

**ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA DE
TELECOMUNICACIÓN
UNIVERSIDAD POLITÉCNICA DE CARTAGENA**



Proyecto Fin de Carrera

**INTEGRACIÓN DE CORRESPONDENCIAS GEOMÉTRICAS EN
UN MARCO DE REGISTRO DE IMAGEN VARIACIONAL**

Raúl Bejerano Urrea

2014

Director: *Dr. D. Jorge Larrey Ruiz*



Resumen

El registro de imagen es un recurso cada vez más utilizado en diversas aplicaciones de procesado digital de imagen cuyo objetivo final sea un correcto análisis, visualización o almacenamiento de la información compartida por conjuntos de datos (imágenes o volúmenes) que representan una misma escena. En este proceso se estima una transformación geométrica que relacione puntos correspondientes entre los conjuntos de datos, consiguiendo en última instancia la mejor superposición de los mismos.

El registro de imagen se utiliza en diversos campos de Ciencia e Ingeniería (sistemas de telemetría, Medicina, cartografía, visión artificial, etc.). También se puede aplicar el registro de imagen al campo de la seguridad. Sin embargo, en los últimos años se ha dado una mayor importancia al campo de Medicina y es precisamente en este campo en el que vamos a probar lo expuesto en este proyecto final de carrera.

En este trabajo revisamos el marco matemático del registro de imagen en el dominio de la frecuencia (dominio de Fourier), centrándonos en un método de registro variacional: el registro fluido. Se pretende integrar correspondencias geométricas en el marco de registro de imagen variacional.

Para la obtención de correspondencias geométricas se propone el uso de la función SURF (Speeded Up Robust Features). En este trabajo explicamos el funcionamiento matemático de dicha función, ofreciendo, también, ejemplos gráficos de su funcionamiento.

La idea básica propuesta en este trabajo es la integración de correspondencias geométricas mediante dos métodos diferentes: el método de *resultante global* y el método de *parches locales*. Ambos métodos se han formulado para su implementación en cualquier dimensión, habiendo comparado resultados tanto en 2D como en 3D.

El método al cual se ha llamado de *resultante global*, ya había sido formulado en la literatura del registro de imagen, aunque el objetivo de este proyecto es llevarlo a la práctica para posteriores comparaciones. Nuestra aportación al registro de imagen es el método de *parches locales*. A raíz de obtener la formulación del método de *resultante global*, se nos ocurrió este método, el cual considerábamos más eficiente a la hora de ponerlo en la práctica.

La implementación de la función SURF para imágenes multimodales y para conjuntos de volúmenes (3D) constituye la principal línea futura de investigación que permitirá una mayor eficiencia en la implementación de lo propuesto en este trabajo.

Agradecimientos

Me gustaría agradecer a muchas personas todo el apoyo prestado durante estos cinco años que he estado en la universidad.

En primer lugar, al director de mi proyecto, el Dr. D. Jorge Larrey Ruiz, Profesor de la *Universidad Politécnica de Cartagena*, agradecerle por toda su dedicación y ayuda a lo largo de este último año. Le agradezco haberme dejado realizar este proyecto y la paciencia que ha demostrado aguantándome todo el año.

También quiero agradecerle todo el apoyo mostrado a mi familia. Primero, me gustaría agradecerle a mi abuela haberme acogido los dos primeros años y haber estado siempre ahí para todo. Por supuesto tengo que agradecerle a mis padres, Juanma y Flory, todos los esfuerzos que han hecho para que pueda llegar hasta aquí, al igual que el apoyo recibido de mi hermano, Alberto.

Y agradecer a Bea toda su fuerza y su cariño transmitidos en este año, lo que ha permitido que la realización de este proyecto fuera mucho más sencilla y llevadera.

Y para finalizar no puedo olvidarme de mis amigos. Empezando por mis compañeros de carrera que han estado aguantándome todos estos años (Adri Flores, Andrés, María Jesús, Ana, Adri Sánchez, Cristina, los “veteranos” y todos los demás). Sin olvidarme de mis amigos de La Manga ni de “los amiguicos”, que siempre han estado ahí.

Índice

| | |
|---|------------|
| Resumen | III |
| Agradecimientos | V |
| 1. Introducción al registro de imagen | 1 |
| 1.1. Definición | 1 |
| 1.2. Aplicaciones | 1 |
| 1.3. Clasificación | 1 |
| 1.3.1. Clasificación atendiendo al modelo de deformación impuesto | 2 |
| 1.3.2. Clasificación atendiendo a las características empleadas | 3 |
| 2. Algoritmo de obtención de correspondencias robustas: SURF | 5 |
| 2.1. Introducción | 5 |
| 2.2. Detección | 5 |
| 2.2.1. Matriz Hessiana de puntos de interés | 6 |
| 2.3. Descripción | 8 |
| 2.3.1. Asignación de la orientación | 8 |
| 2.3.2. Componentes del descriptor | 8 |
| 2.4. Aplicación | 10 |
| 2.4.1. Detección | 10 |
| 2.4.2. Descripción | 11 |
| 3. Formulación variacional del registro de imagen | 13 |
| 3.1. Enfoque variacional en el dominio espacial | 13 |
| 3.2. Registro por difusión | 16 |
| 3.3. Implementación frecuencial de las ecuaciones de Euler-Lagrange | 17 |
| 4. Integración de correspondencias geométricas en un marco de registro de imagen variacional | 18 |
| 4.1. Detección de correspondencias geométricas | 18 |
| 4.2. Formulación | 19 |
| 4.3. Método de resultante global | 20 |
| 4.4. Método de parches locales | 21 |
| 5. Resultados | 23 |
| 5.1. Resultados 2D | 23 |
| 5.1.1. Método de resultante global | 23 |
| 5.1.2. Método de parches locales | 26 |
| 5.1.3. Registro con marco variacional | 28 |
| 5.1.4. Comparación de resultados | 32 |
| 5.2. Resultados 3D | 33 |
| 5.2.1. Método de resultante global | 33 |
| 5.2.2. Método de parches locales | 35 |

| | | |
|-----------|--|-----------|
| 5.2.3. | Registro con marco variacional | 38 |
| 5.2.4. | Comparación de resultados | 41 |
| 6. | Conclusiones y líneas futuras | 43 |
| 7. | Bibliografía | 45 |

1. Introducción al registro de imagen

1.1. Definición

El registro de imagen es el proceso de alinear dos o más imágenes de la misma escena tomadas en diferentes momentos, desde diferentes puntos de vista y/o por diferentes sensores. Dicho registro es un paso crucial en problemas de imagen cuando más de una imagen contiene información valiosa. En este caso, se requiere la alineación espacial para integrar adecuadamente la información útil a partir de las imágenes separadas.

