

En caso de que $C_1 = C_2$ ambos segmentos son parte de la misma línea. En este caso se debe analizar si los segmentos coinciden en un punto, en infinitos puntos o en ningún punto (*Figura 7.B y Figura 7.C*). Para esto, basta con analizar la proyección de las líneas en uno de los ejes, por lo que se vuelve un problema de hallar la intersección de dos segmentos en 1 dimensión, y se puede resolver mediante la siguiente expresión (Benderersky, 2008):

$$(final_1.X \geq inicio_2.X) \wedge (final_2.X \geq inicio_1.X) \quad (11)$$

Así, cuando la expresión (11) es verdadera, los dos segmentos se intersecan en uno o más puntos. El caso donde solo se unen en uno de sus extremos es similar al mostrado en las ecuaciones (9) y (10), pero en una sola dimensión

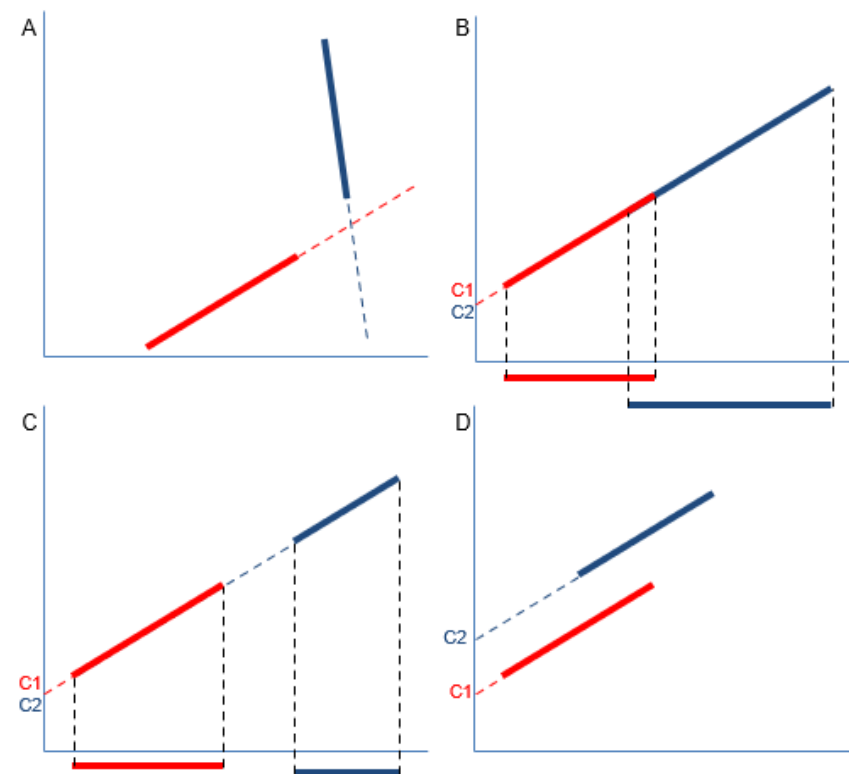


Figura 7. Ilustración de diferentes posibles posiciones para dos segmentos

4.3. Edición de polígonos

Es importante que la herramienta permita la edición de un polígono existente, para que el usuario pueda ajustar segmentaciones que se han realizado previamente (por ejemplo, mediante segmentación automática). Así, se incluyó una función para mover cada uno de los puntos que conforma el polígono por separado y para mover todos los puntos al mismo tiempo. De acuerdo con las restricciones que se plantearon, al mover cada uno de los puntos existentes se comprueba que ninguna arista se cruce con otra.

Por otra parte, para mover todos los puntos al tiempo se guarda un punto inicial, a modo de punto de referencia. Luego, cuando el usuario selecciona el punto de destino se calcula la diferencia entre el punto inicial y el punto final, y se añade esta diferencia a cada punto del polígono.

4.4. Área e intensidad de los polígonos

Es de interés conocer el área encerrada por cada polígono y el histograma de intensidades de los píxeles que lo conforman, ya que esta información permite caracterizar una región y compararla objetivamente con otras.

