Instituto Tecnológico de Buenos Aires - ITBA
Escuela de Ingeniería y Tecnología

Software para facilitar la localización de estructuras subcorticales que son blanco de electroestimulación

Autor:

Simón Oxenford (Legajo: 56210)

Tutor:

Dr. Ing. Federico E. Milano

Consultor médico:

Dr. Eduardo Pasqualini

Trabajo final presentado para la obtención de título de Bioingeniero

Buenos Aires
9 de agosto de 2019

Resumen

La localización de los ganglios basales en imágenes de Resonancia Magnética Nuclear de 1.5T y 3T es un problema difícil, inclusive para neurocirujanos especializados. Una correcta localización de estos núcleos es importante para lograr buenos resultados en procedimientos quirúrgicos de implantación de electrodos profundos para estimulación cerebral profunda (Deep Brain Stimulation, DBS). En este trabajo aplicamos métodos de registración deformable de atlas cerebrales de reciente publicación para obtener una localización adecuada de estas estructuras. A su vez, reconstruimos la localización de los electrodos a partir de imágenes de Tomografía Computada con la finalidad de ver la disposición de los contactos con respecto a los núcleos. De esta forma se plantea asistir al cirujano en la configuración de la estimulación de cada paciente. Evaluamos la localización de los núcleos por medio de la comparación con imágenes pre y post operatorias en base a casos reales que cuentan con dichos estudios. Por último, extendimos un software de análisis y visualización de imágenes médicas al que integramos los algoritmos ya mencionados para que los resultados puedan ser validados por médicos especialistas.

Índice

Siglas VII

1. Introducción **1**
1.1. Estimulación cerebral profunda (DBS) . . . 1
1.2. Motivación . . . 3
1.3. Actualidad . . . 4
1.4. Herramientas utilizadas . . . 5
1.5. Diagrama general del software . . . 7

2. Localización de estructuras subcorticales **9**
2.1. Métodos directos e indirectos . . . 9
2.2. Registración . . . 11
2.2.1. Métricas . . . 12
2.2.2. Interpoladores . . . 13
2.2.3. Transformaciones . . . 13
2.2.4. Optimizadores . . . 14
2.3. Atlas . . . 15
2.4. Descripción general del método . . . 16
2.5. Implementación en ANTs . . . 17
2.5.1. Primer paso: registración de similitud . . . 17
2.5.2. Segundo paso: registración afín . . . 18
2.5.3. Tercer paso: registración deformable . . . 19
2.6. Resultados preliminares . . . 20

3. Reconstrucción de los electrodos **23**
3.1. Obtención de la máscara cerebral . . . 23
3.2. Segmentación de los electrodos . . . 23
3.3. Dirección del electrodo . . . 25
3.4. Imagen del electrodo . . . 26
3.5. Ubicación de los contactos . . . 27
3.6. Modelo del electrodo . . . 27

4. Interfaz de usuario **31**
4.1. Selección del paciente . . . 33
4.2. Cargar imágenes . . . 33
4.3. Parámetros de procesamiento . . . 34
4.4. Visualización . . . 36

4.5. Visualización final 37

5. Resultados **39**
5.1. Caso 1 41
5.2. Caso 2 42
5.3. Caso 3 43
5.4. Caso 4 44
5.5. Caso 5 (adicional) 45

6. Conclusiones **47**
6.1. Limitaciones 47
6.2. Trabajo futuro 47
6.3. Máquina virtual 48

A. Opciones de visualización **49**

B. Instalación de la máquina virtual **55**

Bibliografía **57**

Siglas

AC *Anterior Commissure*

ANTs *Advanced Normalization Tools*

CT *Computed Tomography*

DBS *Deep Brain Stimulation*

GPe *Globus Pallidus external*

GPi *Globus Pallidus internal*

MCP *Mid-Commissural Point*

MER *Micro Electrode Recordings*

MRI *Magnetic Resonance Imaging*

PC *Posterior Commissure*

PFC Proyecto Final de Carrera

RN *Red Nucleus*

SN *Substantia Nigra*

STN *Subthalamic Nucleus*

1 | Introducción

Este Proyecto Final de Carrera (PFC) trata principalmente del procedimiento quirúrgico de implantación de electrodos para *Deep Brain Stimulation* (DBS) y el procesamiento de imágenes biomédicas para la ayuda a la toma de decisiones en este tipo de cirugías. En este trabajo se utilizaron imágenes pre y post operatorias, las primeras para la localización de estructuras subcorticales y las últimas para la reconstrucción de los electrodos implantados con sus respectivos contactos. De esta forma, se busca asistir al médico en la ubicación y programación de dichos electrodos.

El las próximas secciones se describirá brevemente en qué consiste la cirugía de DBS [1, 2], cuál es la motivación de este trabajo y qué herramientas fueron utilizadas para realizar el mismo. Finalmente se presentará un diagrama resumido de cómo es el funcionamiento del software que se plantea en este PFC.

1.1. Estimulación cerebral profunda (DBS)

La cirugía para la implantación de electrodos de DBS es un procedimiento invasivo, durante el cual se colocan electrodos en el cerebro, estimulando regiones específicas del mismo. Los electrodos son programados para establecer la configuración de la estimulación por medio de un neuroestimulador. El mismo se suele implantar en la zona subclavia, tal como sucede con los marcapasos.

Las zonas del cerebro que se buscan estimular son estructuras subcorticales, en particular los núcleos basales. Estos controlan la actividad cortical a través del tálamo junto con el cerebelo. Los mismos tienen gran importancia para el control de los movimientos voluntarios y otras funciones corticales. Entre ellos se encuentran el *Subthalamic Nucleus* (STN), *Substantia Nigra* (SN), *Globus Pallidus external* (GPe), *Globus Pallidus internal* (GPi) y *Striatum*. En la Figura 1.1 se encuentra una ilustración que representa estas estructuras.

En este tipo de cirugía se busca intervenir en los circuitos cerebrales por medio de estimulación eléctrica. La estimulación puede inhibir o excitar distintos elementos neuronales regulando patrones de activación. El mecanismo de acción de esta estimulación sobre los distintos circuitos cerebrales no está totalmente comprendida.

Los pacientes a los que se les lleva cabo esta cirugía suelen presentar movimientos anormales, aunque también se realiza para otras enfermedades neuronales. Entre ellas se incluyen Parkinson, Distonía y Temblor Esencial, y lo que se busca es aliviar los síntomas de las mismas.

El núcleo a estimular se decide dependiendo de la condición que presente el paciente. Por ejemplo, el STN es un blanco común para pacientes con Parkinson avanzado en los

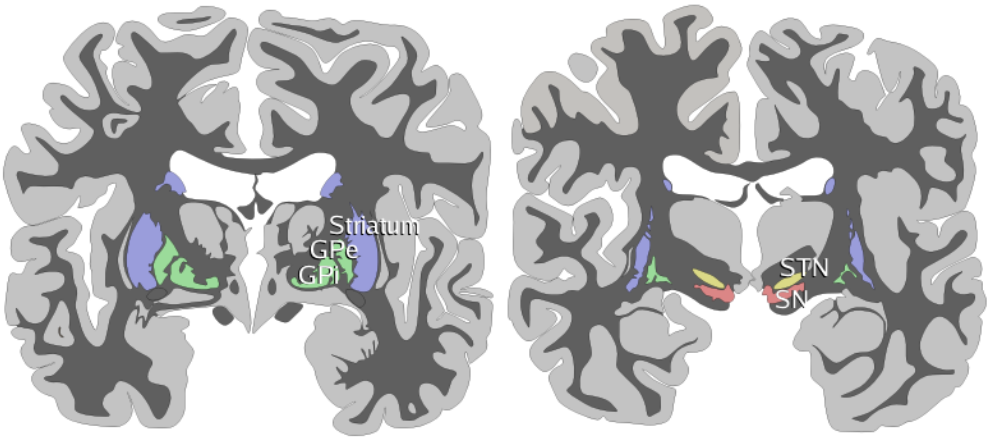


Figura 1.1: Ilustración de los ganglios basales en cortes coronales. Fuente: Wikipedia.

cuales la terapia con drogas no logra mejoras significativas. El Temblor Esencial es otra condición tratada en esta cirugía, donde se suele estimular el VIM. También se ha estudiado su aplicación en pacientes con epilepsia refractaria, optando por una estimulación al núcleo anterior del tálamo (ANT).

En las semanas previas a la operación se obtienen imágenes *Magnetic Resonance Imaging* (MRI) del paciente y se pasa a identificar las estructuras basales de interés. De esta forma se logra obtener una estimación del lugar del cerebro dónde irá dirigido el electrodo. La localización no es trivial, puesto que los núcleos se encuentran en zonas que no son fáciles de diferenciar en una resonancia de uso clínico ($1,5T$ o $3T$) y relativamente pequeñas (por ejemplo el STN mide aproximadamente $4mm \times 5mm \times 6mm$).

Para realizar la operación se utiliza un marco de estereotaxia, el cual permite localizar un blanco dentro del cerebro con gran precisión [3]. La estereotaxia requiere la adquisición de una imagen de *Computed Tomography* (CT) preoperatoria con el marco colocado en el paciente. De esta forma el marco funciona como referencia y se pasa la ubicación a estimular de la resonancia al sistema de coordenadas del marco.

En la cirugía se utiliza la técnica de *Micro Electrode Recordings* (MER) para determinar de forma más precisa la zona cerebral en que se encuentra. Esta consiste en utilizar electrodos de exploración para tomar mediciones de la actividad de las neuronas. Dependiendo de la zona se obtienen distintos patrones de activación y, por lo tanto, se la puede identificar.

Luego se colocan los electrodos que se utilizarán para la estimulación. Hay distintos modelos de electrodos, que difieren en la cantidad de contactos, su separación, longitud, espaciamiento y diámetro. La elección del modelo a utilizar va a depender de la condición a tratar. En la Figura 1.2 hay un ejemplo de un electrodo comercial.

Para la programación de los electrodos hay distintas opciones a configurar. Se puede elegir entre una estimulación monopolar (se utiliza el neuroestimulador como masa), bipolar (estableciendo una diferencia de potencial entre dos contactos) o multipolar. Generalmente se estimula controlando la corriente entregada, configurando la intensidad, ancho de pulso y frecuencia. Estos parámetros se determinan en última instancia en base a prueba y error y evaluando la respuesta clínica del paciente.

Figura 1.2: Electrodo Medtronic 3387. Este electrodo tiene 1,27mm de diámetro y 4 contactos de 1,5mm de longitud y 1,5mm de espaciamiento. Fuente: Medtronic plc (Fridley, MN, USA).

1.2. Motivación

En este proyecto se colaboró con el Dr. Eduardo Pasqualini (Hospital Alemán y Hospital Militar Central), quien se mostró interesado en obtener una estimación certera de dónde dirigir los electrodos de forma previa a que los mismos sean implantados. En caso de obtener una localización confiable en primera instancia se podría optar por hacer una ablación de la zona por medio de técnicas de radioterapia en vez de realizar una cirugía invasiva. Un primer objetivo que se plantea en este PFC es entonces lograr reconocer las estructuras cerebrales de interés para la cirugía de DBS.

- **Objetivo 1:** Segmentar las estructuras subcorticales a partir de MRI preoperatoia.

Por otro lado, como fue explicado, una vez implantados los electrodos se pasa a programar los mismos. En este punto surge otro objetivo: lograr reconstruir los electrodos y ver cómo están dispuestos en relación a las estructuras blanco. Esta información tal vez pueda ayudar a la hora de configurar la estimulación. Un caso podría ser una primer sesión postoperatoria donde se empieza a probar qué estimulación configurar al paciente. Se podría realizar una primer aproximación estimulando con el contacto más cercano al núcleo en vez de ir probando a ciegas.

- **Objetivo 2:** Reconstruir los electrodos a partir de CT postoperatoria.

Actualmente no se entiende completamente cómo es que los electrodos modulan las actividades neuronales. Por lo tanto, tampoco es claro qué zona es preferible estimular ni cuál es la mejor opción cuando se lleva a cabo la programación. Tal vez con el uso de este software (teniendo en cuenta sus limitaciones y a modo de investigación) se pueda lograr comprender mejor el mecanismo de DBS. Esto requeriría de un análisis extensivo involucrando muchos pacientes. Si se sabe con certeza exacta la ubicación de los contactos y de las estructuras (e incorporando más información acerca de los circuitos cerebrales que intervienen en la regulación del movimiento) se podrían corresponder las distintas respuestas clínicas que tienen distintos pacientes con los circuitos neuronales que fueron intervenidos. La estimulación se podría simular por medio de métodos de elementos finitos y cuantificar el volumen de tejido que fue activado. Para poder analizar distintos pacientes y usar los procesamientos desarrollados, se plantea también hacer una interfaz de usuario que los incorpore.

- **Objetivo 3:** Desarrollar una interfaz de usuario que permita realizar los distintos procesamientos y evaluar los resultados.

En las Secciones 2, 3, y 4 se explican, respectivamente, las estrategias utilizadas para la realización de estos objetivos. Luego, en la Secciones 5 y 6, se discuten los resultados y conclusiones finales.

1.3. Actualidad

Al empezar a investigar respecto al área de DBS y al uso de procesamiento de imágenes médicas para su asistencia se notó un amplio crecimiento en los últimos años. En el gráfico de la Figura 1.3 se puede ver los resultados por año al buscar "*DBS OR (deep brain stimulation)*" en PubMed. Esto sirve para darse una idea de la creciente investigación sobre el tema.

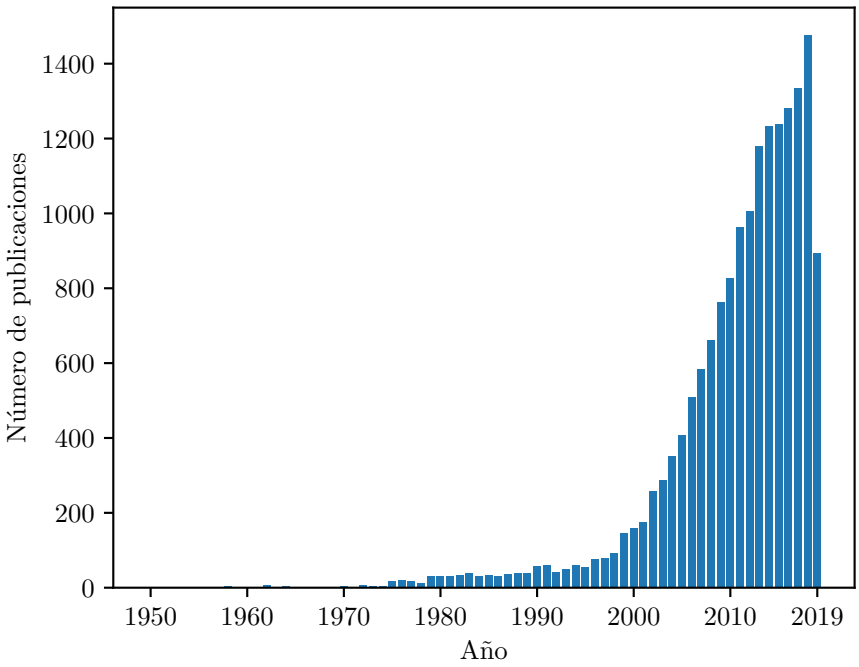


Figura 1.3: Número de publicaciones por año en PubMed al buscar "*DBS OR (deep brain stimulation)*". Aclaración: la búsqueda se realizó en el mes de Junio de 2019.

Hay distintas opciones de software que buscan ayudar al médico en la visualización de las estructuras blanco y la programación de los electrodos. Una primera clasificación general de los mismos puede ser respecto si son o no comerciales.

Por lo general, el software de uso comercial cuenta con aprobaciones de respectivos entes regulatorios (por ejemplo: ANMAT, FDA, CE) indicando que su uso es eficaz y seguro. Son como una caja negra para el usuario ya que este no sabe el procesamiento que se lleva a cabo para lograr los resultados. Suelen ser de uso simple e intuitivo, pensado para

que sea utilizado por un médico y no tenga que intervenir mucho en el funcionamiento. Para el caso de DBS, un software comercial es el GUIDE DBS [4] desarrollado por Boston Scientific. El mismo permite la visualización de los electrodos y simular el resultado de la estimulación. Otra alternativa es el SureTune [5] de Medtronic.

En cambio, el software libre, sí requiere de mayor intervención y conocimiento por parte de usuario para su funcionamiento. Al entender el procesamiento que se lleva a cabo y sus limitaciones el usuario está más capacitado para evaluar los resultados y ajustar los parámetros del procesamiento para obtener una mejor salida. Así se pueden obtener resultados similares o tal vez aún mejores que los de software comercial. Al no contar con aprobaciones de entes regulatorios, este tipo de software se puede actualizar diariamente y contar con los últimos avances realizados en el área. Se puede utilizar también para investigación y aportar conocimiento público.

La principal herramienta en el ámbito de investigación y software libre es el Lead-DBS [6]. Este permite la reconstrucción de electrodos, localización de estructuras por medio de distintas técnicas de registración, simulación de estimulación y estudios de relación entre las áreas estimuladas y afectadas por medio del conectoma (mapa de conexiones neuronales del cerebro). A pasar de que es de código abierto está hecho en Matlab, por lo que requiere una licencia del mismo para su uso. Este software integra distintos avances que realizan grupos de investigación interesados en DBS proporcionando actualizaciones a medida que ocurren, por lo que se mantiene al día con las últimas novedades del área. Su uso no es trivial, requiere conocimientos de Matlab e intervención en las distintas etapas del procesamiento.

1.4. Herramientas utilizadas

En este PFC se decidió trabajar utilizando herramientas de uso libre y se buscó desarrollar una interfaz de usuario para que el médico pueda utilizarlas de forma sencilla. A continuación se hace una breve descripción de las mismas.

Ubuntu

Ubuntu es una distribución de Linux de código abierto. La elección de usar el ambiente Linux es que varias de las herramientas de neuroinformática (como por ejemplo FreeSurfer y FSL) solo se pueden ejecutar en este. Además, siendo principalmente un proyecto de desarrollo de software, se encontró mayor facilidad al usar el entorno con este sistema operativo que con Windows.

ITK

El *Insight Toolkit* (ITK) [7] es un software código abierto para manipular imágenes médicas y realizar segmentación y registración. Está escrito en C++ y es multi-plataforma. Basado en ITK está hecho el *SimpleITK*, que tiene librerías en Python y está pensado para facilitar el uso y prototipado.

VTK

EL *Visualization Toolkit* (VTK) [8] es un software de código abierto para la manipulación y visualización de datos científicos. Permite hacer renders en tres dimensiones y gráficos, entre otras cosas. Así como ITK, está escrito en C++ y es muti-plataforma.

Qt

Qt trae librerías en C++ para el desarrollo de interfaces de usuario y aplicaciones. Su uso para software libre es gratis pero si se desea una aplicación comercial se debe pagar una licencia.

3DSlicer

El 3DSlicer [9] es un software para visualización y procesamiento de imágenes médicas y está escrito principalmente utilizando las librerías de ITK, VTK y Qt. Es abierto y de uso libre y cuenta con una amplia comunidad de desarrolladores y usuarios.