Resulta que el problema de registro es un problema inverso, que no requiere una regularización de sonido y el uso de modelos adecuados. Además, los valores numéricos tienen que hacerse con mucho cuidado. [1]

1.2. Aplicaciones

Hay un gran número de aplicaciones que necesitan el registro. Las áreas van desde el arte, la astronomía, la astrofísica, la biología, la química, la criminología, la genética, la física, o básicamente cualquier área que implica técnicas de imagen. Los ejemplos más específicos apoyados por el registro son, por ejemplo, la teledetección (la construcción de una visión global de los diferentes puntos de vista parciales), la seguridad (comparar las imágenes actuales con una base de datos), la robótica (seguimiento de objetos), y, en particular, la medicina, donde la anatomía computacional, diagnóstico asistido por ordenador, la fusión de diferentes modalidades, la intervención y la planificación del tratamiento, la vigilancia de las enfermedades, corrección de movimiento o la radioterapia.

Dado que las técnicas de formación de imágenes, como la tomografía computarizada (TC), la difusión de imágenes de tensor (DTI), resonancia magnética (MRI), tomografía por emisión de positrones (PET), tomografía computarizada por emisión de fotón único (SPECT) o ultrasonido (EE.UU.) fueron sometidos a una notable mejoría, fascinante y permanente en la última década, un enorme aumento en la utilización de las distintas modalidades de la medicina llevada a cabo.

1.3. Clasificación

Los métodos de registro de imagen se pueden registrar según diversos criterios. A lo largo de los años se han ido proponiendo multitud de clasificaciones en la literatura, pero siempre han compartido un criterio de diferenciación: el basado en las características de las imágenes, es decir, en el tipo de información que guía el proceso de registro hacia la correcta estimación de la transformación que relaciona las imágenes a registrar. La gran mayoría de las clasificaciones dividen este criterio de diferenciación en dos partes: por un lado tenemos los métodos geométricos, que utilizan una distancia geométrica entre características segmentadas de las imágenes; por otro lado están los métodos basados en intensidades, que usan una medida de similitud entre las intensidades de las imágenes.

Desde el punto de vista del número de elementos de los conjuntos de datos sobre los que se aplica la medida de similitud (o distancia) seleccionada, se suele hablar de métodos con

modelo de mapeo global y métodos con modelo de mapeo local. Los métodos globales utilizan todo el conjunto de datos como soporte para estimar una cierta transformación global, que se considera válida en la imagen entera (transformaciones rígidas o de base radial). Los métodos locales tratan la imagen como un mosaico compuesto por múltiples teselas, de forma que los parámetros de la transformación son diferentes para cada una de ellas, ya que dependen de su posición y forma (métodos de block matching (BMA) o algoritmo iterativo generalizado del punto más próximo (ICP)). Los métodos locales necesitan normalmente de técnicas adicionales para mantener la consistencia geométrica de la transformación calculada para las diferentes teselas del mosaico.

Vamos a seguir la clasificación propuesta en [2] que sintetiza la multitud de posibles “ejes” únicamente en dos y que se explican a continuación.

1.3.1. Clasificación atendiendo al modelo de deformación impuesto

Debido al inconveniente del mal condicionamiento del registro de imagen, resulta necesaria la regularización del problema mediante la imposición de un modelo de deformación al algoritmo de registro, incorporando así el conocimiento a priori disponible sobre la clase de la transformación que se pretende estimar. En caso contrario se estimaría el desplazamiento de cada punto (píxel o vóxel) independientemente de los puntos vecinos, de modo que el proceso de registro proporcionaría un campo de vectores de correspondencia que resultaría inverosímil para el escenario de aplicación concreto, y que con una alta probabilidad presentaría numerosas discontinuidades, así como agujeros o pliegues en la imagen. El modelo de deformación debe por tanto restringir la transformación estimada a un espacio de soluciones adecuado. En la literatura se encuentran tres modelos de deformación principales: paramétricos, competitivos e incrementales.

Modelos paramétricos

En los modelos paramétricos se aplica la restricción de que la deformación estimada debe pertenecer a un determinado espacio, típicamente de pocas dimensiones, de transformaciones verosímiles. Matemáticamente, la aproximación paramétrica consigue el registro mediante la minimización de una cierta distancia definida entre los conjuntos de datos que se desea registrar. Entre los espacios de soluciones más populares en la literatura, encontramos por un lado las transformaciones rígidas, y por otro lado el formado por funciones base o núcleos consistentes en thin plate splines o producto tensor de B-splines, bases de deformaciones como la de Fourier, wavelets, componentes principales de un conjunto de deformaciones de entrenamiento, o autofunciones de operadores lineales.

Modelos competitivos

Los modelos competitivos se basan en el uso de un término de regularización que depende de la propia transformación. Mientras que los paramétricos presentan una penalización de tipo binario, la aproximación competitiva penaliza una transformación de forma proporcional a la irregularidad de esta. Los algoritmos competitivos consiguen el registro mediante la minimización de la suma ponderada de la medida de distancia y la energía de regularización, imponiendo en última instancia unas ciertas condiciones de contorno sobre los vectores de correspondencias que se obtendrán. Habitualmente los encontramos como métodos globales e iterativos de un paso. Un término de regularización muy popular en

registro de imagen es la energía de elasticidad lineal, en cuyo caso se habla de registro elástico.

Modelos incrementales

Los modelos incrementales hacen también uso de una energía de regularización que, a diferencia de la aproximación anterior, no dependerá de la transformación en sí misma, sino de su evolución temporal. El caso más simple tiene lugar cuando la energía de regularización depende de la diferencia entre la estimación actual de la transformación y la estimación anterior; se observa por tanto que estos métodos dependen de la estimación inicial de la deformación. Suelen ser métodos globales e iterativos de un paso.