Para calcular ambas medidas, primero se encuentra el rectángulo de menor tamaño que contenga todos los puntos del polígono. Así, se limita el área donde se iterará y se disminuye el costo computacional del cálculo. Luego, se recorre cada fila de este rectángulo, determinando cuales píxeles se encuentran dentro y cuales fuera (*Figura 8*). Un píxel está dentro del polígono si cualquier línea recta que se origine desde este píxel atraviesa un número impar de veces las aristas del polígono. De esta forma, al momento de implementarlo basta con tener en cuenta una sola línea, la cual va desde el centro del píxel hacia la derecha hasta el borde del rectángulo. En caso de que un píxel se encuentre dentro del polígono se añadirá su intensidad y su área a un arreglo. Una vez se han recorrido todos los píxeles se calcula el área y se analiza la distribución de intensidades mediante un histograma.

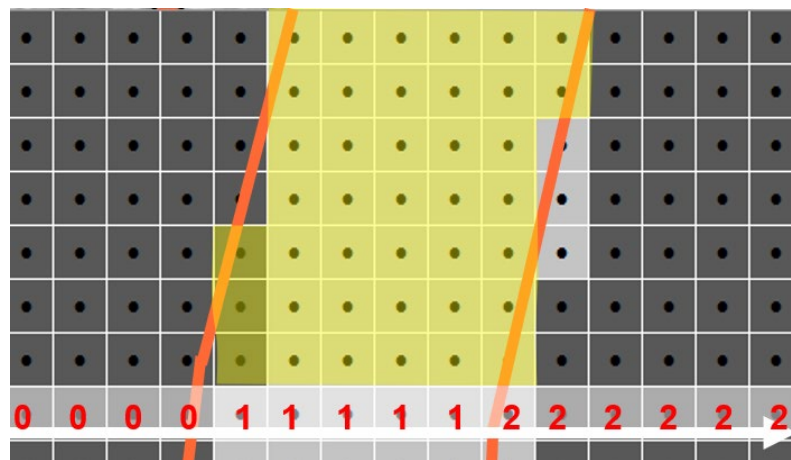


Figura 8. Arreglo de píxeles recorridos fila por fila determinando cuáles están dentro del polígono (en naranja)

5.Arquitectura

Dado que esta herramienta se construye sobre el visualizador web Atix (Vicerrectoría de Investigación, 2020), se respeta la arquitectura plateada por esta, y se incorpora la herramienta a los módulos que existían previamente (*Figura 9*). Los módulos sobre los cuáles se realizaron cambios se muestran marcados en verde

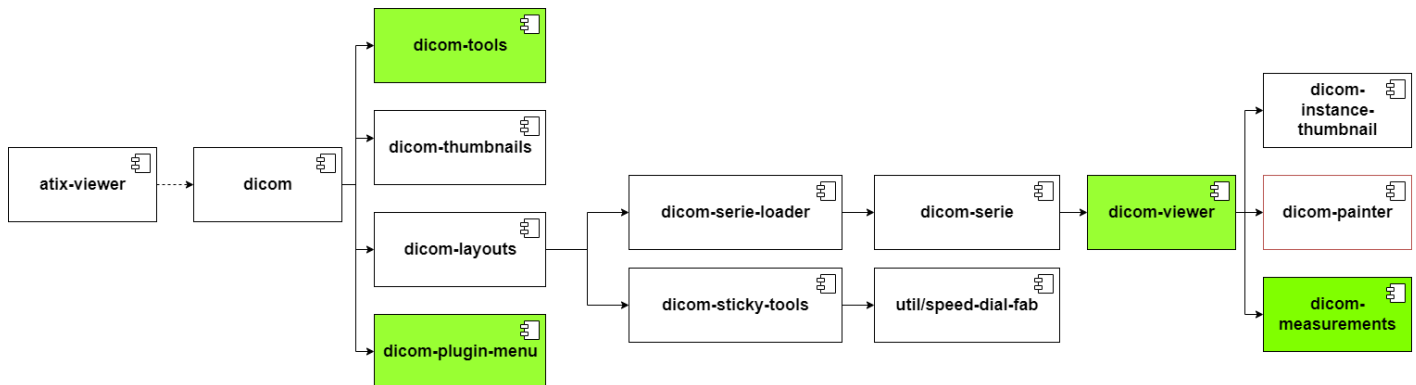


Figura 9. Diagrama de componentes de Atix. Adaptado del [original](#)

A continuación, se mostrará la función general de todos los módulos mostrados. Para una revisión más en profundidad se recomienda el manual del usuario (Vicerrectoría de Investigación, 2020).