Este software permite integrar módulos hechos por usuarios de forma muy simple. Estos pueden estar escritos en C++ o en Python y se puede acceder a todas las funcionalidades ya incorporadas en el 3DSlicer. Además, es posible diseñar una propia interfaz de usuario utilizando los componentes que se deseen.

ANTs

Advanced Normalization Tools (ANTs) [10] es un software abierto, escrito en C++ sobre las librerías de ITK y se ejecuta por línea de comando. Su principal función es el de la normalización (registración).

Python

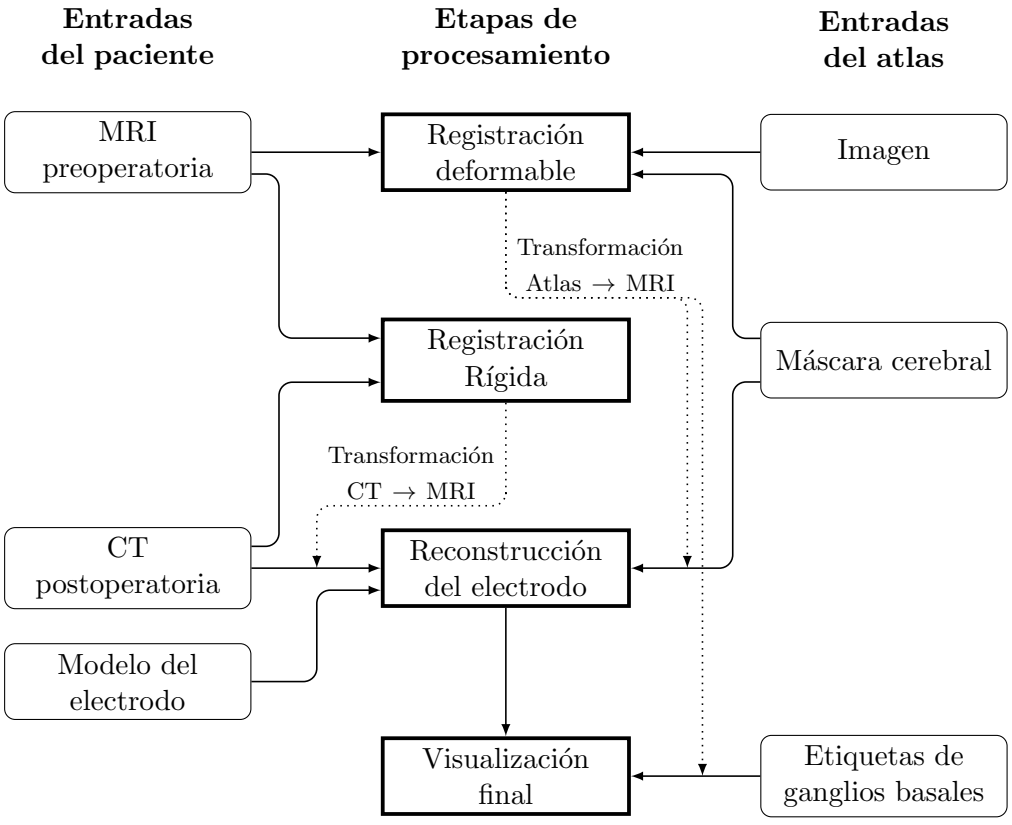
Python es un lenguaje de programación de uso libre y utilizado para una gran variedad de aplicaciones. Como se mencionó antes, se pueden hacer módulos en 3DSlicer programando en Python. Como ventaja de esta función es que se pueden incorporar módulos de Python al procesamiento realizado (por ejemplo sci-py).

1.5. Diagrama general del software

A continuación se presenta un diagrama que resume el funcionamiento del software que se desarrolló en este PFC. Hay que aclarar que se omiten algunos pasos que luego se detallarán. En el diagrama se distinguen cuatro procesamientos:

- Registración deformable: Se registra la imagen de MRI del paciente con la imagen de MRI de un atlas, obteniendo la transformación de un espacio a otro (Sección 2).
- Registración rígida: Se registran las imágenes pre y post operatorias del paciente para luego obtener los resultados en un mismo espacio de referencia.
- Reconstrucción del electrodo: A partir de la CT del paciente y el modelo del electrodo se obtiene la localización espacial de los mismos En este algoritmo también se utiliza la máscara cerebral del atlas (Sección 3).
- Visualización final: Se obtienen tanto los electrodos reconstruidos como las distintas estructuras cerebrales segmentadas en el espacio del paciente (Figura 4.7).

Figura 1.4: Diagrama general del software. Se presentan en tres columnas las entradas propias del paciente (sus imágenes y modelo de electrodo), las entradas del atlas (imagen, máscara cerebral y etiquetas) y las distintas etapas de procesamiento en las que estas se ven involucradas.



2 | Localización de estructuras subcorticales

La localización de estructuras subcorticales es un paso fundamental para la comprensión de la anatomía propia del paciente y la estimación de la zona a estimular. Existen muchas técnicas y herramientas disponibles en lo que respecta a la localización de estructuras subcorticales a partir de imágenes del cerebro [1, 6, 11, 12, 13, 14]. Algunos investigadores utilizan métodos indirectos, basándose en puntos de referencia conocidos o bien atlas que incorporan información anatómica o funcional. Otros utilizan visualización directa sobre MRI obtenida mediante distintos protocolos [11]. Al día de la fecha, no ha sido demostrado que alguno de estos métodos sea superior a los demás, sino que es un tema que continúa debatiéndose [1]. En esta sección se revisarán algunas de estas técnicas y se explicará el método elegido en este PFC.

2.1. Métodos directos e indirectos

La clasificación entre métodos directos e indirectos es la más general. En los métodos directos se plantea visualizar las estructuras a partir de las imágenes adquiridas sin otra información contextual adicional. Para la localización de estructuras subcorticales generalmente se utilizan imágenes T2 o T2* de MRI. Esto se debe a la hipointensidad que presenta la señal en los ganglios basales, facilitando su identificación [15]. Es necesario contar con un ojo entrenado para diferenciar los sutiles cambios en los niveles de grises y un amplio conocimiento de neuroanatomía. La experiencia en este dominio para la realización del presente trabajo fue aportada por medio de la colaboración con el Dr. Pasqualini. En la imagen de la Figura 2.1 se puede ver un ejemplo de este tipo de localización.

Los métodos indirectos, por su parte, utilizan información adicional. Esta se puede obtener a partir de puntos de referencia más fácilmente identificables en la imagen del paciente o adaptando un atlas a la anatomía en cuestión [11].

Los puntos de referencia más utilizados en neuroimágenes son el *Anterior Commissure* (AC) y el *Posterior Commissure* (PC). Una vez localizados estos dos puntos y su punto medio (*Mid-Commissural Point* (MCP)) se puede hacer una estimación acerca de otras estructuras por medio de medidas tomadas a partir del atlas de referencia.

Un atlas consiste de una imagen con sus estructuras identificadas (en la Sección 2.3 se explicará más en profundidad este concepto). Por ejemplo (de [1]), usando el atlas Schaltenbrand Wahren:

- STN: $10\,mm$ lateral, $2,5\,mm$ posterior y $4\,mm$ ventral del MCP.
- GPi: $20\,mm$ lateral, $3\,mm$ anterior y $4\,mm$ ventral del MCP.
- Vim: $12\,mm$ lateral, $5\,mm$ posterior y $0\,mm$ del MCP.

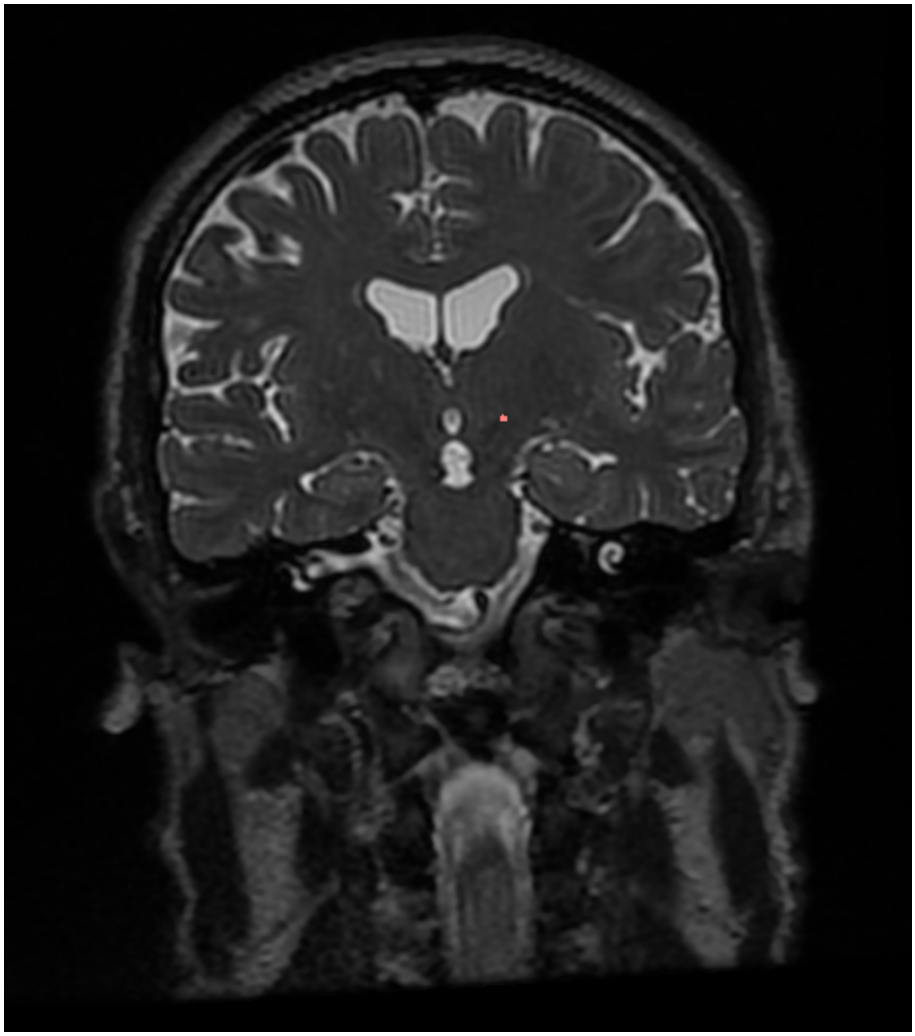


Figura 2.1: Método directo sobre una imagen de resonancia T2. El fiduciario indica el punto más externo del STN.

Una clara limitación del método indirecto basado en puntos fiduciarios es que se utilizan las mismas relaciones para todos los pacientes. Para mitigar este problema, es posible utilizar la información de los atlas en forma específica para cada caso. Por ejemplo, teniendo en cuenta la longitud del segmento AC-PC, utilizar dicha magnitud para escalar las otras distancias [1].

Finalmente, otro método para trasladar la información del atlas al paciente, es por medio de técnicas de procesamiento de imágenes, específicamente por medio de la registración de imágenes. Este es el procedimiento que actualmente refleja el estado del arte y con el que se trabajó a lo largo de este PFC. A continuación se va a detallar en qué consiste la registración de imágenes y luego se describirán los distintos atlas utilizados.

2.2. Registración

La registración de dos imágenes es la puesta en correspondencia de una imagen con la otra. El resultado es una **transformación** que indica esta correspondencia (pixel a pixel o vóxel a vóxel). La Figura 2.2 muestra un ejemplo de dos cilindros y la transformación de uno a otro. En esta sección se describirá la registración siguiendo los manuales [7, 10, 16]. Estos explican el funcionamiento de distintas herramientas que se usaron en este PFC.

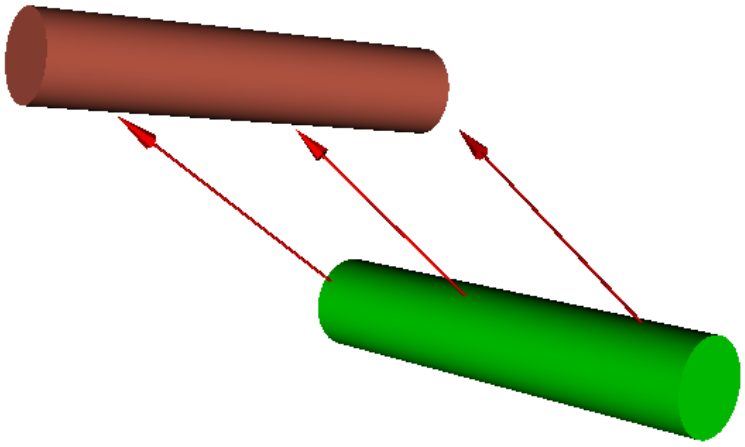


Figura 2.2: Las flechas en rojo muestran la transformación de tres puntos de un cilindro al otro. Esta transformación indica cómo se corresponden los vóxeles de cada uno. (Las flechas están acortadas para facilitar la visualización).

En el proceso de registración están involucradas dos imágenes, denominadas la imagen móvil $I_M(x)$ y la imagen fija $I_F(x)$, de dimensión d (para imágenes de MRI y CT $d = 3$). Cada una está definida en su dominio espacial $\Omega_M \subset \Re^d$ y $\Omega_F \subset \Re^d$. La registración consiste en un problema de optimización donde se busca encontrar la transformación $T(x)$ tal que $I_M(T(x))$ esté alineada espacialmente con $I_F(x)$. La bondad de una determinada alineación espacial es medida por lo que se denomina una **métrica**. Existen diferentes métricas para diferentes problemas de registración.

En la registración también se usa un interpolador para evaluar las intensidades de $I_M(x)$ en los puntos que no estén ubicados en la grilla del espacio Ω_F. La métrica provee una medida de que tan bien se corresponden $I_F(x)$ y $I_M(T(x))$. Esta medida cuantitativa es optimizada por el optimizador, cuyo espacio de búsqueda está definido por los parámetros de T. En el diagrama de flujo de la Figura 2.3 se puede ver cómo se relacionan estos componentes.

No se incluyó en el diagrama la estrategia de utilización de pirámides de multiresolución. Esto consiste en realizar la registración en una serie de pasos. Primero partiendo de las imágenes submuestreadas y/o con un filtro gaussiano, y en las siguientes etapas ir disminuyendo este efecto. Esto permite registrar primero las características más

generales de las imágenes para luego comenzar a evaluar cada vez más detalles de las imágenes en el proceso de registración.

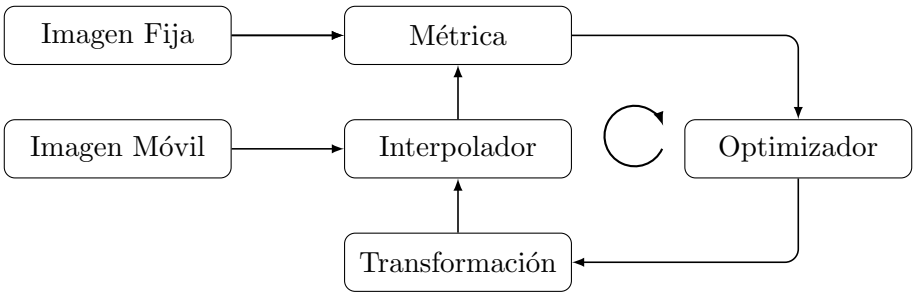


Figura 2.3: Diagrama de flujo básico de una registración.

2.2.1. Métricas

Hay distintas métricas disponibles y descriptas en la literatura científica. Muchas de ellas se encuentran implementadas en software de procesamiento de imágenes de código abierto. A continuación se presentan algunas. La elección de qué métrica de similitud usar va a depender del problema de registración que se trate y el enfoque que se le de al mismo.

Diferencia de cuadrados

Esta métrica toma la diferencia de la intensidad de gris entre la imagen fija y la imagen móvil interpolada para cada vóxel. Eleva al cuadrado y suma cada componente y luego se divide por el total de vóxeles.

$$\mathrm{MSD}(\mu, I_F, I_M) = \frac{1}{\Omega_F} \sum_{x_i \in \Omega_F} (I_F(x_i) - I_M(T_\mu(x_i)))^2 \tag{2.1}$$

En este caso μ indica los parámetros de la transformación T.

Información mutua

La información mutua se define en términos de la entropía. La entropía de una variable aleatoria A se define como:

$$H(A) = -\int p_A(a)\ \log p_A(a)\ da \tag{2.2}$$

y la entropía conjunta de dos variables aleatorias A y B como:

$$H(A,B) = \int p_{AB}(a,b)\ \log p_{AB}(a,b)\ dadb \tag{2.3}$$

Si A y B son independientes entonces $p_{AB}(a,b) = p_A(a)\, p_B(b)$ y $H(A,B) = H(A) + H(B)$. Sin embargo, si hay alguna dependencia entre las variables, entonces $H(A,B) < H(A) + H(B)$. La diferencia se define como información mutua:

$$I(A,B) = H(A) + H(B) - H(A,B) \tag{2.4}$$

Para imágenes, al ser discretas, no se trabaja con integrales sino con sumatorias. La densidad de probabilidad conjunta y marginal (p_{AB} y p_X respectivamente) se evalúan en intervalos discretos distribuidos de forma uniforme en el rango dinámico de las imágenes (histograma). Los valores de entropía luego se computan sumando a lo largo de los intervalos. Ver en la bibliografía para el cálculo concreto de la métrica.

Correlación cruzada

Para simplificar, primero se define:

$$\overline{I_F}(x) = I_F(x) - \frac{1}{\Omega_F} \sum_{x_i \in \Omega_F} I_F(x) \tag{2.5}$$

De la misma forma se tiene $\overline{I_M}(T_\mu(x))$. Entonces, la correlación cruzada se define como:

$$\mathrm{CC}(\mu, I_F, I_M) = \frac{\left(\sum_{x_i \in \Omega_F} \overline{I_F}(x_i)\ \overline{I_M}(T_\mu(x_i)) \right)^2}{\sum_{x_i \in \Omega_F} \overline{I_F}(x_i)^2\ \sum_{x_i \in \Omega_F} \overline{I_M}(T_\mu(x_i))^2} \tag{2.6}$$

2.2.2. Interpoladores

Como se mencionó en las secciones previas, no necesariamente los puntos del espacio de $I_M(T_\mu(x))$ son los mismos que $I_F(x)$, es decir Ω_F. Es por eso que se usa un interpolador.

Vecino más cercano

Este es el más simple, dónde a cada punto de Ω_F se la asigna la intensidad del vóxel más cercando de $I_M(T_\mu(x))$.

Lineal

En este caso, la intensidad es un promedio ponderado de los vóxeles circundantes, tomando su cercanía como peso.

B-Spline

El vecino más cercano y el lineal están ambos contenidos en el interpolador B-spline. Para el primero el orden de B-spline es igual a 0 y para el segundo el orden es 1. Se pueden tomar B-spline de órdenes mayores para lograr mayor calidad de interpolación. Esto requiere también mayor cómputo.