1.3.2. Clasificación atendiendo a las características empleadas

Como ya se ha comentado antes, clasificar los algoritmos de registro en base a qué características de los conjuntos de datos se emplean para estimar la transformación que los relaciona geoméricamente constituye el otro “eje” de clasificación. Así, podemos dividir los algoritmos de registro en tres categorías fundamentales: basados en características geométricas, basados en intensidades y basados en características icónicas.

Métodos basados en características geométricas

Los métodos pertenecientes al primer grupo se sirven de la segmentación de toda la imagen, o de una parte de misma, que tiene lugar antes del proceso de registro propiamente. Los objetos geoméricos segmentados se corresponden generalmente a invariantes que pueden ser anatómicos o estructurales, o bien matemáticos, los descriptores de forma, los momentos invariantes o los coeficientes de similitud. Una vez extraídos, estos rasgos son registrados mediante la minimización de una distancia geométrica que se define entre ellos. Esta distancia suele requerir el cálculo explícito de un conjunto de correspondencias entre los puntos de ambos conjuntos de datos. Se busca entonces la transformación que interpola o aproxima las anteriores correspondencias, a la vez que minimiza la función distancia. Podemos encontrar combinaciones de estos métodos tanto con los modelos paramétricos como con los competitivos o incrementales.

Modelos basados en intensidades

Los niveles de gris en las imágenes suponen otro tipo de información ampliamente utilizada para el registro de la imagen. En los métodos basados en intensidades “crudas”, la distancia que se calcula entre los conjuntos de datos a registrar se define a partir de una de las muchas medidas de similitud existentes en la literatura, como por ejemplo la suma de diferencias absolutas (SAD) o la suma de diferencias cuadráticas (SSD), la información mutua (MI) o el ratio de correlación (CR). La medida de similitud se calcula entre las intensidades de pares de puntos ubicados en la misma posición espacial de ambas imágenes, de modo que cuando la distancia sea mínima, las imágenes se corresponderán geoméricamente. Estos métodos suelen combinarse con los modelos competitivos, dando lugar a lo que se conoce como técnicas de registro de imagen no paramétrico o de registro de imagen variacional. Precisamente en este proyecto nos centramos en un marco de registro de imagen variacional, aunque le añadiremos ciertas correspondencias geométricas.

Métodos basados en características icónicas

Algunos métodos basados en intensidades no funcionan exactamente del modo que se explica en el punto anterior. En este bloque enmarcaremos estos algoritmos, los cuales siempre tienen un modelo de mapeo local, y se encuentran en un punto intermedio entre las dos categorías anteriores, puesto que las características cuya distancia geométrica se trata de minimizar no se obtienen a partir de la segmentación de las imágenes, sino que se emparejan en base a una medida de similitud de niveles de intensidad. El algoritmo iterativo generalizado del punto más próximo (ICP) puede considerarse como perteneciente a este grupo.

2. Algoritmo de obtención de correspondencias robustas: SURF

2.1. Introducción

La tarea de la búsqueda de correspondencias entre dos imágenes de la misma escena u objeto es parte de las muchas aplicaciones de la visión por ordenador. La calibración de la cámara, la reconstrucción 3D, el registro de imágenes y el reconocimiento de objetos son sólo unas pocas.

La detección de características es el proceso en el que examinamos automáticamente una imagen para extraer características, que son exclusivas de los objetos en la imagen, de tal manera que somos capaces de detectar un objeto basándonos en sus características en diferentes imágenes. Esta detección debería ser posible idealmente cuando la imagen muestra el objeto con diferentes transformaciones, principalmente escalado y rotación, o cuando partes del objeto son ocluidas.

La búsqueda de correspondencias se puede dividir en tres pasos principales. En primer lugar la **detección**, en la que se seleccionan "puntos de interés" en ubicaciones distintivas de la imagen, tales como esquinas, *blobs*, y uniones en forma de T. La propiedad más valiosa de un detector de puntos de interés es su repetibilidad, es decir, si encuentra de forma fiable los mismos puntos de interés bajo diferentes condiciones de visión. A continuación la **descripción**, donde la vecindad de cada punto de interés se representa por un vector de características. Este descriptor tiene que ser distintivo y, al mismo tiempo, robusto al ruido, a la detección errores, y a deformaciones geométricas y fotométricas. Finalmente el **registro**, en el que los vectores descriptores se emparejan entre las diferentes imágenes. El registro se basa a menudo en una distancia entre los vectores, por ejemplo la distancia de Mahalanobis o Euclidiana. La dimensión del descriptor tiene un impacto directo sobre el tiempo que tarda, por lo que es deseable un número bajo de dimensiones. [3][4]

2.2. Detección

La Transformación de Características Invariantes en Escala, SIFT (Scale-Invariant Feature Transform) es un enfoque exitoso para detectar características introducido por Lowe [5]. El algoritmo SURF (Speeded-Up Robust Features) se basa en los mismos principios y pasos, pero utiliza un esquema diferente y debe ofrecer mejores resultados en menos tiempo.

Con el fin de detectar puntos característicos de una manera con escala invariante, SIFT utiliza un enfoque de filtrado en cascada. Donde la diferencia de gaussianas, DoG, es calculada en imágenes a escala reducidas progresivamente.

En general, la técnica para lograr invariancia de escalado es la de examinar la imagen en diferentes escalas, escalas espaciales, que utilizan núcleos gaussianos. Tanto SIFT como SURF dividen el espacio de escala en niveles y octavas. Una octava corresponde a una duplicación de σ , y la de la octava se divide en niveles espaciados de manera uniforme.

Ambos enfoques construyen una pirámide de mapas de respuesta, con diferentes niveles dentro de las octavas. Un mapa de respuesta es el resultado de una operación en la imagen. Los puntos de interés son los puntos que son los extremos entre los 8 vecinos en el nivel actual y sus 2x9 vecinos en el nivel inferior y superior. Esta es una supresión no máxima en un vecindario 3x3x3.