5.1. Dicom

Este módulo tiene como selector 'atix-viewer', y es llamado desde el componente principal de Atix. Este componente se organiza la posición donde los demás componentes se renderizarán siguiendo el layout definido en el componente dicom-layout (*Figura 10*)

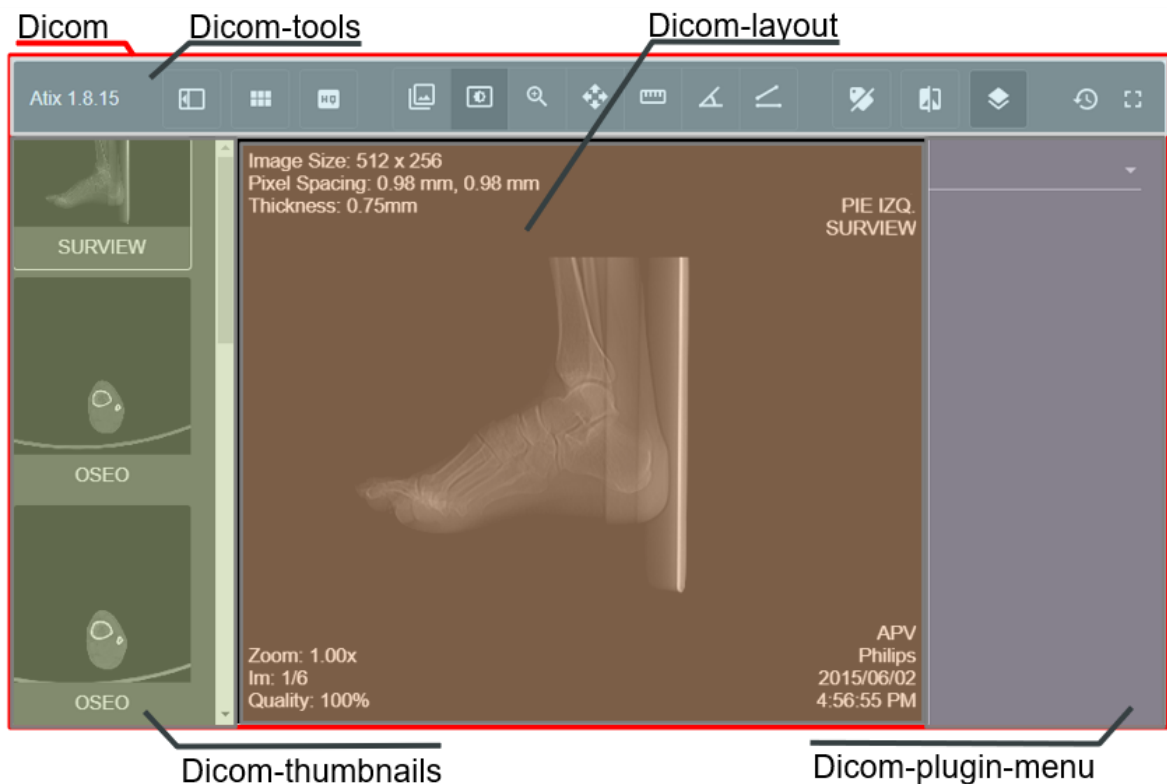


Figura 10. Renderización de componentes dentro del componente dicom

5.2. Dicom-tools

Este módulo renderiza la barra de herramientas que se muestra en la parte superior del visualizador. Cuando el usuario interactúa con alguna herramienta el evento es enviado al módulo dicom-state, el cual hace el cambio público para otros módulos. Algunas de las herramientas permiten modificar la disposición, mostrando u ocultando menús, mientras que otras permiten realizar mediciones sobre las imágenes.

5.3. Dicom-thumbnails

Este módulo recibe los metadatos de un estudio y muestra una miniatura para cada una de las series que haya dentro del estudio. Cuando el usuario selecciona una de las miniaturas, el módulo envía un evento al módulo dicom-state indicando que se seleccionó una nueva serie, para que pueda ser renderizado en el módulo correspondiente. Además, la lista de miniaturas puede mostrarse u ocultarse usando una de las herramientas del componente dicom-tools.

5.4. Dicom-layout

Este módulo funciona como un marco para mostrar una o varias series en pantalla al tiempo. Así, este módulo distribuye la posición de diferentes layouts, en función de lo que el usuario desee. Además, para realizar la carga de series el módulo renderiza en su interior el módulo dicom-serie-loader. En caso de que haya varias series, carga un módulo dicom-serie-loader por cada una. Por otra parte, también renderiza el módulo dicom-sticky-tools, el cual se mostrará más adelante.