2.2.3. Transformaciones

Rígida

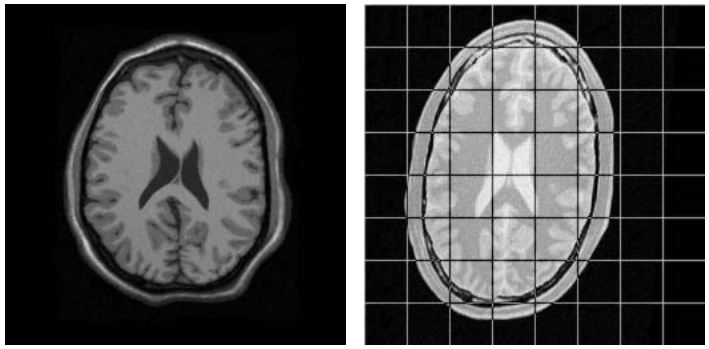
En la transformación rígida a la imagen se la trata como un cuerpo rígido, el cual se puede trasladar y rotar. Esta transformación para imágenes en tres dimensiones queda definida por seis parámetros (cantidad de rotación y traslación en cada dimensión). Ver Figura 2.4c.

Afín

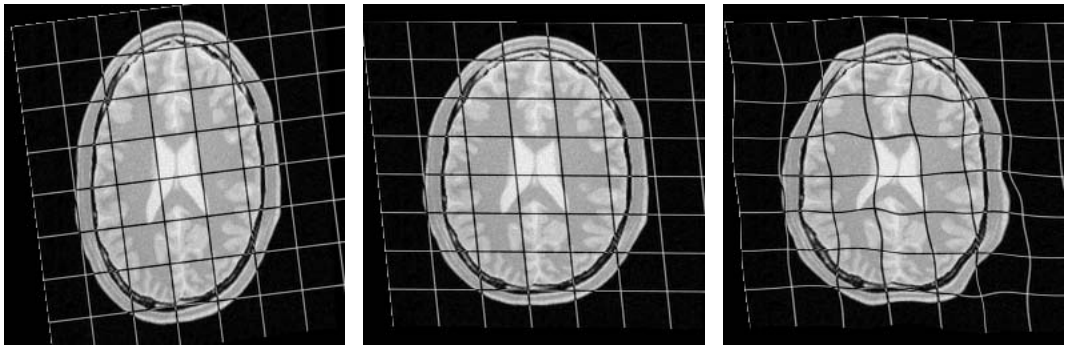
A esta transformación se le agrega la capacidad de escalar y cizalladura, agregando dos parámetro por dimensión más (12 parámetros para el caso de tres dimensiones). En la grilla de la Figura 2.4d se puede ver el resultado de la cizalladura.

Deformable

Una forma de representar la transformación deformable es mediante un conjunto de puntos de control definidos sobre una grilla. A cada uno de estos puntos se los puede desplazar e interpolando entre los mismos se obtiene la transformación de una imagen en la otra. La cantidad de parámetros de esta transformación varía dependiendo del espaciamiento de la grilla (que define la cantidad de puntos de control). Ver Figura 2.4e.



(a) Imagen fija. (b) Imagen móvil.



(c) Rígida. (d) Afín. (e) Deformable.

Figura 2.4: Ejemplos de distintas transformaciones (de [16]). Se incluye una grilla para facilitar la visualización de la transformación.

2.2.4. Optimizadores

Como técnica de optimización (optimizador) solo se mencionará el descenso por el gradiente, en el cual la dirección de búsqueda de un nuevo grupo de parámetros se toma como el negativo del gradiente de la métrica (función de costo). Esta se multiplica por una constante (paso del gradiente) para definir el nuevo punto a evaluar.

2.3. Atlas

Uno de los problemas en imágenes cerebrales es la identificación y localización de estructuras [17]. Un atlas cerebral se compone por un lado de la imagen del mismo (puede ser MRI o cortes histológicos) y por otro (y más importante) una indicación acerca cuáles son y dónde están las estructuras que presenta la imagen. A continuación se describirán distintos avances realizados a lo largo de los años para llegar a describir lo que se utilizó en este PFC.

Talairach y Tournoux

Talairach y Tournoux [18] introdujeron una solución a partir de definir un espacio de coordenadas para el cerebro. Este tiene el AC como origen, la línea AC-PC horizontal y la línea media cerebral en el plano vertical. A partir de una nueva imagen, es posible escalar y rotar la misma llevándola a ese sistema de coordenadas. Luego, utilizando los diagramas etiquetados que los autores realizaron, se infiere la localización de las estructuras en la nueva imagen. Los diagramas de Talairach y Tournoux están basados en un cerebro de una mujer de 60 años.

MNI-305

Este atlas presenta un cerebro promedio desarrollado por el Montreal Neurological Institute a partir de 305 imágenes de MRI. Primero se identificaron fiduciarios en las imágenes para llevarlas al espacio definido por Talairach y Tournoux. Luego, una vez todas en el mismo espacio, se realizó un promedio de sus intensidades de grises y así se obtuvo una sola imagen. A continuación se realizó una registración automática con 9 parámetros para llevar cada imagen original a la promediada [19]. El promedio de las imágenes luego de la registración automática es lo que se conoce como el espacio MNI.

Este avance permitió el uso de este cerebro promedio para realizar registración dentro de un software y trabajar con imágenes en un mismo espacio. A pesar de que está basado en el Talairach - Tournoux (y a veces se los llame de forma indistinta) no son exactamente iguales.

Previamente se ha mencionado que un atlas consistía en una imagen y sus etiquetas. En este caso, el espacio MNI es la imagen que luego es utilizada por distintos grupos para crear sus propias etiquetas. También se suele usar este espacio para registrar imágenes al mismo y crear nuevos cerebros promedio con respectivas etiquetas.

El cerebro promedio más actualizado e incorporado en distintos software de procesamiento de imágenes es el ICBM-152. Este se hizo registrando 152 mágenes MRI al MNI-305. Cuando se hace referencia al espacio MNI por lo general se refiere al ICBM-152 (así se lo llamará en este trabajo).

Xiao (2017)

Este atlas fue realizado utilizando imágenes correspondientes a 25 pacientes con Parkinson [20]. Las mismas fueron primero alineadas al espacio MNI por medio de una registración afín. Luego se realizó un procesamiento de registración grupal, donde se busca minimizar simultáneamente la diferencia en intensidad de gris entre el espacio y la imagen

transformada y la deformación que mapea el atlas a cada sujeto [21]. De esta forma se obtuvo un nuevo espacio (MNI-PD25).

Finalmente, para obtener las etiquetas, se segmentó directamente sobre el espacio MNI-PD25 de la siguiente forma: el núcleo caudado, putamen y tálamo a partir de T1w MPRAGE, el GPi y GPe a partir de T1w MPRAGE y T2* y el STN, SN y *Red Nucleus* (RN) a partir de T2*.

Milichenko (2018)

Este atlas tiene unicamente etiquetado el STN. Para lograrlo, primero se parte de imágenes $7T$ (a diferencia del anterior y de la mayoría de las investigaciones que trabajan con $1{,}5T$ o $3T$). Esta mayor definición permite la segmentación sobre imágenes propias del paciente. Tres médicos experimentados fueron los que segmentaron las estructuras en los pacientes y luego se determinó la segmentación final por un método de voto mayoritario.

En este estudio también se creó un espacio propio basado en el espacio MNI al cual fueron trasladadas las estructuras segmentadas. Hay mayor detalle de este procesamiento en [22].

Ewert (2018)

En este caso no se crea un nuevo espacio sino que las etiquetas están en el espacio MNI. El procesamiento realizado utiliza información de MRI, histológica y conectividad [23]. Para este atlas sí se obtienen segmentaciones de todos los núcleos basales e incluso subsegmentaciones de los mismos, representando distintas funciones.

2.4. Descripción general del método

El problema que se plantea resolver es la registración de una imagen incógnita a un atlas y así obtener la transformación de un espacio al otro. Esto va a permitir luego trasladar las etiquetas del atlas al paciente y de esta forma segmentar las estructuras de interés para una imagen nueva. Más allá de los parámetros propios de la transformación obtenidos por medio de la registración, hay parámetros definidos por el usuario que determinan el procedimiento por el cual se llevará a cabo la misma. Aquí se describe el método de forma general y en la siguiente sección el detalle de cómo se implementó en el software.

Si se quisiera realizar la registración deformable en un solo paso no se obtendrían los resultados que uno está buscando. Esto se debe a que hay muchos parámetros de la transformación por definir y el optimizador se guiará por mínimos locales que deformarán totalmente la imagen en cuestión. Para obtener resultados más coherentes se debe partir de una situación en la que la imagen incógnita y el atlas ya presentan cierta correspondencia.

Por eso se decidió llevar a cabo la registración por medio de una serie de etapas. La aproximación más general es llevar el centro geométrico o de masa de la imagen móvil al mismo punto que el de la fija. Luego se van aumentando progresivamente los grados de libertad por los que la imagen se puede desplazar. Una registración rígida, de seis parámetros, permite eliminar errores de rotación y traslación. Partiendo de esta se puede seguir por una transformación afín, que ya presenta alteraciones en los distintos ejes de forma independiente. A continuación sí se podría optar por una registración deformable,

la cual, partiendo de estos resultados, presentaría una mayor correspondencia. De esta forma se va guiando la registración para lograr el resultado buscado.

Cada una de estas etapas incluso esta dividida en una serie de subetapas caracterizada por el esquema multiresolución utilizado. Previo a la registración de la imagen original se usan versiones submuestreadas y filtradas para ir aproximándose a la solución final.

Otra forma en que el usuario puede intervenir en el procesamiento es en la utilización de máscaras. Para este problema en particular lo que se busca es la localización de los ganglios basales. Sin embargo, la registración deformable de dos imágenes de MRI también desplazará vóxeles del ojo (por ejemplo). Esto alterará el valor de la métrica y la registración puede estar guiada por zonas que no son de interés. Las máscaras alivian este problema permitiendo especificar qué región de la imagen utilizar para calcular el valor de la métrica. Más adelante se explicará el uso que se le dieron a las máscaras en esta situación.

Por último, una dificultad que presenta la registración deformable de dos imágenes, en particular las de MRI, es el costo computacional que requiere. Esto también va a influir en las decisiones que se llevan a cabo para el procedimiento utilizado ya que limita las posibilidades disponibles. Por ejemplo, como se detallará más adelante, para la registración deformable se realizan los cálculos con precisión *float* en vez de *double*. Además, debido al tiempo que lleva (aproximadamente 3 horas con la computadora utilizada), analizar las variaciones de los resultados frente a pequeñas alteraciones en los parámetros de entrada no es sencillo.

2.5. Implementación en ANTs

Como se mencionó antes, en este PFC se utilizó ANTs para realizar las registraciones. A continuación se describen los parámetros utilizados para realizar una registración de una imagen de un paciente al espacio MNI.

2.5.1. Primer paso: registración de similitud

```
1 antsRegistration \
2         --dimensionality 3 \
```

Se especifica que es una imagen de tres dimensiones.

```
3         --output $OUTDIR/1_rigid \
```

En la variable `$OUTDIR` se encuentra el directorio dónde se van a guardar los resultados.

```
4         --winsorize-image-intensities [0.005,0.995] \
```

Con esta opción descarta se descartan los vóxeles cuya intensidad de gris no esté dentro del rango especificado. Ayuda ya que vóxeles con un valor alto pueden tener un mal impacto en la registración [24].

```
5         --use-histogram-matching $HIST_MATCH \
```

La variable `$HIST_MATCH` es 1 o 0 dependiendo si se trata de imágenes de la misma modalidad o no. Por ejemplo, el atlas de Milichenko solo tiene modalidad T2. En caso de que se quiera registrar con una imagen T1 de un paciente, `$HIST_MATCH` se define como 0.

Cuando se registra T2 a T2 o T1 a T1 se pone 1 y entonces se lleva a cabo un paso de preprocesamiento de matcheo de histogramas.

```
6         --initial-moving-transform [$ATLAS,$IMAGE,1] \
```

En `$ATLAS` y `$IMAGE` están la ubicación de la imagen del atlas y del paciente respectivamente. La opción 1 para inicializar la registración indica que se traslada la imagen móvil para juntar los centros de masa (según los valores de intensidad). En este caso se especifica al atlas como imagen fija y a la del paciente como móvil. Esta decisión se basa en que la imagen del atlas es de mayor resolución. De lo contrario, al interpolar se perdería esta mayor resolución que tiene el atlas.

```
7         --transform Similarity[0.1] \
```

En este primer paso se realiza una registración de similitud. Entre corchetes se indica el paso de gradiente del optimizador. Esta transformación le agrega un parámetro más a la rígida: el escalado isotrópico. Se notó que para imágenes de cerebro más pequeños que el atlas, la registración rígida no converge de forma esperada. Partiendo de esta transformación, luego la afín distorsiona la imagen aprovechando mayores grados de libertad. Agregando la capacidad de escalar la imagen por un mismo factor en todas las direcciones se logró una mejor primer aproximación.

```
8         --metric MI[$ATLAS,$IMAGE,1,32,Random,0.25] \
```

Se utiliza información mutua como métrica con 32 *bins* en el histograma. No se utilizan todos lo vóxeles para realizar el cálculo, sino que se toman muestras de forma aleatoria tomando el 25 % de los mismos.

```
9         --convergence [1000x500x250x100,1e-5,10] \
10        --shrink-factors 8x4x2x1 \
11        --smoothing-sigmas 3x2x1x0vox \
```

Por un lado se determina que el algoritmo pare si en las últimas 10 iteraciones el valor de la métrica no se mejoró en 10^{-5}. Además se especifica la pirámide utilizada: se realizan 4 pasos y se determinan el número máximo de iteraciones, el factor por el cual se achica la imagen (*downsampling*) y el valor sigma de la gaussiana aplicada a la imagen para cada paso.

```
12        --verbose 1 \
13        > $OUTDIR/1_rigid0utput.txt
```

Finalmente, se especifica que se guarde la salida del procesamiento para luego poder ver cómo se llevó a cabo la registración y analizar errores en caso de haberlos.

2.5.2. Segundo paso: registración afín

```
14 antsRegistration \
15        --dimensionality 3 \
16        --output $OUTDIR/2_affine \
17        --winsorize-image-intensities [0.005,0.995] \
18        --use-histogram-matching $HIST_MATCH \
19        --initial-moving-transform $OUTDIR/1_rigid0GenericAffine.mat \
20        --transform Affine[0.1] \
21        --metric MI[$ATLAS,$IMAGE,1,32,Random,0.25] \
22        --convergence [1000x500x250x100,1e-5,10] \
```

```
23          --shrink-factors 8x4x2x1 \
24          --smoothing-sigmas 3x2x1x0vox \
25          --masks $BRAIN_MASK \
26          --verbose 1 \
27          > $OUTDIR/2_affineOutput.txt
```

Se puede ver que es similar al paso anterior, excepto que se determina que la transformación es afín (línea 20) y que se parte de la transformación anterior (línea 19). Además, se utiliza un máscara del cerebro (proporcionada por los atlas) para limitar que la métrica solo utilice vóxeles dentro de la misma (línea 25).

2.5.3. Tercer paso: registración deformable

```
28 antsRegistration \
29          --dimensionality 3 \
30          --float 1 \
31          --output $OUTDIR/3_syn \
32          --winsorize-image-intensities [0.005,0.995] \
33          --use-histogram-matching $HIST_MATCH \
34          --initial-moving-transform $OUTDIR/2_affine0GenericAffine.mat \
```

Hasta aquí los parámetros ingresados ya fueron explicados anteriormente. Lo que cambia ahora es el uso de precisión *float* en vez de *double* para reducir el costo computacional (línea 30).

```
35          --transform SyN[0.1,3,0] \
```

La transformación deformable utilizada es la SyN (*symmetric normalization*), la cual se basa en difeomorfismos. Estos son una función de un espacio topológico (estructura matemática) a otro que es diferenciable y cuya inversa es diferenciable. En [25] se explica la aplicación para la registración en cuestión.

El parámetro 0.1 es el paso del gradiente y los otros dos especifican penalidades (una por la deformación en cada iteración y otra por la deformación acumulada). Estas penalidades toman en cuenta cuánto son deformados los vóxeles vecinos para restringir la deformación del vóxel en cuestión. Los valores fueron elegidos luego de realizar distintas pruebas y también basándose en estudios realizados [26].

```
36          --metric CC[$ATLAS,$IMAGE,1,4] \
```

También en [25] sugiere la utilización de correlación cruzada como métrica. Para el cálculo de la misma, en vez de utilizar $x_i \in \Omega_F$ como se explicó en las Ecuaciones 2.5 y 2.6, en este caso se toma un espacio reducido de la vecindad de cada vóxel. Aquí, la vecindad es 4 y se optimiza de forma local para cada dominio.

```
37          --convergence [1000x500x250,1e-5,10] \
38          --shrink-factors 8x4x2 \
39          --smoothing-sigmas 3x2x1vox \
```

Como se vio antes, se establece la pirámide de la registración. Para este caso hay que destacar que el último factor por el cual se achica la imagen es 2. Este se debe a la necesidad de reducir la cantidad de memoria que consume el cómputo de la transformación deformable.

```
40          --masks $SUB_MASK \
```

La máscara utilizada en este caso para focalizar la registración es una máscara subcortical, definida en el espacio MNI que comprende específicamente la zona cerebral donde se encuentran los núcleos a estimular [27].

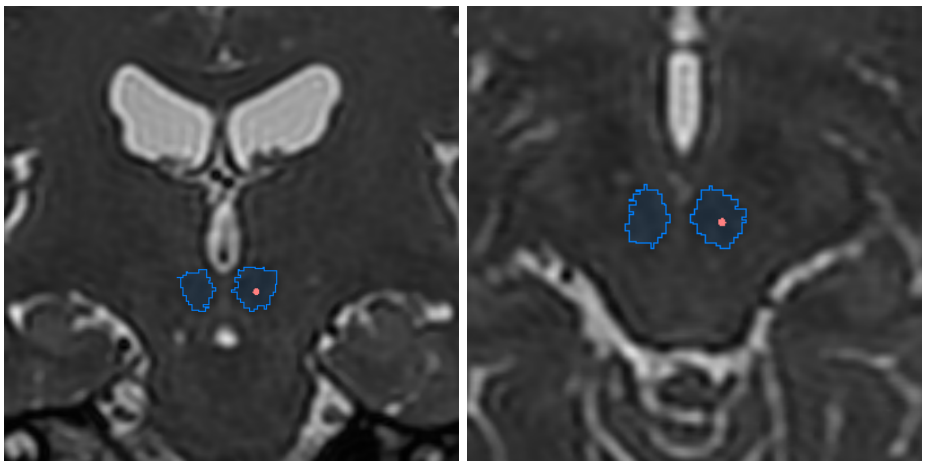
```
41          --verbose 1 \
42          > $OUTDIR/3_synOutput.txt
```

Finalmente, se guarda la salida para posterior evaluación.

2.6. Resultados preliminares

Una vez realizada la registración se obtienen las transformaciones afín (incluye todos los pasos previos a la deformable), deformable y también la inversa de la deformable. Puesto que se utilizó la imagen del atlas como imagen fija, para obtener la segmentación de las estructuras del paciente, las etiquetas se transforman aplicando la transformada deformable inversa y luego la inversa de la afín. De esta forma se proyectan las etiquetas del atlas al espacio de la imagen del paciente (Figura 1.4: Diagrama general del software).

En la Figuras 2.5 y 2.6 se muestra la segmentación de distintas estructuras obtenidas para la imagen de un paciente. También se incluyen fiduciarios que indican la ubicación de ciertos puntos que el médico localizó de forma directa previamente.