2.2.1. Matriz Hessiana de puntos de interés

SURF utiliza un detector hessiano basado en *blobs*(gotas) para encontrar puntos de interés. El determinante de una matriz hessiana expresa la extensión de la respuesta y es una expresión de la variación local alrededor de la zona.

$$H(x,\sigma) = \begin{bmatrix} L_{xx}(x,\sigma) & L_{xy}(x,\sigma) \\ L_{xy}(x,\sigma) & L_{yy}(x,\sigma) \end{bmatrix}, \quad (2.1)$$

donde

$$L_{xx}(x,\sigma) = I(x) * \frac{\partial^2}{\partial^2 x} g(\sigma), \quad (2.2)$$

$$L_{xy}(x,\sigma) = I(x) * \frac{\partial^2}{\partial xy} g(\sigma), \quad (2.3)$$

$L_{xx}(x,\sigma)$ en la ecuación 2 es la convolución de la imagen con la segunda derivada de la gaussiana. El corazón de la detección SURF es la supresión no-máxima de los determinantes de las matrices hessianas. Las convoluciones son muy costosas computacionalmente y se aproximan y aceleran con el uso de las imágenes integrales y núcleos aproximados.

Una imagen integral $I(x)$ es una imagen en la que cada punto $x = (x, y)^T$ almacena la suma de todos los píxeles en un área rectangular entre el origen y x (véase la ecuación 2.4).

$$I(x) = \sum_{i=0}^{i \leq x} \sum_{j=0}^{j \leq y} I(x, y), \quad (2.4)$$

El uso de imágenes integrales permite calcular la respuesta en un área rectangular con tamaño arbitrario usando 4 búsquedas como se ilustra en la Figura 2.1.

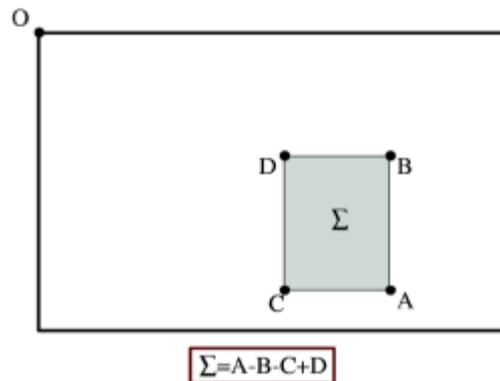


Figura 2.1: 4 búsquedas de memoria es suficiente para calcular la suma de un área rectangular con una imagen integral

Los núcleos gaussianos de segundo orden $\frac{\partial^2}{\partial^2 y} g(\sigma)$ utilizados para la matriz hessiana deben ser discretizados y recortados antes de que podamos aplicarlos. El algoritmo SURF aproxima estos núcleos con cajas rectangulares, filtros de caja.

$$\text{Det}(H_{\text{aprox}}) = D_{xx}D_{xy} - (\omega D_{xy})^2, \quad (2.5)$$

Los núcleos discretos y aproximados se conocen como D_{yy} para $L_{yy}(x,\sigma)$ y D_{xy} para $L_{xy}(x,\sigma)$. El valor de $\sigma = 1,2$ es la escala más baja que el algoritmo SURF puede manejar. Cuando se utilizan los núcleos aproximados para calcular el determinante de la matriz Hessiana, tenemos que ponderar con ω en la ecuación 5, esto es para asegurar la conservación de la energía en las gaussianas. El término ω es teóricamente sensible al escalado, pero se puede mantener constante a 0,9. [3]

Para detectar características en toda la escala, tenemos que examinar varias octavas y niveles, donde SIFT escala la imagen hacia abajo para cada octava y utiliza núcleos de Gaussianas progresivamente más grandes. Las imágenes integrales permiten al algoritmo SURF calcular las respuestas con núcleos arbitrariamente más grandes.

Esto plantea dos retos, la forma de escalar los núcleos aproximados y cómo esto influye en los posibles valores de σ . Los núcleos tienen que tener un tamaño desigual para tener un pixel central y las áreas rectangulares. Los lóbulos, tienen que tener el mismo tamaño.

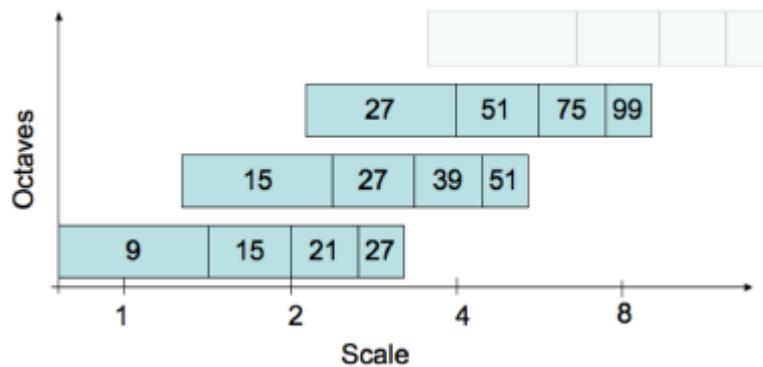


Figura 2.2: La escala (σ), el tamaño del filtro y las escalas en una escala logarítmica

El papel de SURF entra en detalle con estas consideraciones. El resultado es que la división de espacio de escala en los niveles y octavas se fija como se ilustra en la Figura 2.2. El tamaño del filtro será grande y si la convolución se hiciera con un núcleo gaussiano regular esto tendría un elevado coste. El uso de imágenes integrales no sólo hace esto posible; también hace que sea rápido, y sin la necesidad de reducir el escalado de la imagen. Cabe señalar que este enfoque con grandes cajas de filtros puede preservar y ser sensible a ruido de alta frecuencia.

Cuando buscamos una extrema en una de las octavas mayores, el área cubierta por el filtro es bastante grande y esto introduce un error significativo para la posición del punto de interés. Para remediar esto, la ubicación exacta del punto de interés es interpolada ajustando una cuadrática 3D en el espacio escalado. Un punto de interés se encuentra en el espacio escalado por (x, y, s) donde x, y son coordenadas relativas ($x, y \in [0; 1]$) y s es la escala.

2.3. Descripción

El propósito de un descriptor es proporcionar una descripción única y robusta de una característica. Un descriptor puede ser generado basándose en el área que rodea a un punto de interés [3]. El buen desempeño de SIFT en comparación con otros descriptores es notable. Su mezcla de información crudamente localizada y la distribución de gradiente relacionando características parece dar un buen poder distintivo mientras que se defiende de los efectos de errores de localización en términos de escalado o de espacio. Usando fuerzas relativas y orientaciones de los gradientes se reduce el efecto de los cambios fotométricos.