5.5. Dicom-plugin-menu

Este módulo muestra un menú interactivo donde se puede solicitar que se realice una segmentación con alguno de los plugins incorporados en Atix. Cuando se carga una segmentación, las herramientas para editarla se muestran en esta zona. Además, puede mostrarse u ocultarse con una de las herramientas del módulo dicom-tools.

5.6. Dicom-sticky-tools

Este módulo renderiza el módulo speed-dial-fab, el cuál se localiza dentro del módulo util. Este módulo permite acceder al menú de herramientas mediante un menú localizado dentro del módulo dicom-layout (*Figura 11*), y se usa como alternativa a la barra de herramientas en tablets.

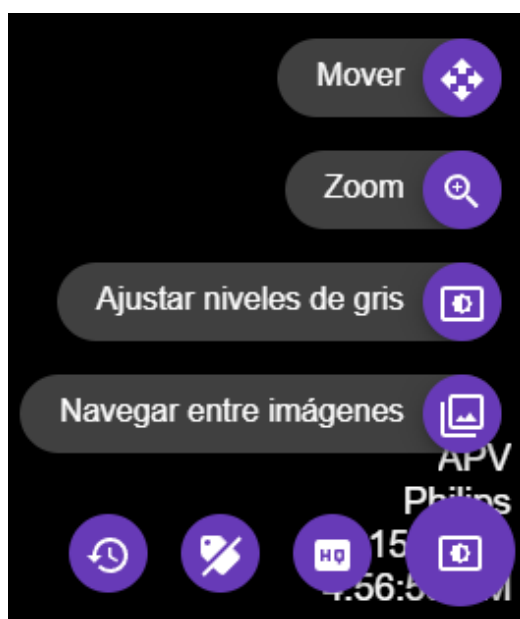


Figura 11. Menú de herramientas del módulo dicom-sticky-tools

5.7. Dicom-serie-loader

Este módulo se encarga de descargar la información de los metadatos de cada imagen/instancia de una serie, de forma que al momento de renderizarla ya está lista. Además, realiza una revisión en caso de que la serie no tenga ninguna instancia.

5.8. Dicom-serie

Este módulo carga los metadatos que fueron previamente descargados, se conecta al dicom-state y luego le pasa las variables necesarias al módulo dicom-viewer, para que renderice la imagen y los metadatos.

5.9. Dicom-viewer

Este módulo muestra como texto al usuario los metadatos del estudio y las coordenadas donde está el puntero. Además, cuenta con dos canvases en su interior, uno para las instancias (la imagen médica) y otro para los datos de una máscara. Así, tiene los métodos para mostrar una imagen a partir de los pixeles del archivo DICOM, para ajustar la imagen y para procesar los clicks cuando hay una herramienta seleccionada.

En su interior se localiza el módulo dicom-instance-thumbnail, el módulo dicom-painter y el módulo dicom-measurements

5.10. Dicom-painter

Este módulo despliega un canvas dedicado a la herramienta pincel, la cual permite editar segmentaciones hechas mediante máscaras y delimitar áreas manualmente.

5.11. Dicom-measurements

Este módulo controla las medidas que se dibujan sobre una imagen. Así, hay medidas sencillas que pueden ser de tipo punto o línea, y medidas complejas que se forman uniendo las sencillas. De este modo, un ángulo está formado por tres puntos, un ángulo de Cobb (TwoLines) está formado por dos líneas y un polígono está formado por una colección de puntos (*Figura 12*).

Así, cada instancia tiene asociado un arreglo de medidas, de los tipos antes mencionados, las cuáles son renderizadas en un canvas independiente.