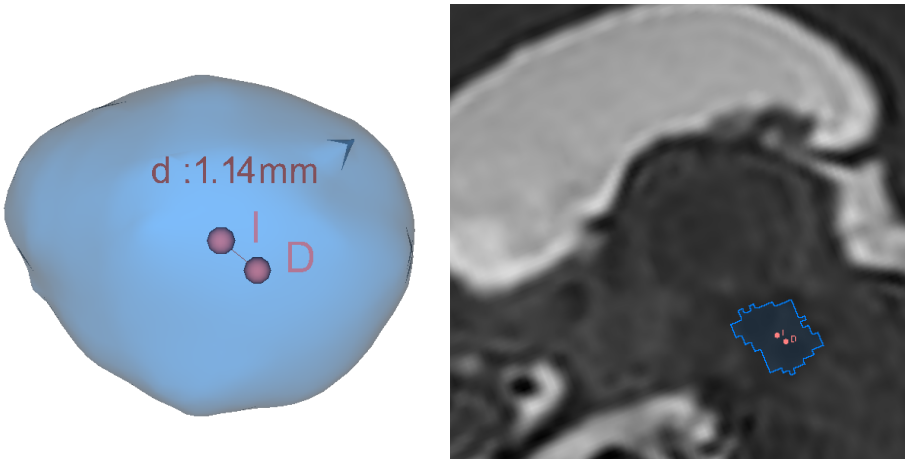


(a) Corte coronal. (b) Corte axial.

Figura 2.5: Segmentación del RN de un paciente obtenido por registración deformable. El fiduciario indica el centro geométrico del RN identificado por el médico de forma directa.

Para obtener una medida cuantitativa, en el caso del RN es posible calcular el centro geométrico de la segmentación y compararlo con la localización directa. La distancia entre el centro obtenido de forma indirecta por registración y el obtenido de forma directa por parte del médico es de $1,14mm$. En la Figura 2.7 se puede ver este resultado.

Figura 2.6: Segmentación del STN de un paciente obtenido por registración deformable. El fiduciario indica el punto más externo del STN identificado por el médico de forma directa.



(a) Representación tridimensional. (b) Corte sagital.

Figura 2.7: Distancia entre centros del RN obtenidos de forma indirecta (I) por registración y directa (D).

3 | Reconstrucción de los electrodos

En el presente capítulo describiremos el proceso utilizado para realizar la reconstrucción de los electrodos a partir de la imagen de CT. Las entradas de este proceso son la imagen de CT, la máscara cerebral del atlas y el modelo de electrodo usado. Este último sirve para tener información acerca del diámetro, la cantidad de contactos y el espaciamiento entre los mismos.

La mayor parte de este método está basado en el estudio de Husch [28] donde se describe una implementación de un software escrito en Matlab para la reconstrucción de electrodos. En el caso de este PFC se utilizaron módulos del 3DSlicer y Python para facilitar la integración de este procesamiento al resto del software.

3.1. Obtención de la máscara cerebral

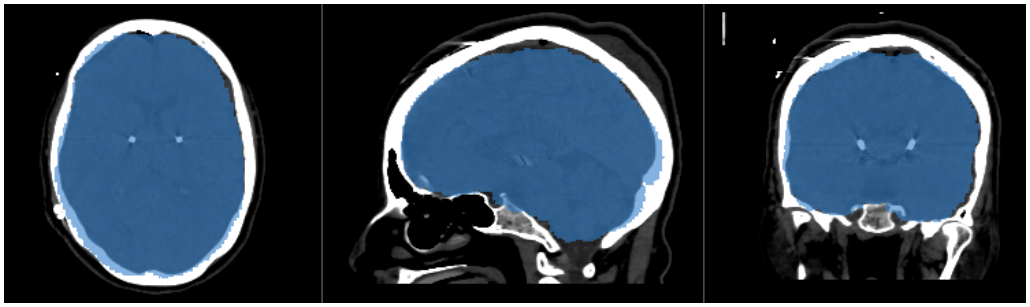
La primera etapa del algoritmo busca obtener una segmentación de la imagen donde solo esté incluida la región cerebral (sin cráneo ni fondo). Los atlas cerebrales descriptos en el capítulo anterior por lo general poseen una máscara de segmentación cerebral. En el flujo de uso normal del software descripto en nuestro trabajo, primero se realiza la registración de la imagen preoperatoria con el atlas, y luego la reconstrucción de los electrodos en base a la imagen postoperatoria. Por lo tanto es posible proyectar sobre el paciente la máscara de segmentación cerebral del atlas (Figura 1.4: Diagrama general del software). De esa forma se obtiene la región cerebral como se muestra en la Figura 3.1a.

Luego, para asegurarse de que no haya cráneo en la misma, se pasó a hacer una operación de erosión de morfología matemática. Se utilizó una implementada en *SimpleITK*, accedida mediante 3DSlicer. Para determinar el radio del elemento estructurante se tuvo en cuenta el espaciamiento que tiene la imagen en sus tres direcciones. Así se obtuvo la salida que se ve en la Figura 3.1b.

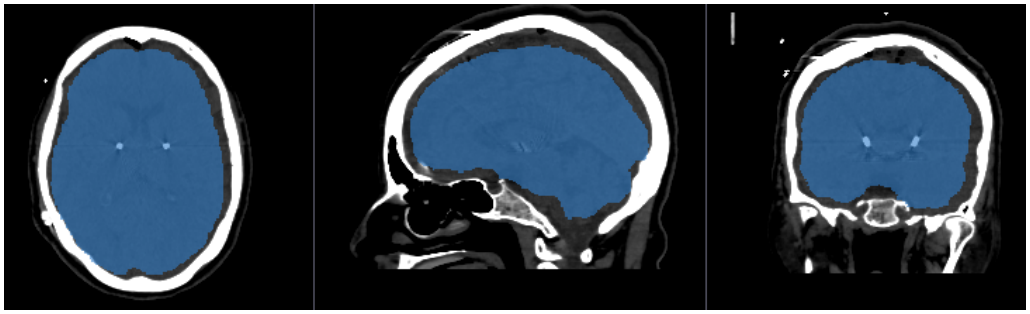
Finalmente, esta segmentación es la que se usa como máscara para recortar la imagen original y quedarse solamente con la región que la misma delimita. Se puede ver que los electrodos quedan dentro de la máscara mientras que no queda cráneo en la misma. Esto es muy importante para los siguientes pasos ya que se usan filtros que tienen en cuenta las intensidades de los vóxeles y tanto el cráneo como los electrodos son los que más intensidad poseen. Se puede ver la imagen luego de aplicarle la máscara en la Figura 3.1c.

3.2. Segmentación de los electrodos

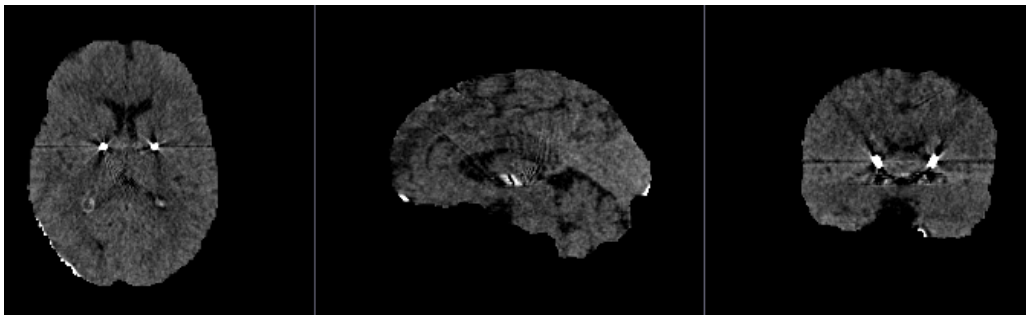
Partiendo de la imagen anterior, lo que se busca a continuación es obtener una etiqueta para cada electrodo. Se puede observar que por medio de un umbral se podría obtener



(a) Máscara obtenida al transformar la máscara del atlas.



(b) Salida luego de erosionar la máscara anterior.



(c) Salida luego de cortar la imagen de CT con la máscara.

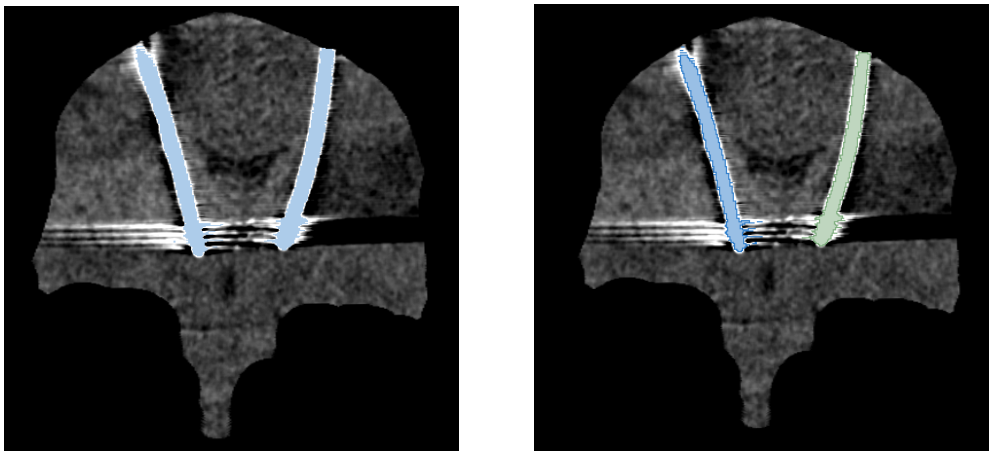
Figura 3.1: Obtención de la máscara cerebral. La máscara está en azul con un 50 % de opacidad.

una segmentación inicial aceptable ya que la intensidad de los electrodos es mayor que la del cerebro (ver Figura 3.1c). La mejor manera para determinar un umbral es hacerlo a mano pero como se quiere un algoritmo automático se utilizó otro método. Luego de probar con distintos filtros se eligió el de *Maximum Connected Components* disponible en *SimpleITK*. Este determina un valor de umbral para la imagen buscando maximizar el número de objetos en la imagen que son mayores a un tamaño mínimo dado. El tamaño mínimo especificado en este caso se determinó a partir del conocimiento real del tamaño del electrodo, del espaciado de las imágenes y a partir de los resultados que se fueron obteniendo. En la Figura 3.2a se puede apreciar el resultado.

Esta segmentación tiene solamente una etiqueta. Para obtener una etiqueta diferente para cada electrodo se utilizo el filtro *Islands* del módulo de segmentación de 3DSlicer tomando como operación *Split Islands to Segments*. Lo que hace esto es separar cada conjunto de vóxeles conexo como segmentos. Además permite descartar segmentos menores

a un determinado tamaño, lo cual se utilizó para eliminarlos en caso de que se encontrasen en la imagen segmentada.

Otro método de control que se tomó para quedarnos solamente con dos etiquetas (una para cada electrodo) fue analizar cada segmento obtenido y dejar los dos que presenten los vóxeles de mayor intensidad. Esto se realizó por medio del módulo de *Label Statistics* de 3DSlicer y el resultado se puede ver en la Figura 3.2b



(a) Salida luego del filtro *Maximum Connected Components*.

(b) Salida luego de separar los conjuntos de vóxeles conexos y seleccionar los de mayor intensidad con un tamaño mínimo dado.

Figura 3.2: Primer parte del método de segmentación de los electrodos. Aclaración: el plano de la imagen está rotado para mejorar la visualización.

Finalmente, la última operación que se le realizó a esta segmentación fue la de apertura de morfología matemática. Lo que se busca por medio de este método es suavizar cada etiqueta y obtener una forma más parecida a la del electrodo. La segmentación obtenida luego se puede ver en la Figura 3.3 (solo se muestra el resultado para un electrodo).

3.3. Dirección del electrodo

En este algoritmo los $25mm$ del extremo inferior de los electrodos se consideran como rectos. Se busca entonces esa dirección para luego hallar los contactos en la misma.

Lo primero que se realiza es un esqueleto de la segmentación de los electrodos. Esto se logra mediante el filtro de adelgazamiento *Binnary Thinning Image Filter* de *SimpleITK*. Luego se toman los puntos inferiores del esqueleto y se los aproxima a una recta minimizando la distancia entre esta y los puntos. Esta recta ajustada a la nube de puntos es la que se tomará como dirección de los electrodos. La misma se puede ver en la Figura 3.4.

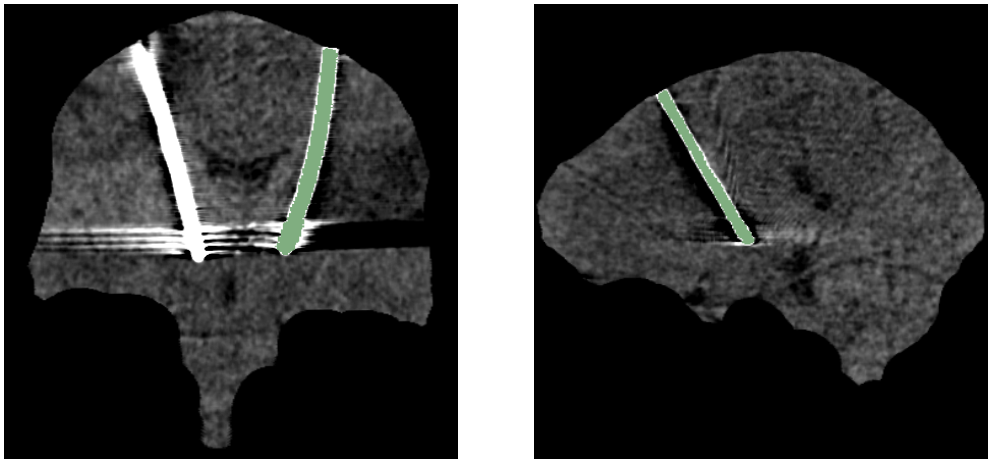
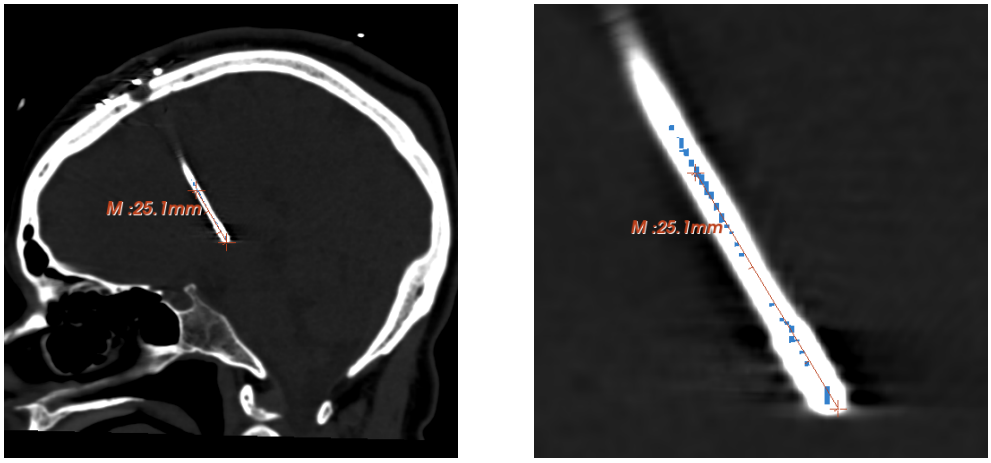


Figura 3.3: Segunda parte del método de segmentación de los electrodos: apertura. Aclaración: los planos de la imagen están rotados para mejorar la visualización.



(a) Vista global. (b) Vista de detalle.

Figura 3.4: Dirección del electrodo. En la Figura se puede ver el esqueleto de la segmentación en azul y una regla indicando la dirección obtenida.

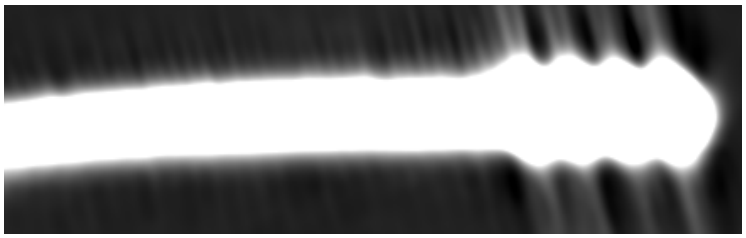
3.4. Imagen del electrodo

Una vez obtenida esta recta, se obtiene la transformación (rígida) que la lleva a estar alineada al eje anteroposterior. Esta transformación también se le aplica a la imagen (una vez para cada electrodo). Esto se realiza para facilitar los siguientes pasos, es más sencillo trabajar utilizando coordenadas cartesianas sobre un eje del sistema cartesiano que sobre una recta arbitraria en el espacio.

A continuación se crea una región de interés sobre el electrodo y se la utiliza como entrada al módulo *Crop Volume* de 3DSlicer. A este método también le ingresa un factor para escalar el espaciado de la imagen a otro de mayor resolución utilizando un interpolador BSpline. La razón por la cual se desea tener una imagen del electrodo con mayor resolución es por las dimensiones que tienen los mismos. Por ejemplo, los contactos pueden tener

dimensiones de 1,5mm con una separación de 0,5mm entre ellos, esto está en el mismo orden de magnitud que el tamaño de vóxeles de la imagen. En la Figura 3.5 se puede ver la imagen del electrodo obtenida.

(a) W=26371, L=13722.



(b) W=1000, L=400.

Figura 3.5: Imagen del electrodo obtenida con distintos ajustes W/L.

3.5. Ubicación de los contactos

La imagen obtenida en el paso anterior es la que se utiliza para hallar la posición de los contactos. Primero se obtiene el promedio de intensidad de gris de cada corte de la imagen (tomando planos perpendiculares a la dirección del electrodo). Esto se puede ver en la Figura 3.6.

Luego se buscan los máximos de intensidad que tienen estos datos. Esto se realizó por medio de la función *find_peaks* del módulo *scipy.signal* de Python. También se ingresó a la función un mínimo espaciado entre los picos para que no se detecten contactos más cercanos que lo real. Tomando la cantidad de máximos como cantidad de contactos se obtiene una primera estimación de las posiciones de los mismos.

El siguiente paso a realizar es el de correlacionar esta posición con el espaciado real que tienen los contactos. De esta forma se obtiene en qué posición de la dirección del electrodo están sus centros. El resultado se puede ver en la Figura 3.7.

3.6. Modelo del electrodo

Finalmente se realizó un modelo de cada electrodo usando cilindros de la biblioteca VTK e importándolos como segmentaciones dentro de 3DSlicer. Para los contactos, el centro del cilindro se tomó como centro de los contactos obtenidos en el paso anterior y como diámetro el diámetro real del electrodo. Luego se hizo una representación del electrodo con una longitud de 25mm (parte del mismo que se estimó como una recta). El resultado se puede ver en las Figuras 3.8 y 3.9.