El descriptor SURF propuesto se basa en propiedades similares, cuya complejidad está simplificada aún más. La primera etapa consiste en la fijación de una orientación reproducible basada en la información de una región circular alrededor del punto de interés. Luego, construimos una región cuadrada alineada con la orientación seleccionada, y extraemos el descriptor SURF de ella. Estos dos pasos se explican ahora en turnos. Por otra parte, también se propone una versión vertical del descriptor (U-SURF) que no es invariante a la rotación de la imagen y por lo tanto es más rápido para calcular y más adecuado para aplicaciones en las que la cámara se mantiene más o menos horizontal. [4]

2.3.1. Asignación de la orientación

Con el fin de ser invariantes con la rotación, identificamos una orientación reproducible para los puntos de interés. Para ello, primero calculamos las respuestas Haar de las ondas Haar en la dirección x e y , y esto en una zona circular de radio $6s$ alrededor del punto de interés, siendo s la escala a la que el punto de interés fue detectado. También el paso de muestreo es de escala dependiente y elegido como s . En consonancia con el resto, también las respuestas *wavelet* se calculan en esta escala s . En consecuencia, a altas escalas del tamaño de las *wavelets* es grande. Por lo tanto, utilizamos otra vez imágenes integrales para un filtrado rápido. Sólo se necesitan seis operaciones para calcular la respuesta en la dirección x o y en cualquier escala. La longitud del lado de las ondas es $4s$.

Una vez que las respuestas de las ondas se calculan y se ponderan con una Gaussiana ($\sigma = 2.5s$) centrada en el punto de interés, las respuestas se representan como vectores en un espacio con la respuesta horizontal de la fuerza a lo largo de la abscisa y la respuesta vertical de la fuerza a lo largo de la ordenada. La orientación dominante se estima por el cálculo de la suma de todas las respuestas dentro de una ventana de orientación deslizante cubriendo un ángulo de $\pi/3$. Las respuestas horizontales y verticales dentro de la ventana son sumadas. Las dos respuestas sumadas a continuación producen un nuevo vector.

El vector más largo presta su orientación hacia el punto de interés. El tamaño de la ventana deslizante es un parámetro que ha sido elegido experimentalmente. Tanto los tamaños de ventana excesivamente pequeños como los excesivamente grandes pueden ofrecer resultados falseados. Ambos resultan en una orientación inestable de la región de interés. Cabe destacar que el U-SURF se salta este paso. [4]

2.3.2. Componentes del descriptor

Para la extracción del descriptor, el primer paso consiste en la construcción de una región cuadrada centrada alrededor del punto de interés, y orientada a lo largo de la

orientación seleccionada en la sección anterior. Para la versión vertical, esta transformación no es necesaria. El tamaño de esta ventana es de 20s.

La región se divide regularmente en pequeñas subregiones cuadradas 4×4 . Esto guarda importante información espacial. Para cada sub-región, calculamos unas pocas características simples en puntos de muestreo 5×5 espaciados regularmente. Por razones de simplicidad, llamamos dx la respuesta *wavelet* Haar en la dirección horizontal y dy la respuesta de la *wavelet* de Haar en la dirección vertical (2s el tamaño del filtro). "Horizontal" y "vertical" aquí se definen en relación a la orientación del punto de interés seleccionado. Para aumentar la robustez hacia deformaciones geométricas y errores de localización, las respuestas dx y dy son primero ponderadas con una Gaussiana ($\sigma = 3.3S$) centrada en el punto de interés.

A continuación, las respuestas *wavelet* dx y dy se resumen en cada subregión y forman un primer conjunto de entradas al vector de la función. Con el fin de aportar información sobre la polaridad de los cambios de intensidad, también extraemos la suma de los valores absolutos de las respuestas, la $|dx|$ y $|dy|$. Por lo tanto, cada sub-región tiene un vector descriptor v de cuatro dimensiones de su estructura subyacente de intensidad $v = (\sum d_x, \sum d_y, \sum |d_x|, \sum |d_y|)$. Esto resulta en un vector descriptor para todas las subregiones 4×4 de longitud 64. Las respuestas de las ondas son invariantes a un sesgo de la iluminación (offset). La invariancia de contraste (un factor de escala) se logra convirtiendo el descriptor en un vector unitario.

Para llegar a estos descriptores SURF, experimentamos con menos y más características *wavelets*, utilizando d_x^2 y d_y^2 , *wavelets* de orden superior, PCA, valores de mediana, valores medios, etc. A partir de una evaluación exhaustiva, los conjuntos propuestos en [4] resultaron ser mejores. Entonces variaron el número de puntos de muestreo y subregiones. La solución de la división en subregiones 4×4 proporcionó los mejores resultados. Teniendo en cuenta que las subdivisiones más finas parecían ser menos robustas y aumentarían demasiado los tiempos de detección. Por otra parte, el corto descriptor con subregiones 3×3 (SURF-36) actúa peor, pero permite una detección muy rápida y es todavía bastante aceptable en comparación con otros descriptores en la literatura.

También probaron una versión alternativa del descriptor SURF que añade un par de características similares (SURF-128). Utiliza de nuevo las mismas sumas que antes, pero ahora se dividen estos valores aún más. Las sumas de dx y $|dx|$ se calculan por separado para $dy < 0$ y $dy \geq 0$. Del mismo modo, las sumas de dy y $|dy|$ están divididas de acuerdo con el signo de dx , duplicando así el número de características. El descriptor es más distintivo y no mucho más lento de calcular, pero más lento para detectar debido a su mayor dimensionalidad.

Para la indexación rápida durante la etapa de adaptación, el signo del Laplaciano (es decir, la traza de la matriz Hessiana) para el punto de interés subyacente está incluido. Por lo general, los puntos de interés se encuentran en estructuras de tipo *blob*. El signo del Laplaciano distingue *blobs* brillantes sobre fondos oscuros desde la situación inversa. Esta función está disponible sin coste computacional extra, ya que fue computada durante la fase de detección. En la etapa de adaptación, sólo comparamos características si tienen el mismo tipo de contraste. Por lo tanto, esta mínima información permite una adaptación más rápida y da un ligero aumento en el rendimiento.

2.4. Aplicación

En este apartado entraremos en detalle con la implementación que [3] ha hecho de SURF.

Utiliza el marco OpenCV, este es un marco C++ para visión por computador. Se carga con su propia implementación de SURF y SIFT y otros algoritmos de visión por computador. Fue elegido, ya que proporciona un buen bajo nivel de rutinas para trabajar con imágenes y facilitar la carga y el almacenamiento de los diferentes formatos de imagen.