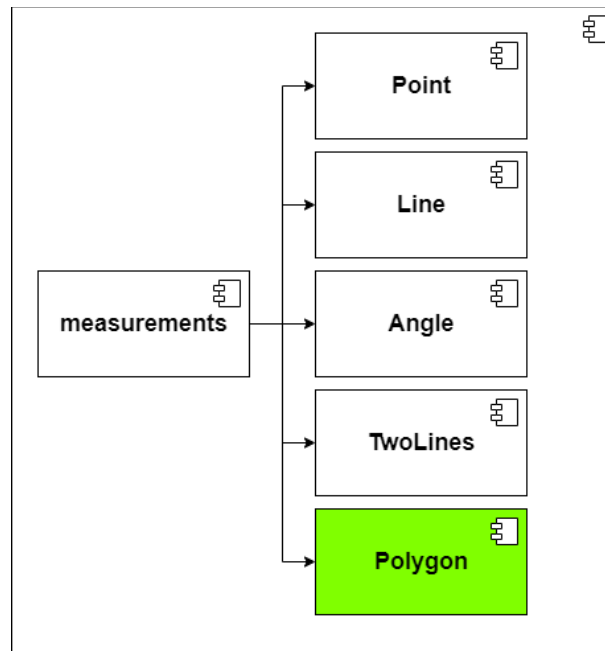


Figura 12. Detalle del componente measurements.

5.12. Dicom-instance-thumbnail

Este módulo sólo se muestra cuando un usuario realiza un zoom en la imagen. Si se pasa un umbral, se muestra una miniatura de la instancia a la que se le está haciendo zoom junto con un cuadrado naranja que muestra el área que está ampliada (Figura 13).

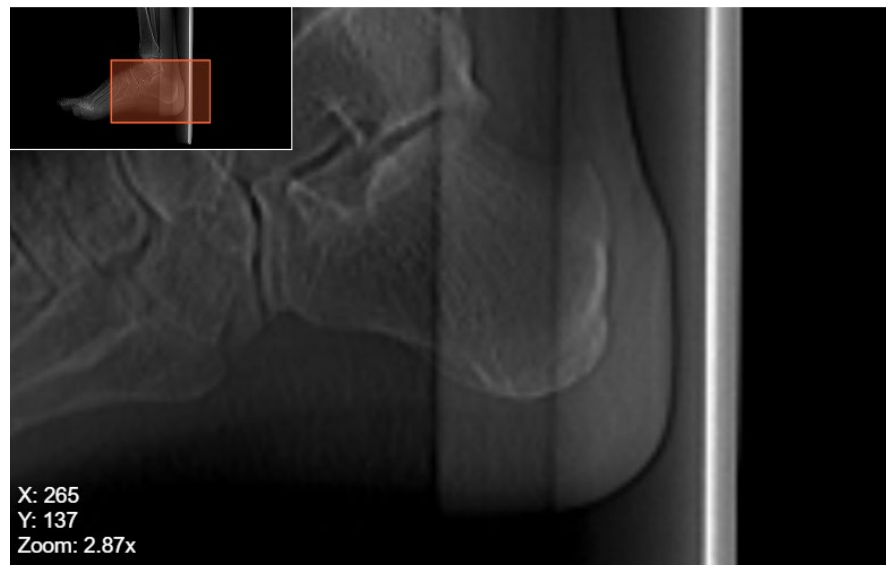


Figura 13. Componente dicom-instance-thumbnail al realizar zoom a una imagen

6. Resultados

Se desarrolló una herramienta para dibujar polígonos en Atix, de forma que estos sean mostrados sobre las instancias que se cargan desde un servidor. Una vez las instancias han sido cargadas, el usuario puede seleccionar la herramienta para dibujar polígonos y obtener datos del área que haya delimitado.

En primer lugar, el usuario selecciona el logo de la herramienta en el menú dicom-tools. Esto actualiza la herramienta seleccionada en el módulo dicom-state, y permite que cuando se realice un click, este sea traducido a coordenadas relativas del canvas y finalmente llegue al módulo dicom-measurements. Allí, este módulo ejecuta la lógica para evaluar si al añadir el nuevo punto se cumplen con las reglas definidas anteriormente. Si no se detectan errores, se dibujan las medidas en un canvas superpuesto al de la imagen DICOM (*Figura 14*).



Figura 14. Segmentación por contornos realizada dentro de Atix

Mientras se está creando el polígono, el usuario mueve un punto activo que se resalta en amarillo. Para añadir el punto activo al polígono basta con hacer click. Mientras el punto esté activo hay una línea punteada que lo conecta con el primer punto del polígono, de forma que si el usuario quiere cerrar el polígono sepa donde hacer click (*Figura 15*).