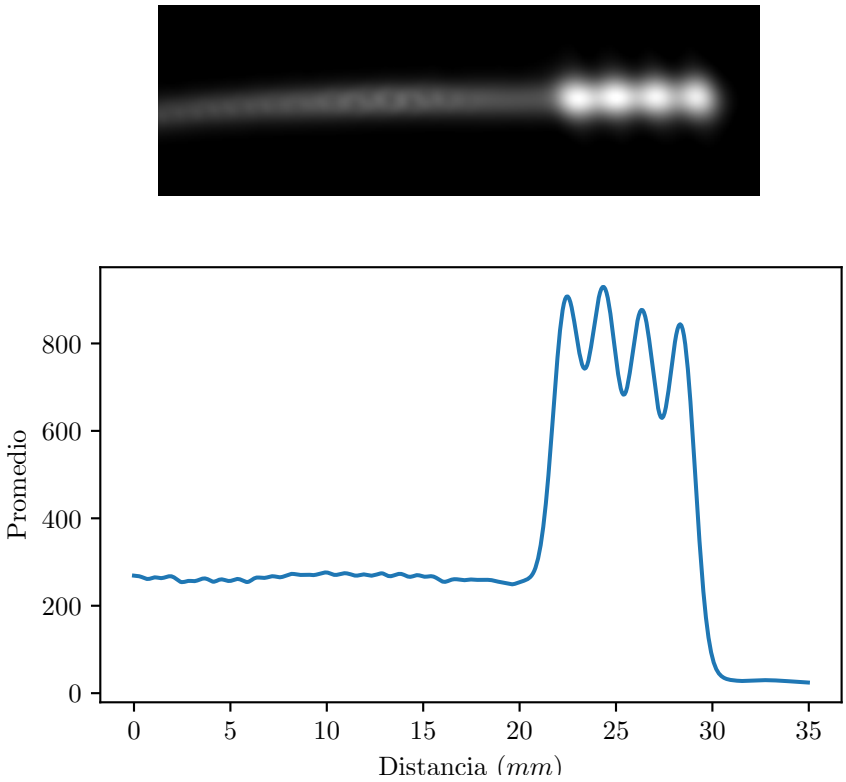


Figura 3.6: Electrodo y promedio de intensidad de gris de cada *slice*.

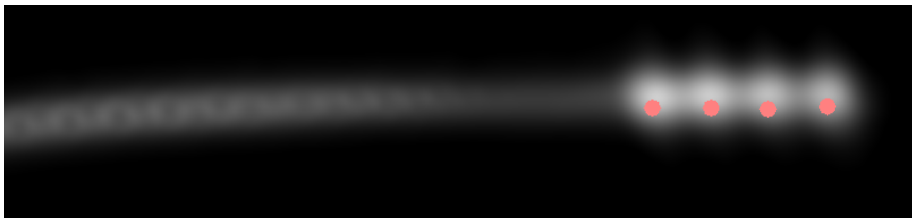


Figura 3.7: Ubicación de los contactos obtenida.

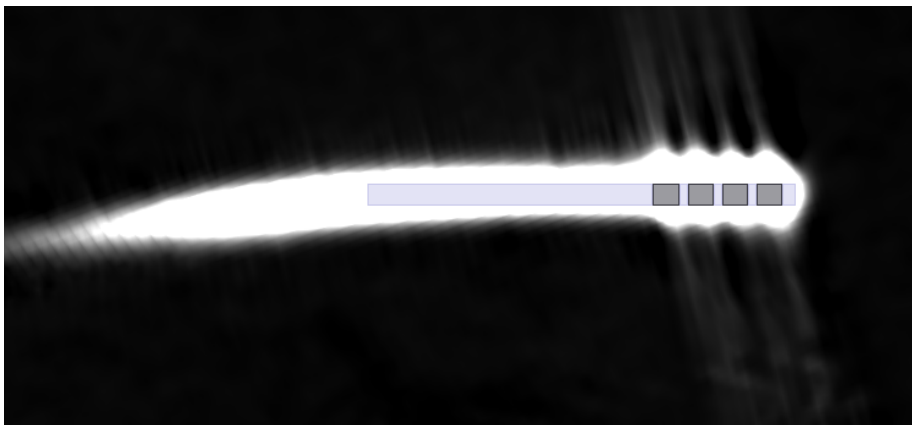
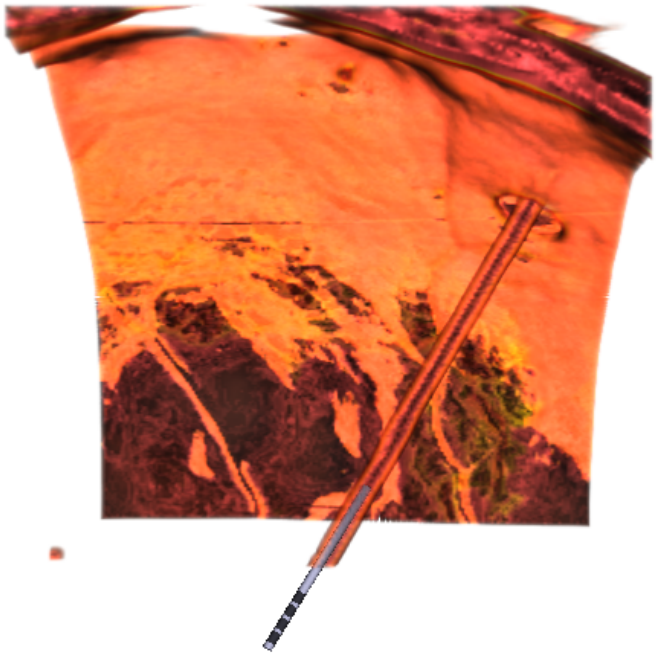


Figura 3.8: Electrodo reconstruido con imagen de CT.

(a) Vista anterior.



(b) Vista lateral derecha.

Figura 3.9: Electrodo reconstruido con render de CT.

4 | Interfaz de usuario

En base a lo planteado en los objetivos, la idea de este software es que pueda ser utilizado por un médico sin mayores dificultades. Para ello se diseñó una interfaz de usuario que integra los distintos algoritmos descriptos y que ofrece opciones de visualización que permiten evaluar y representar los resultados de diferentes maneras.

Puesto que por medio de Python se puede programar en 3DSlicer y se tiene acceso a sus librerías, incluyendo Qt, se puede prescindir de la ventana principal del 3DSlicer y programar una interfaz de usuario usando los componentes que uno desee. Así se puede desarrollar algo pensado directamente para la aplicación que tendrá y de esta forma solo incluir lo que sea necesario.

La aplicación está diseñada para que la misma guíe al usuario a través de una serie de pasos, siguiendo un flujo de trabajo. Primero se selecciona el paciente a analizar, luego se cargan las imágenes y se establecen los parámetros de procesamiento. Finalmente se muestran los resultados por medio de distintas opciones de visualización. Estos pasos se describirán detalladamente en las siguientes secciones. Se eligió este formato, comunmente conocido como *wizard* o asistente, para simplificar la interacción. Además, antes de ingresar al siguiente paso, el software realiza una verificación de que las parámetros ingresados por el usuario en el paso previo sean correctos. Así se evitan errores en el procesamiento y se logra un uso intuitivo de la aplicación.

En la Figura 4.1 se puede ver una captura de pantalla de la interfaz cuando se ejecuta el programa. A continuación se describen los componentes identificados por las letras en la imagen.

A Se incluye el indicador "No apto para uso clínico" ya que este software no se encuentra registrado en el ANMAT. Su uso es exclusivamente de investigación.

B El visualizador estándar de 3DSlicer, en el cual se puede navegar por los distintos cortes de la imagen (axial, sagital y coronal). También cuenta con un cuadro para visualizaciones en tres dimensiones. Más adelante se detallará el uso del mismo.

C Indica el estado de la aplicación. Se encuentra "Listo" cuando el usuario puede interactuar con la aplicación y "Cargando..." cuando se está realizando algún tipo de procesamiento. En este último caso también cambia el formato del cursor, señalando que no se puede interactuar con el programa.

D Por medio de este menú se selecciona la configuración de visualización de **B**. Las opciones disponibles son los tres cortes y 3D (por defecto), solo de cortes o solo 3D.

E Este botón permite bloquear o desbloquear la modificación del contraste de la imagen cuando se selecciona y desplaza el cursor en las vistas de corte.

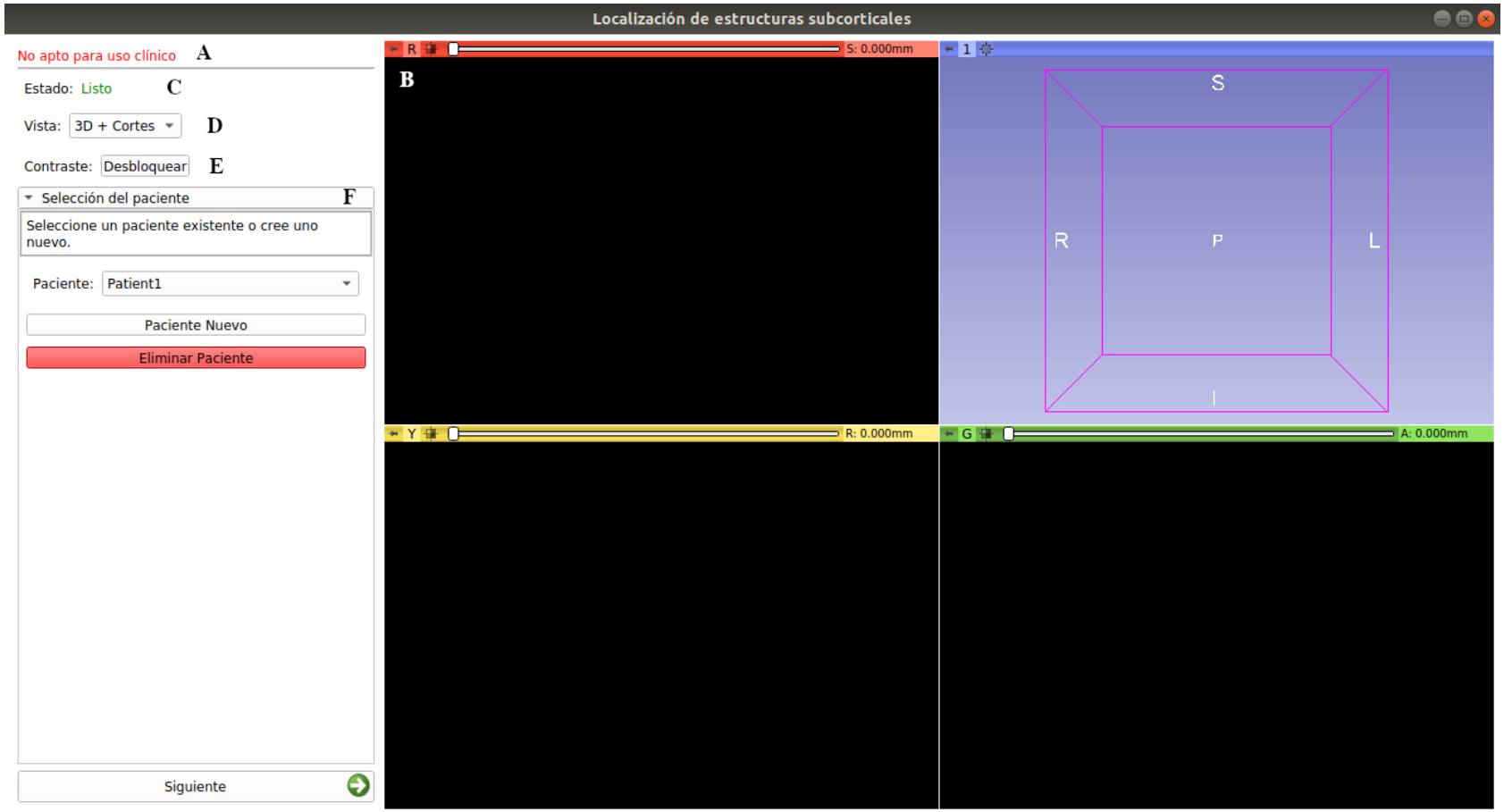


Figura 4.1: Interfaz de usuario.

F Aquí se muestra el paso en que se encuentra. Cuenta con un título, una descripción y las opciones correspondientes. Abajo se puede seleccionar para ingresar al siguiente paso o retroceder al anterior.

Todos los componentes de la interfaz cuentan con un *tooltip* que indica su función. Así, al situar el cursor sobre los mismos se tiene una ayuda en caso de no saber cuál es su uso.

4.1. Selección del paciente

El primer paso es seleccionar el paciente a analizar. Se puede elegir entre pacientes ya creados, crear un nuevo paciente o eliminar el paciente seccionado. El software tiene especificado un directorio donde cada carpeta es un paciente distinto. En la lista de pacientes disponibles las opciones son los nombres de estas carpetas.

En caso de crear un paciente nuevo aparecerá una ventana pidiendo al usuario que ingrese el nombre del paciente (solo se aceptan como entrada caracteres alfanuméricos). Al crear un nuevo paciente se crea una nueva carpeta en el directorio. En caso de eliminar el paciente aparecerá una ventana pidiendo confirmación de la acción ya que no se pueden recuperar los archivos eliminados. Esto hará que se elimine la carpeta correspondiente. Haciendo click en *Siguiente* se procede al paso posterior.

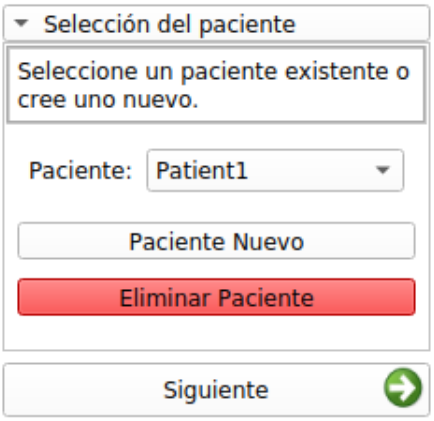


Figura 4.2: Paso 1: Selección del paciente

4.2. Cargar imágenes

A continuación se pide al usuario que seleccione las carpetas donde se encuentran las series DICOM que se desean analizar. Se eligió el formato DICOM ya que es un estándar en imágenes médicas. Permite almacenar tanto imágenes como datos del paciente, centro de salud y modalidad de imagen entre otras cosas. Todos los equipos y sistemas de transmisión de imágenes médicas y hospitales utilizan el estándar DICOM.

Se debe especificar también la modalidad de la resonancia. Esto va a intervenir luego en el procesamiento para definir si se utiliza o no matcheo de histogramas (ver Sección 2.5.1). Al cargar las imágenes por primera vez se guarda una copia de las mismas en formato

`.nii.gz` en la carpeta del paciente. Si es un paciente con el cual ya se había trabajado antes se deshabilitan los botones de carga de imágenes ya que se utilizan las previas.

En este paso se puede ver el cuadro de texto con información acerca del paso anterior (Figura 4.3 en verde). Específicamente, se puede ver cuál es el paciente con el que se está trabajando.

En caso de hacer click en *Siguiente* y no haber carpetas DICOM seleccionadas o imágenes disponibles aparecerá una ventana de alerta indicando que no es posible continuar sin imágenes.

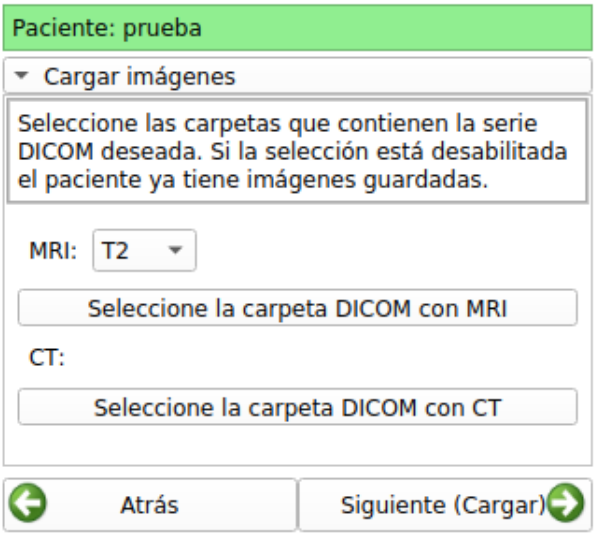


Figura 4.3: Paso 2: Cargar imágenes

4.3. Parámetros de procesamiento

En este paso, luego de cargar las imágenes, se abren ventanas pidiendo al usuario que especifique cuál es el núcleo blanco y cuál es el modelo del electrodo. Estos datos son guardados de tal manera que la próxima vez que se ingrese a este paso con el mismo paciente no se piden.

El usuario debe especificar también el atlas que desea utilizar para realizar el procesamiento. Dentro de las opciones desplegables están los atlas utilizados previamente para este paciente. También se puede utilizar otro haciendo click en *nuevo*. Para este caso, se abre una ventana para determinar qué atlas utilizar.

Al mismo tiempo, en el visualizador es posible navegar las imágenes importadas para confirmar que son las correctas.

En este paso es necesario especificar el atlas a utilizar para poder proceder con el procesamiento. En caso contrario aparecerá una ventana de alerta como se explicó en el paso anterior.

En caso de que ya se haya realizado el procesamiento, al hacer click en *Siguiente (Procesar)*, se cargan los resultados previos. En cambio, si es la primera vez que se utiliza al paciente seleccionado, se llevan a cabo los siguientes pasos:

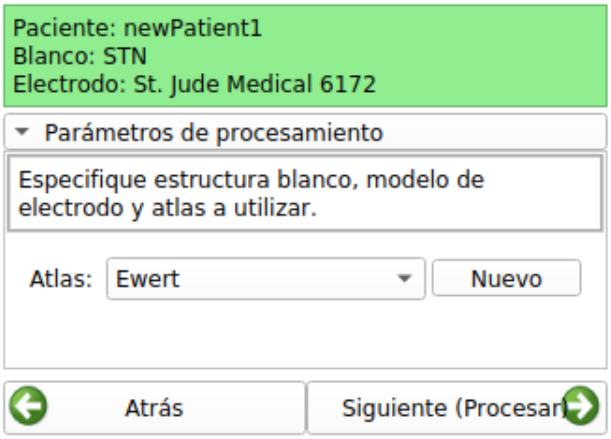


Figura 4.4: Paso 3: Parámetros de procesamiento

1. En el caso de que sea el primer atlas se realiza un preprocesamiento de *Bias Field Correction* y se guarda el resultado. Para realizar esto se utiliza el módulo de 3DSlicer correspondiente.

2. Se realiza la registración como se explicó en la Sección 2.5 con el atlas seleccionado y se cargan las transformaciones resultantes.

3. En caso de que sea el primer atlas se sigue a 4, sino pasa a 6.

4. Se realiza la transformada AC-PC. Para ello primero se transforman las coordenadas de AC, PC y fiduciarios de la línea media del cerebro al espacio del paciente. Luego se calcula una transformación rígida de modo que el segmento AC-PC esté horizontal y la línea media contenida en el plano anterosuperior. Esto se hace por medio del módulo *AC-PC Transform* de 3DSlicer. Luego se traslada la imagen para que el AC esté en el origen de coordenadas. De esta forma se lleva a todos los pacientes al origen del 3DSlicer y se facilita la visualización. Además, ver la imagen en esta orientación hace más simple la interpretación de la misma.

5. Se genera la máscara del cerebro del paciente (transformando la máscara del atlas) y se la guarda.

6. En caso de que haya una imagen de CT cargada se pasa a 7, sino termina.

7. Se registra la imagen de CT a la MRI utilizando una transformación rígida. Se utilizan parámetros similares a los utilizados en 2.5.1, sin hacer matcheo de histogramas.