2.4.1. Detección

La imagen de ejemplo que usan es el ajuste de barbacoa (Figura 2.3), pero durante las pruebas de puesta en práctica de otras imágenes se han utilizado para verificar los resultados.

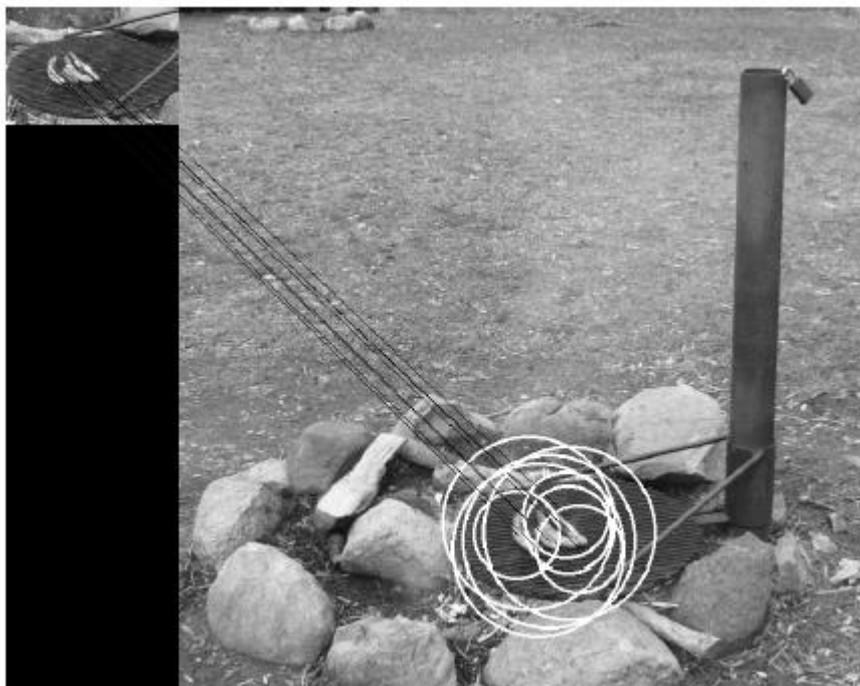


Figura 2.3: Detección de rasgos comunes entre dos imágenes con diferente escala y recorte

El primer paso es crear una imagen integral, esto se hace usando una rutina incorporada en OpenCV. Entonces se crean los mapas de respuesta para 4 niveles en 4 octavas. El tamaño de los filtros es fijo y cuando se calculan los mapas de respuesta, se calcula el tamaño exacto de los lóbulos y sus offsets en función del tamaño del filtro.

Los filtros de caja se utilizan para encontrar D_{xx} , D_{yy} y D_{xy} y luego calcular los Hessianos y Laplacianos aproximados. Antes de utilizar $D_{??}$ son normalizados; de tal manera que las respuestas no dependen del tamaño del filtro.

Las dimensiones de los mapas de respuesta son iguales dentro de una octava, y se redujo a la mitad cuando se aumenta una octava, esto es implementado mediante el aumento de la distancia entre las muestras.

En la aplicación de esta parte era muy importante tener una salida depurada significativa. Este tipo de salida acoplada con imágenes de prueba simples demostró ser una herramienta valiosa para inspeccionar el comportamiento del detector.

Después de haber construido los mapas de respuesta el siguiente paso es hacer una supresión no máxima en los valores del determinante Hessiano ($Det(H_{approx})$). Habiendo identificado un máximo, el siguiente paso es para interpolar la posición en el espacio escalado utilizando el método de Lowe que ofrece buenas valoraciones en los detalles de este paso. En esencia hay que adaptarse a una expresión cuadrática 3D que expresa la Hessiana como una función $H(x, y, \sigma)$ usando una expansión de Taylor (ecuación 2.7) para encontrar los extremos mediante el establecimiento de la derivada a cero y resolviendo la ecuación (6) para encontrar $\hat{x} = (x, y, \sigma)$:

$$H(\mathbf{x}) = H + \frac{\partial H^T}{\partial \mathbf{x}} \mathbf{x} + \frac{1}{2} \mathbf{x}^T \frac{\partial^2 H^T}{\partial \mathbf{x}^2} \mathbf{x}, \quad (2.6)$$

$$\hat{\mathbf{x}} = \frac{\partial^2 H^{-1}}{\partial^2 \mathbf{x}} \frac{\partial H}{\partial \mathbf{x}}, \quad (2.7)$$

Las derivadas se calculan a partir de las diferencias finitas en los mapas de respuesta. Y la inversa se encuentra usando una función de OpenCV con una descomposición SVD. Esto ha sido implementado de tal manera que es posible compilar con o sin interpolación.

Los resultados son consistentemente mejor sin interpolación. La interpolación es el resultado de un ajuste cuadrático 3D en espacio escalado y la interpolación en sí puede dar resultados erróneos, es decir resultados que no son razonables alejándose del punto inicial. Cuando se estudia el origen de openSURF y el *plugin* de la imagen J era obvio que el criterio exacto para un mal resultado hasta en la interpretación. El resultado de la interpolación es relativa a los *offsets* desde el punto original, y para escardar hacia fuera los malos resultados, los resultados de interpolación con cualquier componente > 0.5 descartan el *offset*.

2.4.2. Descripción

En el apartado anterior se ha descrito cómo identificar puntos de interés, en esta sección vamos a entrar en detalles sobre la descripción de estos puntos.

En [3] se ha implementado la variante U-SURF, en la que no asigna una orientación hacia el punto de interés. Esta debería ser invariante a la rotación y la asignación de la orientación podría ser implementada en una etapa posterior para mejorar el rendimiento.

Para calcular los descriptores, cada una de las 16 subáreas que rodean el punto de interés son muestreadas para obtener la respuesta *wavelet* dx y dy juntas con sus valores absolutos. La respuesta *wavelet* se calcula utilizando la imagen integral con la caja de filtros con filtros de tamaño 2s. Para el caso dx este es el valor del cuadrado de la derecha, menos el valor del cuadrado de la izquierda del punto de interés. Esta respuesta se pondera con una gaussiana ($\sigma = 3.3$) centrada en el punto de interés.