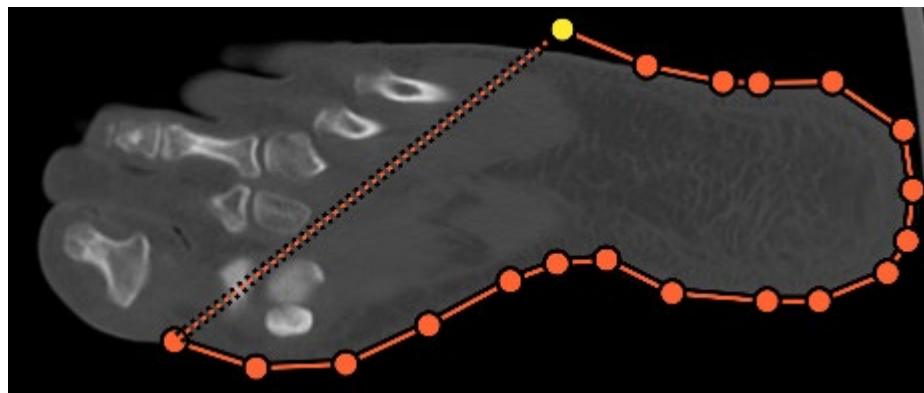


Figura 15. Creación de un contorno

Una vez el usuario haga click en el primer punto del polígono, lo cierra. Si el usuario desea eliminar el polígono puede hacer click sobre un punto del polígono y oprimir la tecla suprimir. Por otra parte, si el usuario desea editar el contorno puede seleccionar uno de los puntos y moverlo, o seleccionar una arista y mover todos los puntos como un conjunto (*Figura 16*). Así, mientras lo mueve se mostrará un contorno punteado para que el usuario pueda ajustar dónde lo pondrá.

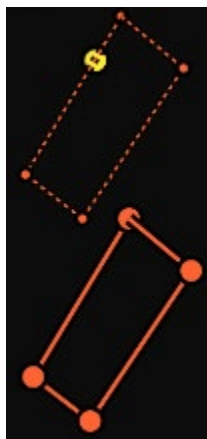


Figura 16. Polígono siendo movido.

El cálculo de áreas del polígono trazado se hace enviando las coordenadas de cada punto al componente dicom-viewer. Allí las coordenadas de los puntos se ajustan a las coordenadas de la imagen original y se determina qué pixeles están dentro del polígono siguiendo el método descrito anteriormente. De esta forma, a medida que se recorren los pixeles va calculando el área y la intensidad de los pixeles dentro del polígono. Finalmente se muestra el cálculo de la intensidad y el área en el menú dicom-plugin-menu

7. Conclusión y trabajo futuro

La segmentación por contornos es una herramienta valiosa para el análisis de imágenes biomédicas. Incorporar esta funcionalidad al visualizador web de imágenes médicas Atix permitirá a profesionales de la salud cuantificar áreas de interés y interactuar con segmentaciones hechas previamente. Sin embargo, queda pendiente realizar una validación con profesionales para identificar si la herramienta es intuitiva y fácil de usar. Además, se debe definir en qué etiqueta se almacenarán los polígonos dentro del formato DICOM e incorporar esta funcionalidad a las segmentaciones. Asimismo, es deseable incorporar funcionalidades para eliminar o añadir un punto a los polígonos luego de que han sido creados.

Por otra parte, a futuro se espera poder usar esta herramienta para construir poliedros. Para esto, se deben poder crear polígonos en diferentes cortes de forma que estos delimiten una misma entidad. También se puede considerar incluir algoritmos de detección de bordes para ajustar los puntos a medida que se crea un polígono, de forma que la herramienta pase de ser manual a interactiva.

Finalmente, el desarrollo de un algoritmo para segmentar las regiones definidas por el puntaje ASPECTS (Pexman et al., 2001) a partir de tomografías craneales está siendo desarrollado como parte del proyecto Circa. Se espera que dicho algoritmo pueda ser desplegado siguiendo el diseño de plugins para Atix. Con esto, una vez la imagen sea segmentada se mostrará en Atix para que un especialista pueda consultar el puntaje, analizar la segmentación y corregir los contornos en caso de que sea necesario. Así, se agilizará el proceso diagnóstico de pacientes que hayan sufrido accidentes cerebrovasculares isquémicos. Esto es un ejemplo de las aplicaciones que se espera poder desarrollar utilizando esta herramienta de segmentación.