8. Se realiza la reconstrucción del electrodo como se explicó en la Sección 3.

4.4. Visualización

Este paso está pensado para que el usuario evalúe los resultados del procesamiento. Como se ve en la Figura 4.5 se tiene la posibilidad de elegir entre distintas configuraciones

de visualización para analizar los resultados obtenidos. Por medio del primer *slider* el usuario puede deslizarse entre la imagen en *Background* y la imagen en *Foreground*. El segundo *slider* especifica la opacidad de las etiquetas. Abajo de este último se puede ver el nombre de la etiqueta cuando se sitúa el cursor sobre la misma. A continuación se describen las distintas opciones de visualización disponibles y en el Anexo A las Figuras correspondientes.

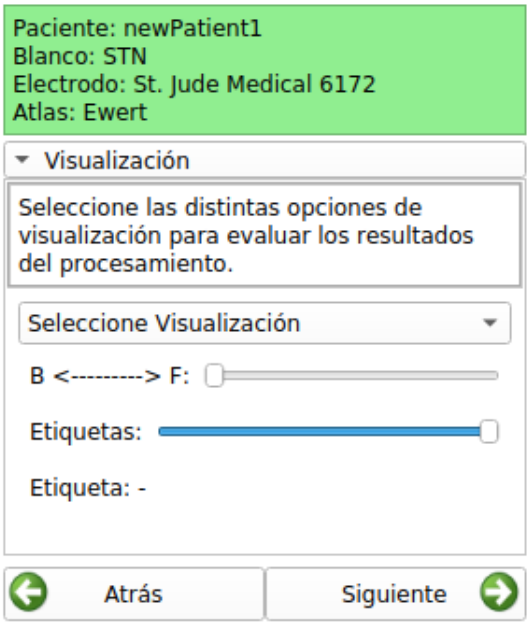


Figura 4.5: Paso 4: Visualización.

MRI afín y MRI deformable

Esta opción permite ver en *Background* la imagen del paciente transformada con la transformación afín y en *Foreground* con la transformación deformable. Además se puede visualizar la transformación deformable por medio de las flechas que indican en qué dirección se desplazaron los vóxeles. En la Figura A.1 se puede ver un ejemplo de esta visualización.

MRI deformable y atlas

En esta opción de visualización se pone como *Background* la imagen del paciente deformada y en el *Foreground* el atlas. Se puede ver el resultado de la registración y superponer las etiquetas en el espacio del atlas. Ver Figura A.2.

MRI y etiquetas

Para este caso las etiquetas del atlas son transformadas para llevarlas al espacio del paciente. De esta forma se puede visualizar la imagen del paciente original (a la cual se le

realizó la transformación AC-PC) y la segmentación de sus estructuras. En la Figura A.3 se puede ver esta opción para distintos cortes.

CT y contactos

Esta opción permite visualizar la imagen de tomografía y la ubicación de los contactos obtenida. A estos se los representa en 3DSlicer por medio de fiduciarios. Se puede ver un ejemplo en la Figura A.4.

CT render y reconstrucción del electrodo

En este caso se realiza un render de la tomografía utilizando el módulo de *Volume Rendering* de 3DSlicer. Superpuesto al render se coloca la reconstrucción de los electrodos en el visor 3D. También se presenta una región de interés que el usuario puede editar para seleccionar que parte desea visualizar. Por medio de esta visualización y la anterior se puede determinar que tan fiel es el procesamiento realizado para la reconstrucción. Ver Figura A.5.

MRI y CT

Para este último caso se ubica en *Background* la imagen de MRI y en *Foreground* la imagen de CT. A ambas imágenes se les cambia el color para que sea más simple diferenciarlas. Esta opción permite evaluar la registración rígida entre estas imágenes. En la Figura A.6 se pueden ver distintos cortes sagitales para esta visualización.

4.5. Visualización final

Este es el último paso en el cual se combinan los resultados de la segmentación de las estructuras subcorticales y la reconstrucción de los electrodos. Se obtienen modelos 3D de las estructuras y se pasa al visor en tres dimensiones para ver dichas estructuras y los electrodos. En este paso también se mide la distancia entre los contactos y el centro del núcleo blanco y se representa la menor medida mediante una regla de 3DSlicer. El usuario puede visualizar o no los electrodos y especificar su opacidad. Lo mismo para cada estructura por separado o todas juntas. También puede poner las vistas de la resonancia y desplazarlas en el espacio. Finalmente, tiene la opción de visualizar el render de CT y acotarlo con una región de interés. En la Figura 4.6 se pueden ver los controles y en la Figura 4.7 se puede ver un ejemplo de esta visualización.

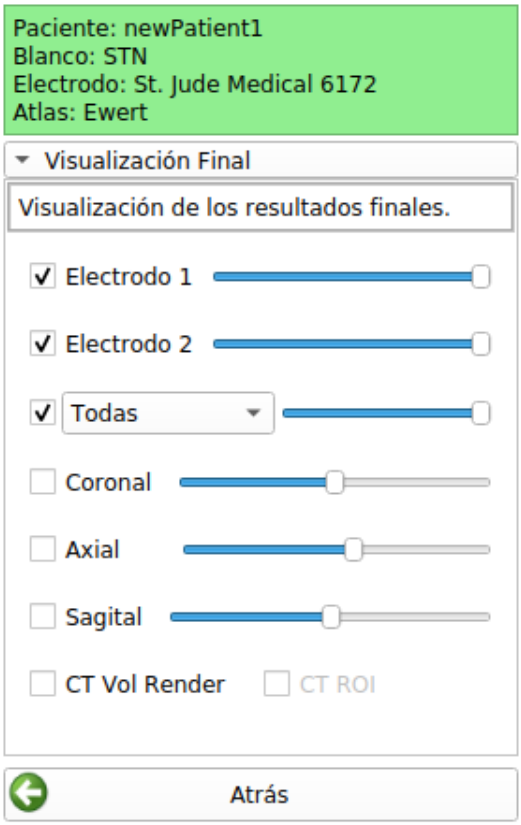


Figura 4.6: Paso 5: Visualización final.

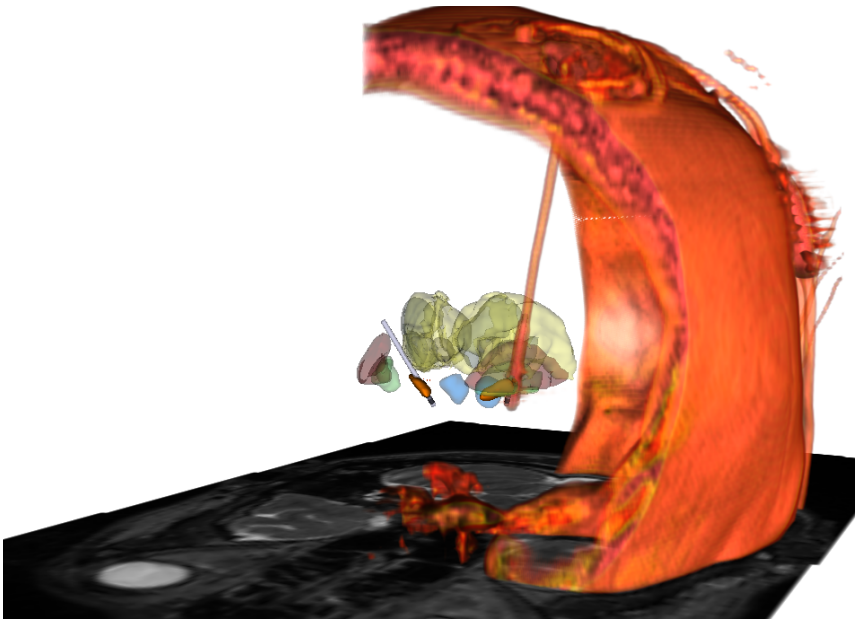


Figura 4.7: Ejemplo de visualización final. Se pueden ver los electrodos reconstruidos, las estructuras segmentadas (en particular el STN con mayor opacidad), el render del CT y un corte de la MRI en vista axial.

5 | Resultados

El análisis de resultados para obtener una medición cuantitativa del desempeño de los procesos y algoritmos descriptos no es una tarea trivial. En lo que respecta a la segmentación de estructuras cerebrales, una forma de evaluarla sería contrastando contra una segmentación que opere como *ground truth*. Sin embargo, dado que se busca segmentar estructuras que no son fácilmente identificables en la MRI, no es posible obtener un *ground truth* del todo certero. Además, para confeccionar el mismo es necesario que se le dedique mucho tiempo por parte de médicos expertos a la segmentación manual de los ganglios basales de cada imagen. Para simplificar esta tarea se analizaron solamente ciertos puntos que el consultor médico identificó de forma directa en una imagen. En la Sección 2.6 se puede ver que para el centro geométrico del RN la distancia entre la localización indirecta y directa es de $1,14mm$. Esta medida da una idea de que para esa imagen en esa zona sí se tiene una registración con un error de poco más de un milimetro. Luego, la validación de la segmentación se realiza mediante la inspección visual por parte de un especialista experimentado en neuroanatomía.

Lo mismo aplica para el procesamiento de reconstrucción de los electrodos. Dado que no se sabe de antemano cuál es la ubicación exacta, no se puede cuantificar el resultado obtenido de forma objetiva. Por eso también el software que se desarrolló tiene una etapa dedicada a la evaluación de los resultados proponiendo distintas visualizaciones.

Finalmente, el resultado último es la respuesta clínica que tiene el paciente frente a la estimulación cerebral. Hacer un estudio de cómo se relaciona la respuesta clínica con las distintas zonas estimuladas excede el alcance de este PFC. Sin embargo puede utilizarse la información acerca de qué electrodos activos lograron la mejor respuesta clínica. El método aplicado entonces consiste en estimar la distancia entre los contactos y el centro del núcleo blanco, verificando si hay coincidencia entre los electrodos activos y los electrodos que nuestro procedimiento estima que son los más cercanos al núcleo.

En las Figuras 5.2, 5.3, 5.4 y 5.5 se presentan los resultados obtenidos para las imágenes disponibles a analizar al día de la fecha. Se procesó además un caso adicional provisto por el grupo que desarrolla Lead-DBS (Figura 5.6). Estas Figuras muestran el resultado obtenido al registrar con el atlas de Ewert debido a que se vio una segmentación más fiel y cuenta con todas las estructuras estimuladas.

Así, se tienen en total 5 casos. En la Tabla 5.1 se detallan algunas características de las imágenes utilizadas. Se puede destacar que ante la variedad de modalidades y espaciamiento de vóxeles, se obtuvieron los resultados por medio de un algoritmo de procesamiento completamente automático.

De 4 de los 5 casos se tiene información de la estimulación (los que fueron presentados por el médico). Para el caso 4, a pesar de que fueron implantados ambos electrodos, solo se estimula con el izquierdo por la condición clínica del paciente.

Por ello se tiene un total de siete electrodos de los cuales se sabe su configuración clínica actual. Uno de ellos (caso 2, lado izquierdo) presenta una estimulación que pareciera contradecirse con la ubicación de los contactos respecto al blanco. Para el resto de los casos se puede ver que los contactos elegidos están relacionados con su cercanía a la zona a estimular.

Caso	Imagen	Modalidad	Espaciamiento (mm)		
			I	J	K
1	MRI	T2	0,500	0,500	0,600
	CT	-	0,586	0,568	1,250
2	MRI	T1	1,071	1,071	1,200
	CT	-	0,448	0,448	0,300
3	MRI	T1	0,488	0,488	1,000
	CT	-	0,586	0,568	0,587
4	MRI	T1	0,469	0,469	1,000
	CT	-	0,586	0,568	1,250
5	MRI	T2	0,651	0,651	2,000
	CT	-	0,415	0,415	0,300

Tabla 5.1: Detalle de las imágenes utilizadas.

Son pocos los casos como para hacer un estudio detallado que relacione la distancia entre los contactos y los centros del blanco. Sin embargo, una primera aproximación se puede ver en la Figura 5.1. Descartando el lado izquierdo para el caso 2, para el blanco STN se tiene una estimulación monopolar con uno de los contactos más cercanos activo. Para los tres casos de estimulación al GPi posterior se puede ver en las respectivas Figuras que se elige una configuración multipolar activando los contactos próximos al más cercano.

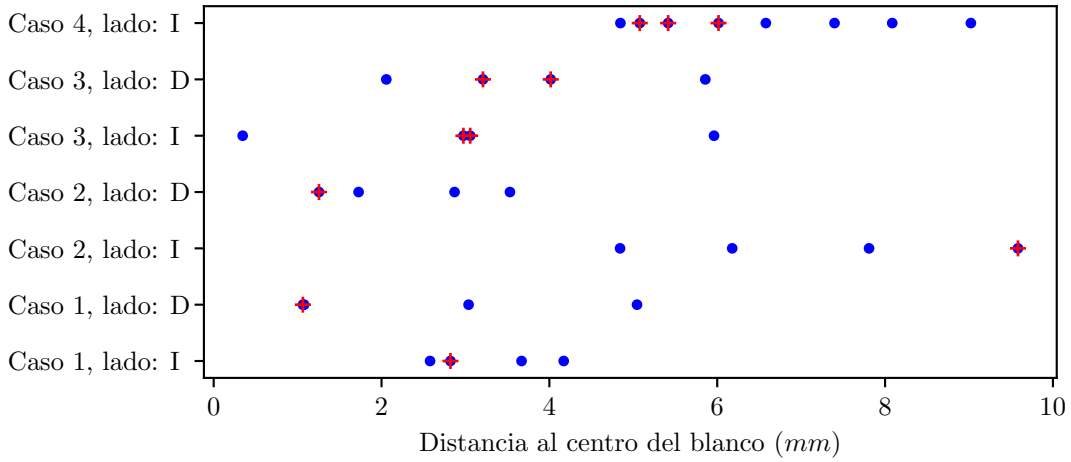
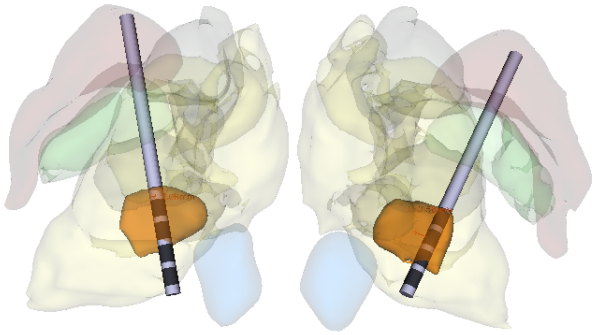


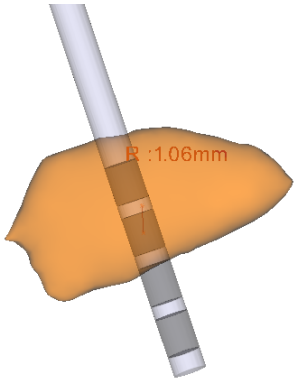
Figura 5.1: Resumen de los casos. Las filas representan cada lado para los casos. Los puntos en azul representan la distancia de cada contacto al centro del blanco. Los contactos activos se encuentran marcados con un signo + rojo.

5.1. Caso 1

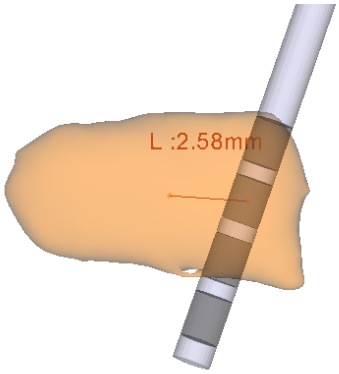
Figura 5.2: Resultados para el caso 1 (blanco: STN). Se puede ver la distancia del contacto más cercano al centro del blanco para cada electrodo y la configuración de estimulación al día de la fecha. El contacto en rojo es utilizado como cátodo.



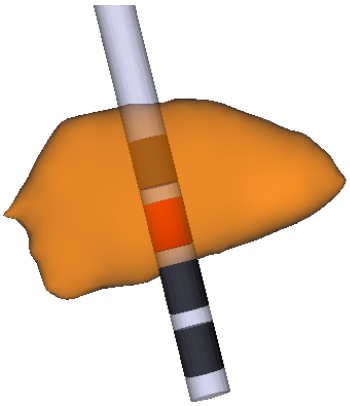
(a) Estructuras y electrodos.



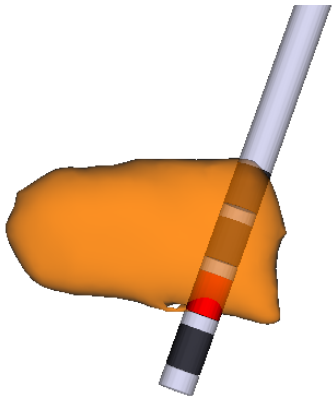
(b) Lado derecho. $d_{min} = 1,06mm$



(c) Lado izquierdo. $d_{min} = 2,58mm$



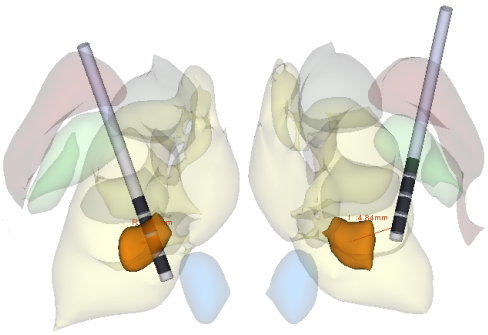
(d) Configuración del lado derecho.



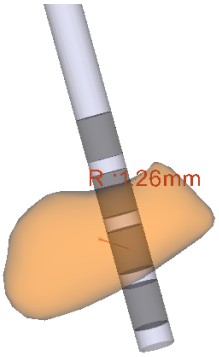
(e) Configuración del lado izquierdo.

5.2. Caso 2

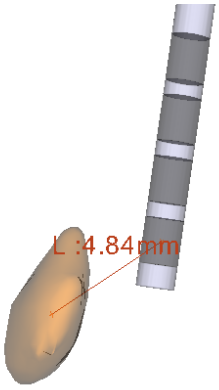
Figura 5.3: Resultados para el caso 2 (blanco: STN). Se puede ver la distancia del contacto más cercano al centro del blanco para cada electrodo y la configuración de estimulación al día de la fecha. El contacto en rojo es utilizado como cátodo.



(a) Estructuras y electrodos.



(b) Lado derecho. $d_{min} = 1,26mm$



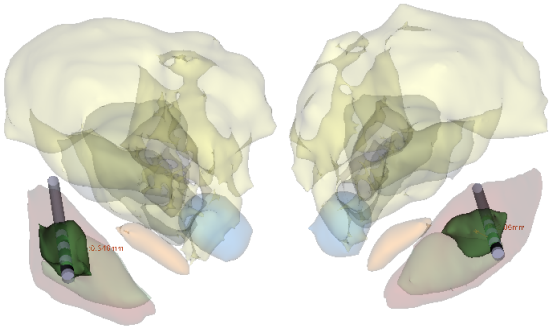
(c) Lado izquierdo. $d_{min} = 4,84mm$

(d) Configuración del lado derecho.

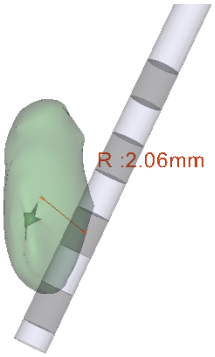
(e) Configuración del lado izquierdo.