El vector formado por los valores concatenados de los 64 descriptores como último paso se normaliza. Los puntos de interés se pueden guardar en una lista básica ASCII donde la precisión es importante. Fue interesante observar que con la precisión predeterminada para

almacenar y cargar una lista y las pruebas en contra de la misma imagen, no le daría un 100% de coincidencia.

3. Formulación variacional del registro de imagen

En los apartados anteriores se ha hecho una introducción al registro de imagen y al funcionamiento del programa SURF. En cambio, en este apartado nos centraremos en la formulación empleada en [2] para el registro de imagen en un marco variacional.

Se puede seguir cualquiera de los caminos marcados en la figura 3.1 para alcanzar el registro de imagen en un marco variacional. En este trabajo seguiremos el camino marcado por las “cajas” subrayadas.

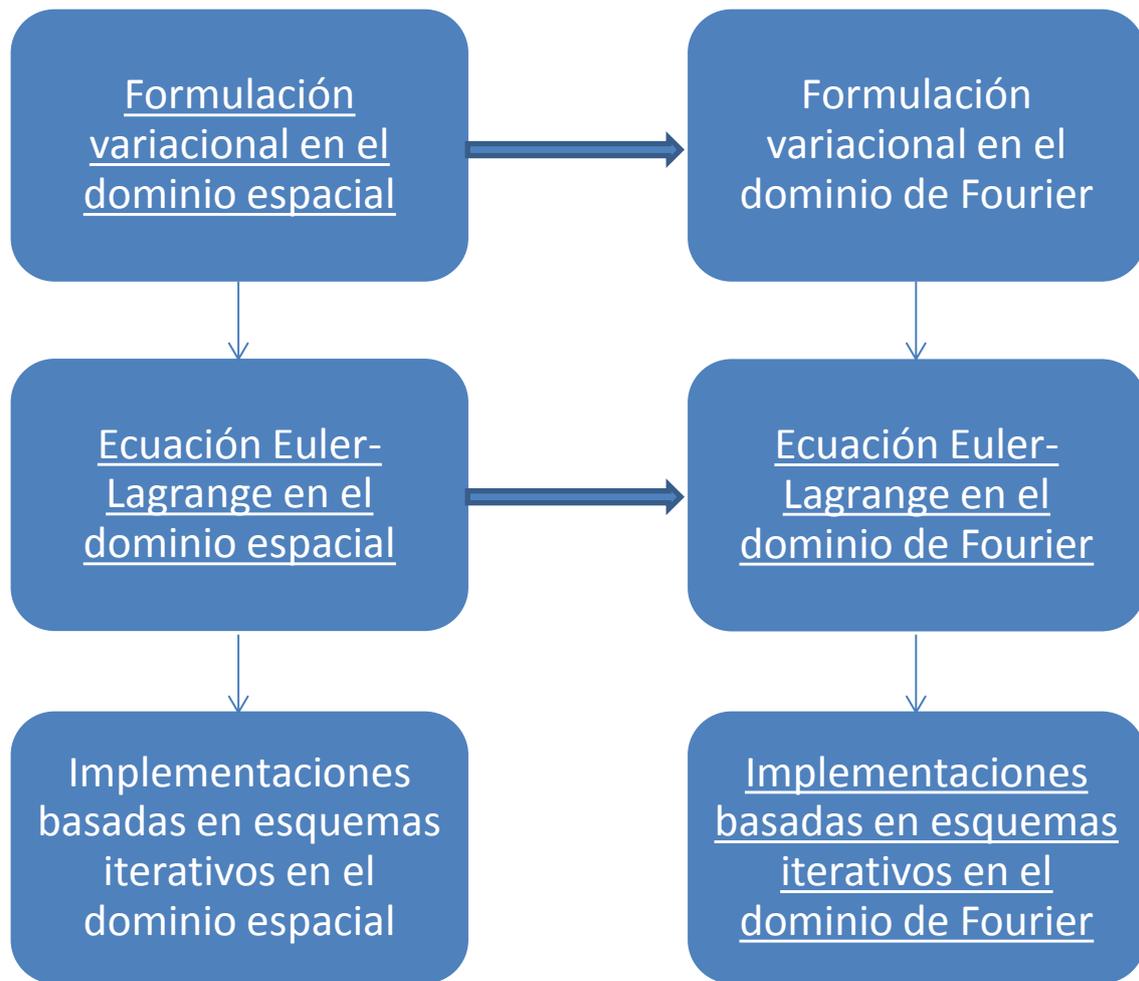


Figura 3.1: Bloques conceptuales del Capítulo 3

3.1. Enfoque variacional en el dominio espacial

Sean R y T dos conjuntos de datos, uno de referencia y otro objetivo, respectivamente, que representan el mismo objeto mediante la misma modalidad de imagen, o mediante modalidades de imágenes distintas. Se asume que estos conjuntos de datos pueden representarse mediante funciones soportadas de forma compacta en un dominio $\Psi :=]0,1[^d$ con contorno de Lipschitz $\partial\Psi$. El objetivo del registro de imagen es encontrar una transformación φ que haga que el conjunto de datos $T_\varphi(\mathbf{x}) := T(\varphi(\mathbf{x}))$ se corresponda geoméricamente con el conjunto de datos de referencia, es decir,

$$T_\varphi(\mathbf{x}) \approx R(\mathbf{x}). \quad (3.1)$$

Lo necesario para la resolución de dicha transformación es un término de regularización apropiado, que aporte un cierto conocimiento a priori sobre la transformación que se desea estimar, favoreciendo la obtención de las soluciones más verosímiles. Estos regularizadores dan lugar a las distintas técnicas de registro variacional encontradas en la literatura. En este trabajo nos centraremos en el registro *por difusión*.

Atendiendo a las consideraciones anteriores, el problema puede ser finalmente formulado de la siguiente manera: dados dos conjuntos de datos, R y T , y un parámetro de regularización positivo $\alpha > 0$, se desea encontrar una deformación \mathbf{u} tal que el siguiente funcional de energía conjunta:

$$J[\mathbf{u}] := \mathcal{D}[R, T; \mathbf{u}] + \alpha \mathcal{S}[\mathbf{u}], \quad (3.2)$$

sea mínimo, es decir,

$$\mathbf{u} = \min_{\mathbf{u}} J[\mathbf{u}], \quad (3.3)$$

siendo \mathcal{D} el término de energía que mide la disparidad (o distancia) entre los conjuntos de datos, y donde \mathcal{S} es el término de regularización, que penaliza la irregularidad de los vectores de correspondencia estimados; el parámetro de regularización, por su parte, es un escalar que pondera la influencia del término de regularización frente al término de distancia.