8. Referencias

- Bendersky, E. (2008, August 15). *Intersection of 1D segments*. Eli Bendersky's Website. <https://eli.thegreenplace.net/2008/08/15/intersection-of-1d-segments>
- Camacho Daza, J. C., Rincón Diaz, N., Hernández Hoyos, M., & Aparicio Baquen, C. C. (2021). *Desarrollo de plugins web de procesamiento de imágenes médicas*. <https://repositorio.uniandes.edu.co/handle/1992/52895>
- Cano, J. P., & Hernández Hoyos, M. (2021). *Visualizador de imágenes médicas para dar trazabilidad a tumores cerebrales*. <https://repositorio.uniandes.edu.co/handle/1992/52887>
- de Berg, M., Cheong, O., van Kreveld, M., & Overmars, M. (2008). Computational geometry: Algorithms and applications. In *Computational Geometry: Algorithms and Applications*. Springer Berlin Heidelberg. <https://doi.org/10.1007/978-3-540-77974-2>
- González, R. C., & Woods, R. E. (2018). *Digital Image Processing* (Fourth edition). Pearson.
- Hankey, G. J. (2017). Stroke. *Lancet (London, England)*, 389(10069), 641–654. [https://doi.org/10.1016/S0140-6736\(16\)30962-X](https://doi.org/10.1016/S0140-6736(16)30962-X)
- Lassalle, L., Turc, G., Tisserand, M., Charron, S., Roca, P., Lion, S., Legrand, L., Edjlali, M., Naggara, O., Meder, J. F., Mas, J. L., Baron, J. C., & Oppenheim, C. (2016). ASPECTS (Alberta Stroke Program Early CT Score) Assessment of the Perfusion-Diffusion Mismatch. *Stroke*, 47(10), 2553–2558. <https://doi.org/10.1161/STROKEAHA.116.013676>
- Meneses Castillo, J. M., Hernández Hoyos, M., & Uriza Hoyos, L. F. (2020). *Visualizador de imágenes médicas para el seguimiento de tumores cerebrales*. <https://repositorio.uniandes.edu.co/handle/1992/51496>
- Open Health Imaging Foundation. (2023). *Cornerstone*.
- Pexman, J. H. W., Barber, P. A., Hill, M. D., Sevick, R. J., Demchuk, A. M., Hudon, M. E., Hu, W. Y., & Buchan, A. M. (2001). Use of the Alberta Stroke Program Early CT Score (ASPECTS) for Assessing CT Scans in Patients with Acute Stroke. *AJNR: American Journal of Neuroradiology*, 22(8), 1534. [/pmc/articles/PMC7974585/](https://pubs.rsospub.com/PMC7974585/)
- Rajendra Chaudhari, P. (2020). *Find point of 2D Line Intersection*. Opendgenus IQ. <https://iq.opendgenus.org/2d-line-intersection/>
- Rodríguez Castiblanco, J. C., Pérez Escalante, M. A., & Hernández Hoyos, M. (2022). *Desarrollo de plugins web de procesamiento de imágenes médicas*. <https://repositorio.uniandes.edu.co/handle/1992/55295>
- The Medical Imaging Technology Association. (2023). *The DICOM Standard*. <https://www.dicomstandard.org/>

- Trimpl, M. J., Primakov, S., Lambin, P., Stride, E. P. J., Vallis, K. A., & Gooding, M. J. (2022). Beyond automatic medical image segmentation-the spectrum between fully manual and fully automatic delineation. *Physics in Medicine and Biology*, 67(12). <https://doi.org/10.1088/1361-6560/AC6D9C>
- van Timmeren, J. E., Cester, D., Tanadini-Lang, S., Alkadhi, H., & Baessler, B. (2020). Radiomics in medical imaging—"how-to" guide and critical reflection. *Insights into Imaging*, 11(1). <https://doi.org/10.1186/S13244-020-00887-2>
- Vicerrectoría de Investigación. (2020). *Atix. Manual del usuario* (No. 4). Transferencia Uniandes.
- Yushkevich, P. A., Piven, J., Hazlett, H. C., Smith, R. G., Ho, S., Gee, J. C., & Gerig, G. (2006). User-guided 3D active contour segmentation of anatomical structures: Significantly improved efficiency and reliability. *NeuroImage*, 31(3), 1116–1128. <https://doi.org/10.1016/J.NEUROIMAGE.2006.01.015>
- Zwanenburg, A., Leger, S., Vallières, M., & Löck, S. (2016). Image biomarker standardisation initiative - feature definitions. *Computing Research Repository*, *abs/1612.07003*. <http://arxiv.org/abs/1612.07003>