5.3. Caso 3

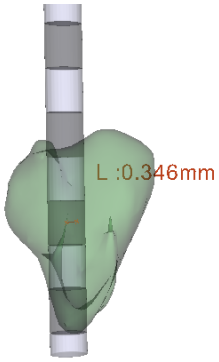
Figura 5.4: Resultados para el caso 3 (blanco: GPi posterior). Se puede ver la distancia del contacto más cercano al centro del blanco para cada electrodo y la configuración de estimulación al día de la fecha. Los contactos en rojo son utilizados como cátodos.



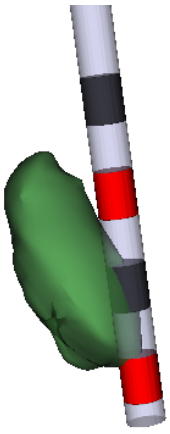
(a) Estructuras y electrodos.



(b) Lado derecho. $d_{min} = 2,06mm$



(c) Lado izquierdo. $d_{min} = 0,35mm$

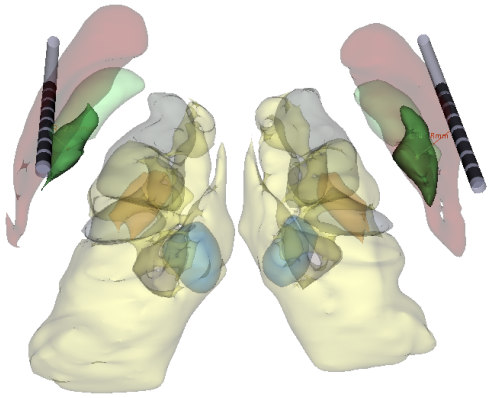


(d) Configuración del lado derecho.

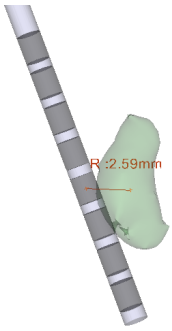
(e) Configuración del lado izquierdo.

5.4. Caso 4

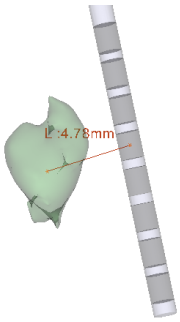
Figura 5.5: Resultados para el caso 4 (blanco: GPi posterior). Se puede ver la distancia del contacto más cercano al centro del blanco para cada electrodo y la configuración de estimulación al día de la fecha. Los contactos en rojo son utilizados como cátodos y el azul como ánodo. Para este caso solo se estimula el lado izquierdo.



(a) Estructuras y electrodos.



(b) Lado derecho. $d_{min} = 2,59mm$

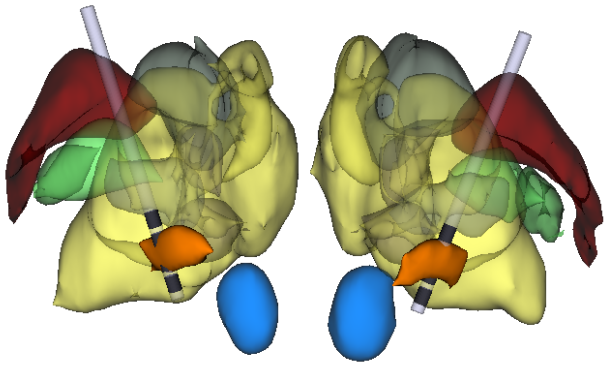


(c) Lado izquierdo. $d_{min} = 4,78mm$

(d) Configuración del lado izquierdo.

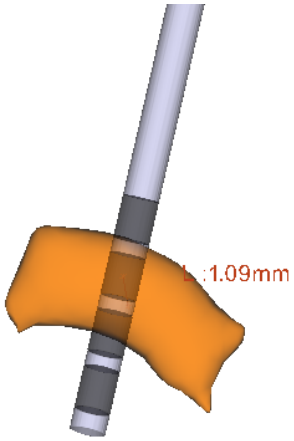
5.5. Caso 5 (adicional)

Figura 5.6: Resultados para el caso adicional provisto por el grupo que desarrolla Lead-DBS (blanco: STN). Se puede ver la distancia del contacto más cercano al centro del blanco para cada electrodo. El paciente está deidentificado y no se tiene la configuración de la estimulación.



(a) Estructuras y electrodos.

(b) Lado derecho. $d_{min} = 1,52mm$



(c) Lado izquierdo. $d_{min} = 1,09mm$

6 | Conclusiones

En el transcurso de este PFC se utilizaron diversas herramientas del actual estado del arte en procesamiento de imágenes biomédicas. Se investigó y desarrolló un conjunto de algoritmos y procesos sobre imágenes de MRI para lograr la registración deformable a un atlas. Se desarrolló un algoritmo de procesamiento de imágenes CT para reconstruir los electrodos del paciente y obtener la ubicación de los contactos. Se investigó y también implementó un método de registración multimodal CT-MRI para poder analizar ambos resultados anteriores en conjunto. De esta forma se cumplieron con los objetivos planteados al principio de este trabajo.

Por otra parte, se trabajó en la programación de módulos en 3DSlicer. Se desarrolló una interfaz de usuario para poder ejecutar todas las etapas de procesamiento siguiendo una serie de pasos que requieren una mínima intervención del especialista. Dentro de esta interfaz se incluyeron distintas configuraciones de visualización que permiten evaluar los resultados por medio de inspección visual.

Estos resultados se presentaron en la sección anterior y muestran un posible potencial de esta aplicación para ayudar en la programación de los electrodos.

Otro aspecto a destacar de este PFC es que se trabajó en conjunto con un médico. Este permitió también entender mejor el rol del bioingenierio en un grupo de trabajo multidisciplinario. Contar con un consultor médico fue fundamental para la comprensión de los resultados y la cirugía de DBS.

6.1. Limitaciones

Una de las limitaciones de este PFC es la cantidad de casos con los que se trabajó. Contar con más casos para el procesamiento permitiría por un lado mejorar la robustez del software (en caso de que hayan errores frente a imágenes de otras características) y validar el mismo.

Por otro lado, en lo que respecta el procesamiento en sí, la principal limitación es el hardware de cómputo utilizado. Se trabajó con un procesador Intel Core i5-3230M, CPU @ 2.60GHz $\times$ 4 y 8GB de memoria RAM. Con esta configuración la registración deformable lleva aproximadamente 3 horas. No se utilizó la resolución original de las imágenes para realizar la misma (ver Sección 2.5.3) debido a la memoria que requiere.

6.2. Trabajo futuro

Las limitaciones mencionadas dan pie a seguir trabajando en línea con este proyecto. A medida que hayan nuevos casos para esta cirugía se pueden ir incorporando nuevas

imágenes. También se pueden involucrar más médicos interesados. Hacia el final de este PFC surgió la posibilidad de colaborar con otro médico por lo que probablemente se puedan incorporar más casos.

Con respecto a la registración se puede probar conseguir financiamiento para obtener equipamiento que permita el procesamiento. Otra posibilidad es encarar el problema de segmentación utilizando técnicas de *machine learning*. Este es un tema que se está investigando y tiene resultados que alientan a seguir la misma [14].

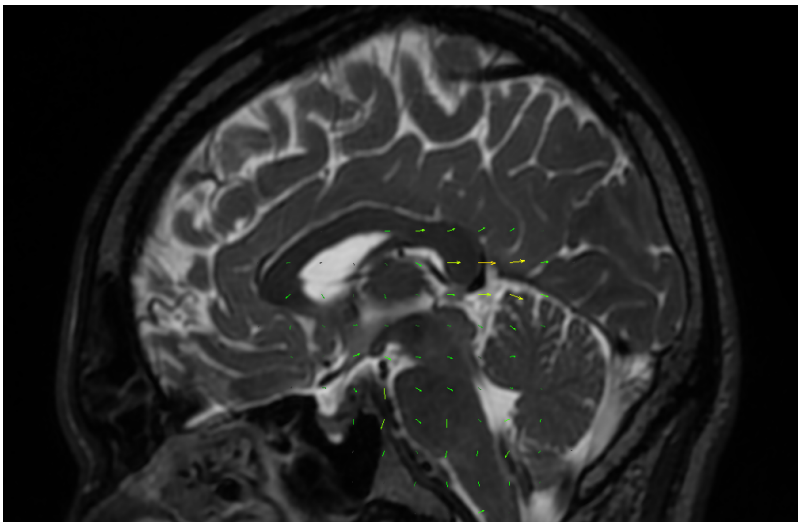
Como se mencionó en la introducción, todas las herramientas utilizadas en este PFC son de uso libre. Originalmente, un objetivo que se planteó fue desarrollar un módulo de 3DSlicer que esté disponible al público. Esta tarea sin embargo no es trivial ya que ANTs no está incorporado en 3DSlicer. Se puede seguir trabajando para hacer un módulo de registración de ANTs en 3DSlicer y luego utilizarlo como parte del software. Se puede encontrar lo realizado al día de la fecha en un repositorio privado en *GitLab*. Se debe pedir permiso al autor para tener acceso.

6.3. Máquina virtual

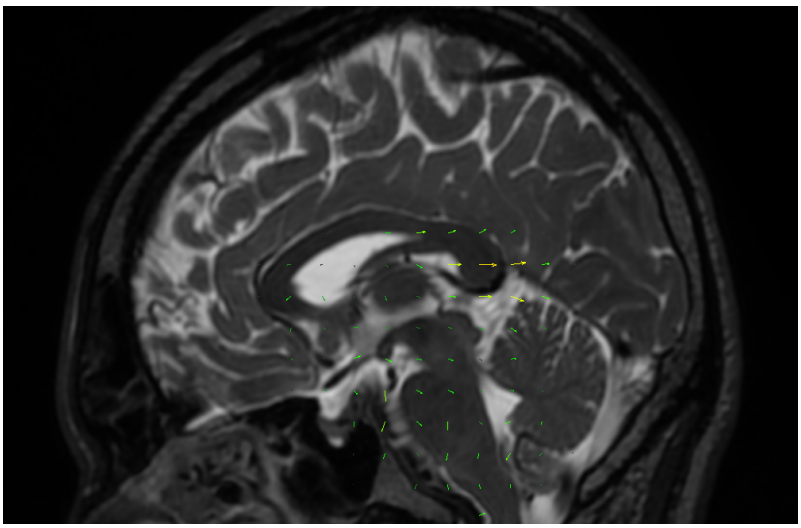
Para que el software pueda ser utilizado por el médico (y cualquier interesado con cualquier sistema operativo) lo que hizo fue armar una máquina virtual que contiene todo lo necesario para poder ejecutarse. Esta también cuenta con el caso adicional deidentificado ya procesado a modo de prueba. En el Anexo B se encuentra una breve guía para la instalación de la misma.

A | Opciones de visualización

Figura A.1: Visualización de la deformación afín y deformable. Las flechas indican el desplazamiento de los vóxeles por el resultado de la deformación.

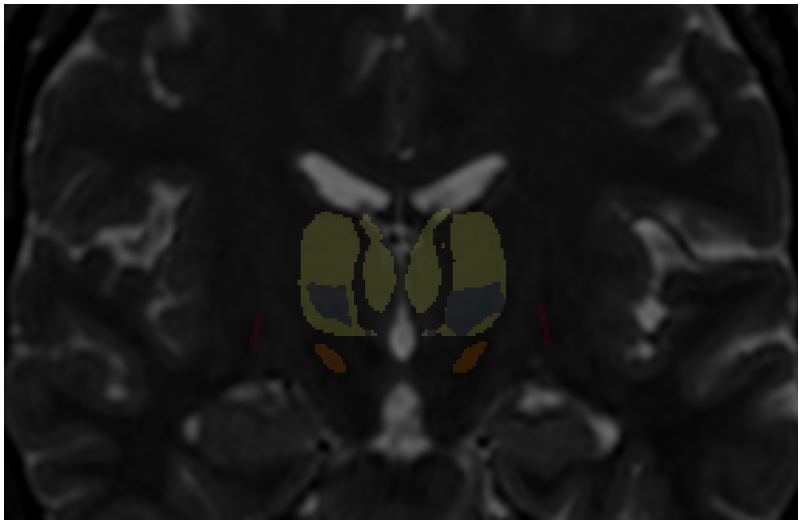


(a) *Foreground*: 0 %

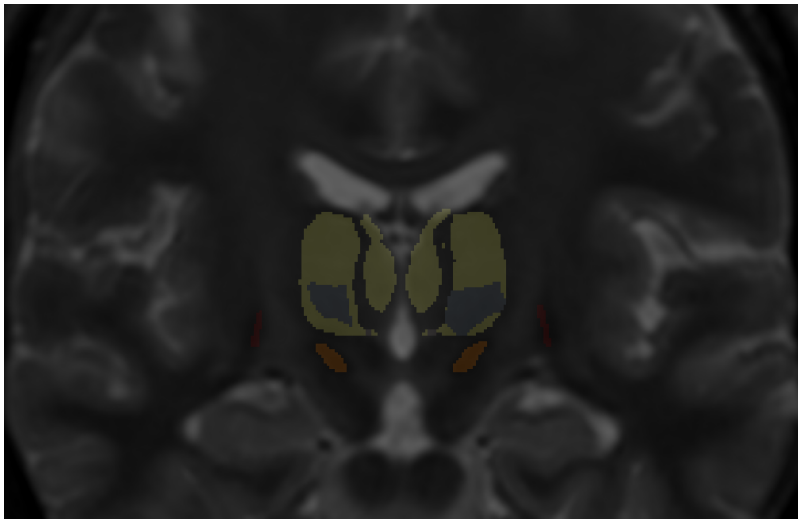


(b) *Foreground*: 100 %

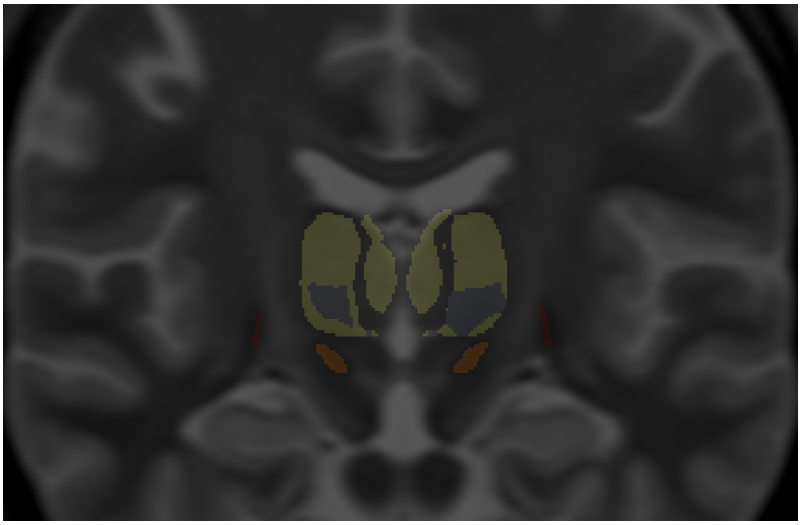
Figura A.2: Visualización de la deformación deformable y el atlas.



(a) *Foreground*: 0 %

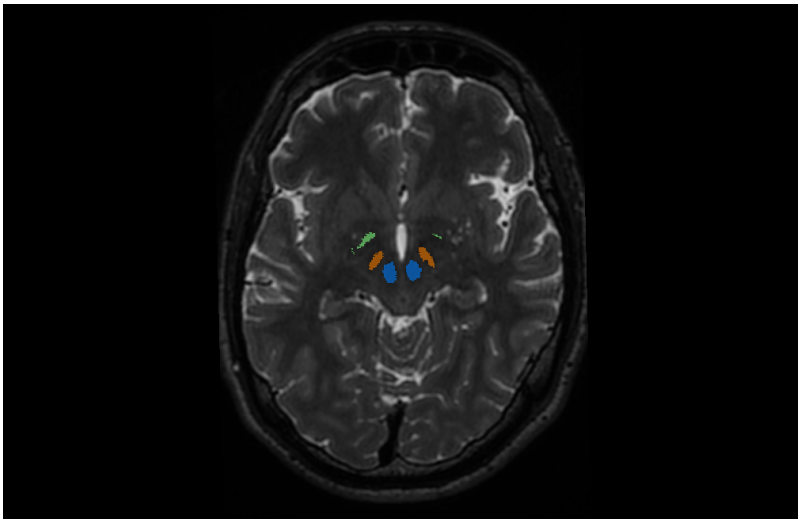


(b) *Foreground*: 50 %

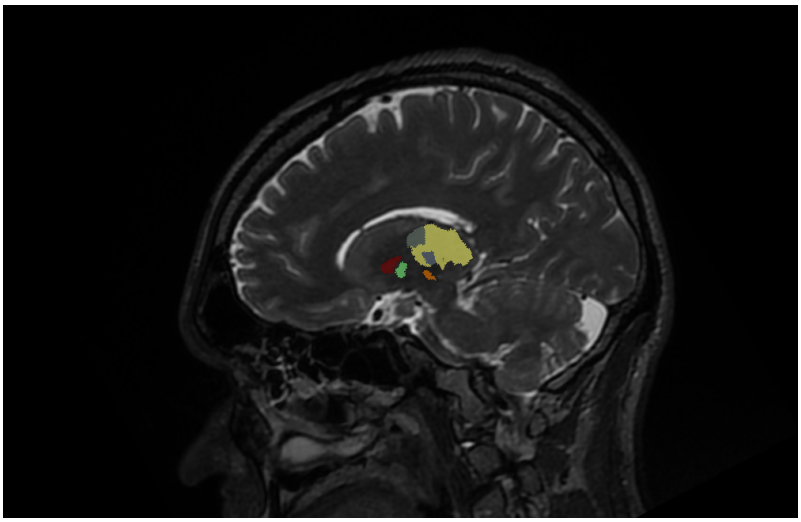


(c) *Foreground*: 100 %

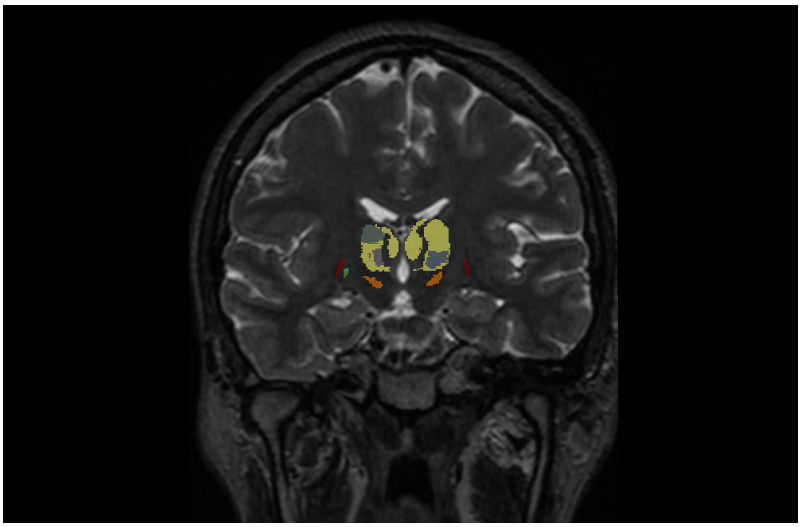
Figura A.3: Visualización de la imagen del paciente y las etiquetas del atlas en el espacio del paciente.



(a) Corte axial.

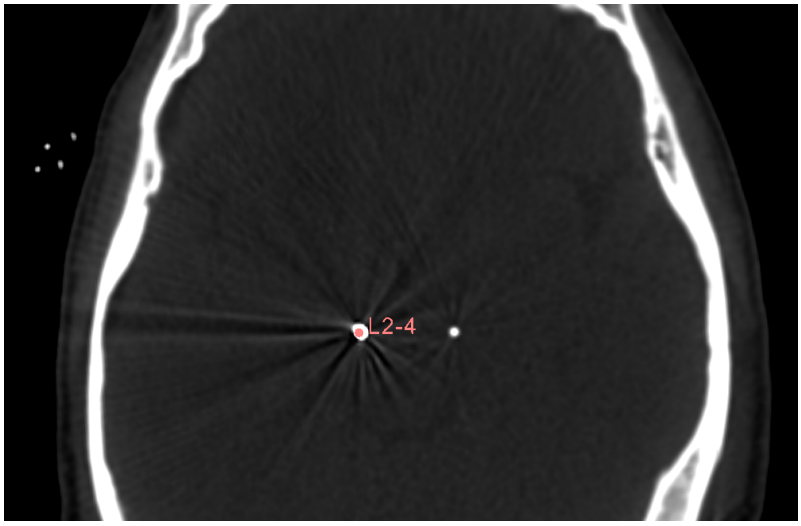


(b) Corte sagital.



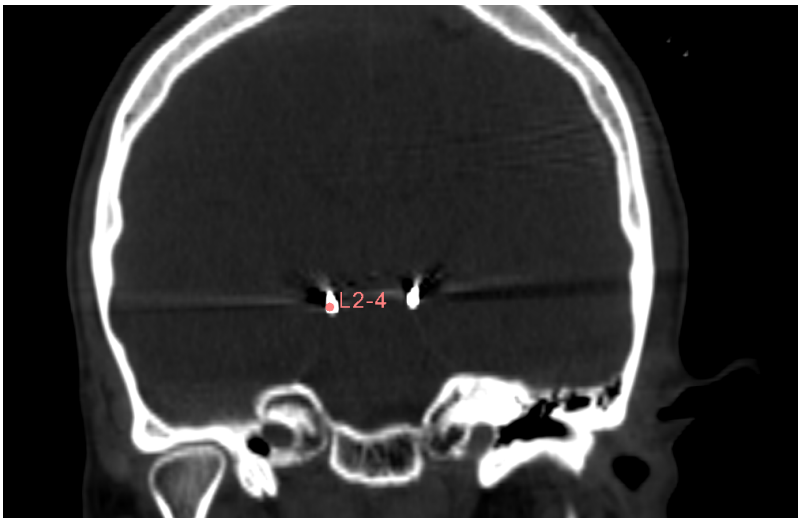
(c) Corte coronal.

Figura A.4: Visualización de la imagen CT y un fiduciario que representa un contacto de electrodo.



(a) Corte axial.

(b) Corte sagital.



(c) Corte coronal.

Figura A.5: Visualización del render la imagen CT y la reconstrucción de los electrodos.

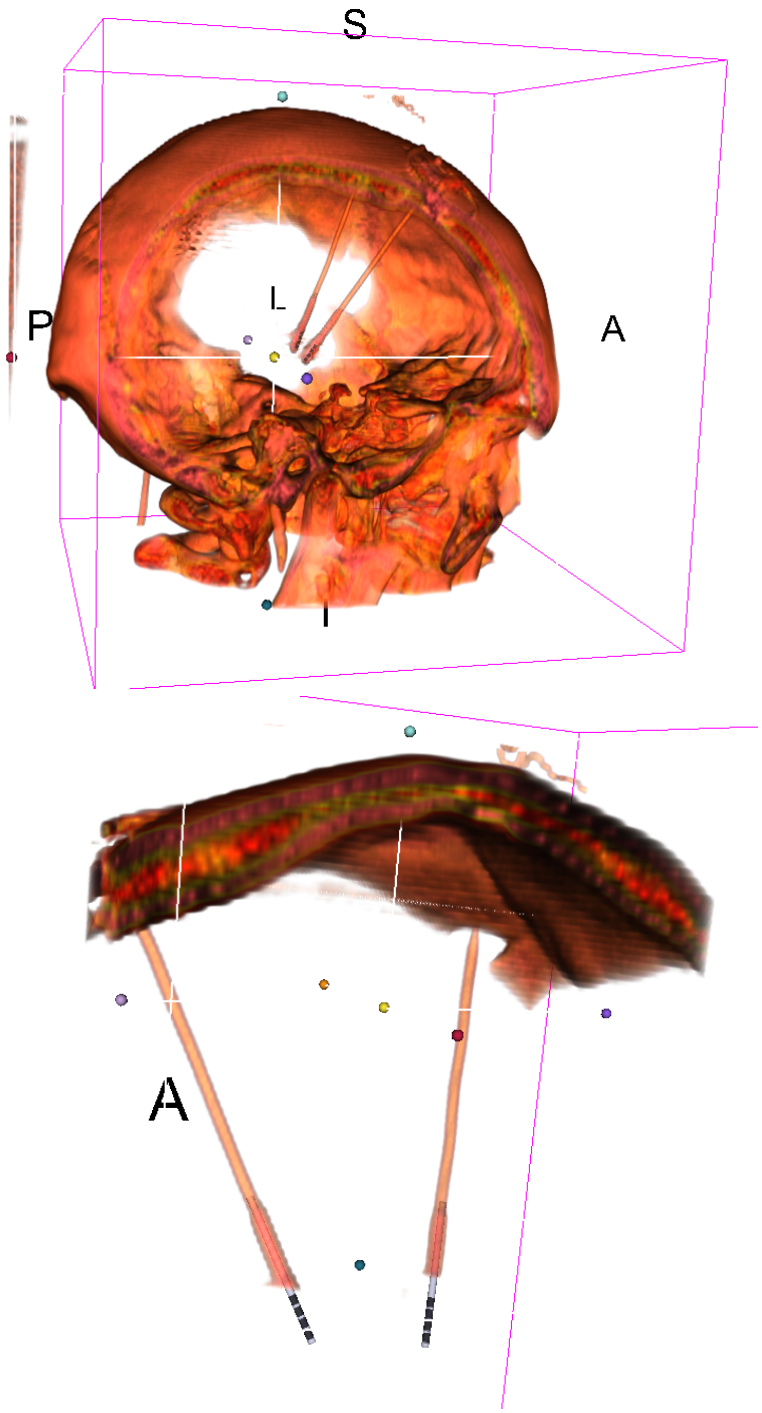
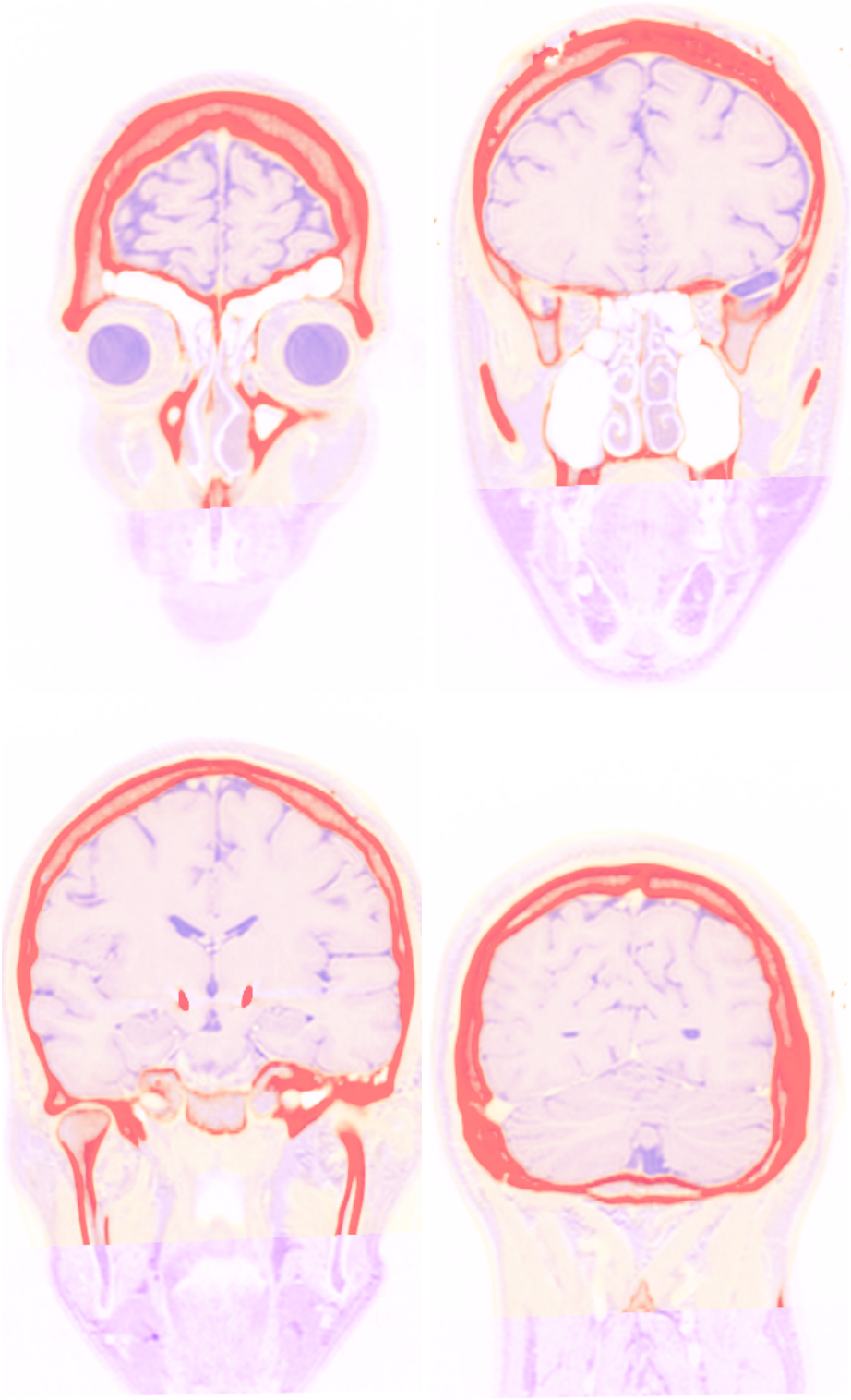


Figura A.6: Visualización de MRI y CT para distintos cortes coronales.



B | Instalación de la máquina virtual

El software de virtualización recomendado es el *VMWare Player* ya que soporta la versión 3.2 de OpenGL, necesaria para los gráficos 3DSlicer. Al día de la fecha *VM VirtualBox* no cuenta con esta especificación. A continuación se describen una serie pasos para ejecutar la máquina virtual en *VMWare Player*. En caso de utilizar otro software de virtualización se deben llevar a cabo pasos similares. Se necesitan 20GB disponibles en el disco para importar la máquina.

1. Descargar la máquina virtual de aquí.
2. Instalar *VMWare Player*.
3. En la pantalla de inicio de *VMWare Player* seleccionar "Open a Virtual Machine" y seleccionar el archivo descargado en 1.
4. Especificar nombre y directorio y seleccionar "Import".
5. Seleccionar "Retry" en caso de un aviso de falla de importación.
6. Una vez finalizada la importación seleccionar "Edit virtual machine settings".
   - 6.1. En la pestaña "Memory" establecer por lo menos 8GB en caso de querer procesar un nuevo paciente.
   - 6.2. En la pestaña "Display" habilitar "Accelerate 3D graphics" y establecer 768MB de "Graphics memory".
   - 6.3. Cerrar la ventana de configuración con OK.
7. Ejecutar la máquina con "Play virtual machine".
8. Bajo el menú "Player" > "Manage" seleccionar "Install VMWare Tools..." y proceder a la instalación (ver aquí para más detalles). La contraseña del usuario es "1234" (sin comillas).
9. Reiniciar.
10. Fin. La aplicación tiene un acceso directo en el escritorio con el nombre de "DBSLoc".

Bibliografía

[1] T. Itakura, *Deep Brain Stimulation for Neurological Disorders: Theoretical Background and Clinical Application.* Springer International Publishing, 2014.

[2] E. Montgomery, *Deep Brain Stimulation Programming: Mechanisms, Principles, and Practice.* Oxford University Press, 2016.

[3] E. A. Spiegel, H. T. Wycis, M. Marks, and A. Lee, "Stereotaxic apparatus for operations on the human brain," *Science*, vol. 106, no. 2754, pp. 349–350, 1947.

[4] https://www.bostonscientific.com/en-EU/products/deep-brain-stimulation-systems/Guide-DBS.html, [Online; accedido 11-Mayo-2019].

[5] http://newsroom.medtronic.com/phoenix.zhtml?c=251324&p=irol-newsArticle&ID=2279215, [Online; accedido 11-Mayo-2019].

[6] A. Horn and A. A. Kühn, "Lead-DBS: a toolbox for deep brain stimulation electrode localizations and visualizations," *Neuroimage*, vol. 107, pp. 127–135, 2015.

[7] H. J. Johnson, M. McCormick, L. Ibáñez, and T. I. S. Consortium, *The ITK Software Guide*, 3rd ed., Kitware, Inc., 2013, *In press.* [Online]. Available: http://www.itk.org/ItkSoftwareGuide.pdf

[8] W. J. Schroeder, B. Lorensen, and K. Martin, *The visualization toolkit: an object-oriented approach to 3D graphics.* Kitware, 2004.

[9] A. Fedorov, R. Beichel, J. Kalpathy-Cramer, J. Finet, J.-C. Fillion-Robin, S. Pujol, C. Bauer, D. Jennings, F. Fennessy, M. Sonka *et al.*, "3D Slicer as an image computing platform for the quantitative imaging network," *Magnetic resonance imaging*, vol. 30, no. 9, pp. 1323–1341, 2012.

[10] T. N. Avants B.B. and J. H., *Advanced Normalization Tools (ANTS)*, 2014. [Online]. Available: https://github.com/stnava/ANTsDoc/raw/master/ants2.pdf

[11] E. J. Brunenberg, B. Platel, P. A. Hofman, B. M. ter Haar Romeny, and V. Visser-Vandewalle, "Magnetic resonance imaging techniques for visualization of the subthalamic nucleus: a review," *Journal of neurosurgery*, vol. 115, no. 5, pp. 971–984, 2011.

[12] B. Li, C. Jiang, L. Li, J. Zhang, and D. Meng, "Automated segmentation and reconstruction of the subthalamic nucleus in parkinson's disease patients," *Neuromodulation: Technology at the Neural Interface*, vol. 19, no. 1, pp. 13–19, 2016.

[13] S. Lorio, S. Fresard, S. Adaszewski, F. Kherif, R. Chowdhury, R. S. Frackowiak, J. Ashburner, G. Helms, N. Weiskopf, A. Lutti *et al.*, "New tissue priors for improved automated classification of subcortical brain structures on MRI," *NeuroImage*, vol. 130, pp. 157–166, 2016.

[14] J. Kim, Y. Duchin, R. R. Shamir, R. Patriat, J. Vitek, N. Harel, and G. Sapiro, "Automatic localization of the subthalamic nucleus on patient-specific clinical MRI by incorporating 7T MRI and machine learning: Application in deep brain stimulation," *Human brain mapping*, vol. 40, no. 2, pp. 679–698, 2019.

[15] B. Drayer, "Basal ganglia: significance of signal hypointensity on T2-weighted MR images." *Radiology*, vol. 173, no. 2, pp. 311–312, 1989.

[16] S. Klein and M. Staring, *Elastix, the manual*, 2018. [Online]. Available: https://github.com/SuperElastix/elastix/releases/download/4.9.0/elastix-4.9.0-manual.pdf

[17] M. Brett, I. S. Johnsrude, and A. M. Owen, "The problem of functional localization in the human brain," *Nature reviews neuroscience*, vol. 3, no. 3, p. 243, 2002.

[18] J. Talairach and P. Tournoux, *Co-planar Stereotaxic Atlas of the Human Brain: 3-dimensional Proportional System : an Approach to Cerebral Imaging*, ser. Thieme classics. G. Thieme, 1988.

[19] D. L. Collins, P. Neelin, T. M. Peters, and A. C. Evans, "Automatic 3D intersubject registration of MR volumetric data in standardized talairach space." *Journal of computer assisted tomography*, vol. 18, no. 2, pp. 192–205, 1994.

[20] Y. Xiao, V. Fonov, M. M. Chakravarty, S. Beriault, F. Al Subaie, A. Sadikot, G. B. Pike, G. Bertrand, and D. L. Collins, "A dataset of multi-contrast population-averaged brain MRI atlases of a parkinson's disease cohort," *Data in brief*, vol. 12, pp. 370–379, 2017.

[21] V. Fonov, A. C. Evans, K. Botteron, C. R. Almli, R. C. McKinstry, D. L. Collins, B. D. C. Group *et al.*, "Unbiased average age-appropriate atlases for pediatric studies," *Neuroimage*, vol. 54, no. 1, pp. 313–327, 2011.

[22] M. Milchenko, S. A. Norris, K. L. Poston, M. C. Campbell, M. Ushe, J. S. Perlmutter, and A. Z. Snyder, "7T MRI subthalamic nucleus atlas for use with 3T MRI," *Journal of Medical Imaging*, vol. 5, no. 1, p. 015002, 2018.

[23] S. Ewert, P. Plettig, N. Li, M. M. Chakravarty, D. L. Collins, T. M. Herrington, A. A. Kühn, and A. Horn, "Toward defining deep brain stimulation targets in MNI space: a subcortical atlas based on multimodal MRI, histology and structural connectivity," *Neuroimage*, vol. 170, pp. 271–282, 2018.

[24] https://github.com/ANTsX/ANTs/wiki/Anatomy-of-an-antsRegistration-call, [Online; accedido 25-Mayo-2019].

[25] B. B. Avants, C. L. Epstein, M. Grossman, and J. C. Gee, "Symmetric diffeomorphic image registration with cross-correlation: evaluating automated labeling of elderly and neurodegenerative brain," *Medical image analysis*, vol. 12, no. 1, pp. 26–41, 2008.

[26] S. Ewert, A. Horn, F. Finkel, N. Li, A. A. Kühn, and T. M. Herrington, "Optimization and comparative evaluation of nonlinear deformation algorithms for atlas-based segmentation of DBS target nuclei," *NeuroImage*, vol. 184, pp. 586–598, 2019.

[27] T. Schönecker, A. Kupsch, A. Kühn, G.-H. Schneider, and K.-T. Hoffmann, "Automated optimization of subcortical cerebral MR Imaging- atlas coregistration for improved postoperative electrode localization in deep brain stimulation," *American Journal of Neuroradiology*, vol. 30, no. 10, pp. 1914–1921, 2009.

[28] A. Husch, M. V. Petersen, P. Gemmar, J. Goncalves, and F. Hertel, "PaCER-a fully automated method for electrode trajectory and contact reconstruction in deep brain stimulation," *NeuroImage: Clinical*, vol. 17, pp. 80–89, 2018.