La elección de la energía de distancia \mathcal{D} depende en gran medida del dispositivo de adquisición de imágenes. Si los histogramas de los conjuntos de datos son similares (escenario de registro de imagen monomodal), típicamente se utiliza la suma de diferencias cuadráticas (SSD), que se define como

$$\mathcal{D}^{SSD}[R, T; \mathbf{u}] := \frac{1}{2} \|T_{\mathbf{u}} - R\|_{L_2(\Psi)}^2 = \frac{1}{2} \int_{\Psi} (T_{\mathbf{u}} - R)^2 dx, \quad (3.4)$$

siendo la ecuación del campo de fuerzas externas:

$$\mathbf{f}^{SSD}(\mathbf{x}; \mathbf{u}) = (R(\mathbf{x}) - T_{\mathbf{u}}(\mathbf{x})) \nabla T_{\mathbf{u}}(\mathbf{x}). \quad (3.5)$$

La medida de similitud que utilizaremos cuando se consideran imágenes provenientes de distintas fuentes o modalidades (escenario de registro de imagen multimodal) es el ratio de correlación (CR), cuya expresión analítica es:

$$\mathcal{D}^{CR}[R, T; \mathbf{u}] := -CR[R, T_{\mathbf{u}}] = -1 + \frac{E\{Var\{T_{\mathbf{u}}|R\}\}}{Var\{T_{\mathbf{u}}\}}, \quad (3.6)$$

siendo la ecuación del campo de fuerzas externas:

$$\mathbf{f}^{CR}(\mathbf{x}; \mathbf{u}) = \frac{1}{\mathcal{V}_{\Psi}} \left(\hat{G}_{\beta} * \frac{\partial L_{\mathbf{u}}^{CR}}{\partial i_t} \right) [R(\mathbf{x}), T_{\mathbf{u}}(\mathbf{x})] \nabla T_{\mathbf{u}}(\mathbf{x}), \quad (3.7)$$

donde

$$\frac{\partial L_{\mathbf{u}}^{CR}}{\partial i_t} = -\frac{2}{\text{Var}\{T_{\mathbf{u}}\}}(E\{T_{\mathbf{u}}\} - E\{T_{\mathbf{u}}|R\} + CR[R, T_{\mathbf{u}}](i_t - E\{T_{\mathbf{u}}\})). \quad (3.8)$$

Por su parte, el regularizador que se contempla en el presente trabajo puede expresarse en términos de una forma bilineal positiva $a[\mathbf{u}, \mathbf{u}] \in \mathbb{R}$, como se muestra en la siguiente ecuación:

$$\mathcal{S}[\mathbf{u}] := \frac{1}{2}a[\mathbf{u}, \mathbf{u}] = \frac{1}{2} \int_{\Psi} \langle \mathcal{B}[\mathbf{u}], \mathcal{B}[\mathbf{u}] \rangle_{\mathbb{R}^d} dx = \frac{1}{2} \int_{\Psi} \langle \mathcal{A}[\mathbf{u}], \mathbf{u} \rangle_{\mathbb{R}^d} dx, \quad (3.9)$$

De acuerdo al cálculo variacional, para que un desplazamiento \mathbf{u} satisfaga la ecuación, éste debe cumplir necesariamente que la primera variación de $\mathcal{J}[\mathbf{u}]$ en cualquier dirección arbitraria \mathbf{z} (es decir, que la derivada *Gâteaux* del funcional de energía conjunto) sea nulo, es decir,

$$\delta \mathcal{J}[\mathbf{u}; \mathbf{z}] = \delta \mathcal{D}[R, T; \mathbf{u}; \mathbf{z}] + \alpha \delta \mathcal{S}[\mathbf{u}; \mathbf{z}] := 0, \forall \mathbf{z} \in \mathbb{R}^d, \quad (3.10)$$

Para calcular la derivada *Gâteaux* de \mathcal{D} se utiliza la expresión que sigue:

$$\begin{aligned} \delta \mathcal{D}[R, T; \mathbf{u}; \mathbf{z}] &= \lim_{\epsilon \rightarrow 0} \frac{1}{\epsilon} (\mathcal{D}[R, T; \mathbf{u} + \epsilon \mathbf{z}] - \mathcal{D}[R, T; \mathbf{u}]) = \\ &= \left. \frac{\partial}{\partial \epsilon} \mathcal{D}[R, T; \mathbf{u} + \epsilon \mathbf{z}] \right|_{\epsilon=0} := \int_{\Psi} \langle \mathbf{f}(\mathbf{u}), \mathbf{z} \rangle dx, \end{aligned} \quad (3.11)$$

donde se ha asumido la existencia de una función $\mathbf{f} : \mathbb{R}^d \rightarrow \mathbb{R}^d$ que hace que se cumpla la última igualdad; a esta función se le conoce con el nombre de campo de fuerzas, y se encarga de representar las fuerzas externas que conducen el proceso de registro.

Por otro lado, aplicando a la definición de derivada *Gâteaux* a \mathcal{S} , obtenemos la siguiente igualdad:

$$\begin{aligned} \delta \mathcal{S}[\mathbf{u}; \mathbf{z}] &= \lim_{\epsilon \rightarrow 0} \frac{1}{\epsilon} (\mathcal{S}[\mathbf{u} + \epsilon \mathbf{z}] - \mathcal{S}[\mathbf{u}]) = \left. \frac{\partial}{\partial \epsilon} \mathcal{S}[\mathbf{u} + \epsilon \mathbf{z}] \right|_{\epsilon=0} = a[\mathbf{u}, \mathbf{z}] \\ &= \int_{\Psi} \langle \mathcal{A}[\mathbf{u}], \mathbf{z} \rangle dx, \end{aligned} \quad (3.12)$$

donde el operador en derivadas parciales \mathcal{A} puede entenderse como un campo de fuerzas que modela las restricciones internas (es decir, el conocimiento previo que se impone mediante el término de regularización), las cuales se oponen a las fuerzas externas.

Sustituyendo (3.11) y (3.12) en (3.10), se obtiene la ecuación en derivadas parciales que modela la situación de equilibrio de fuerzas – conocida como ecuación de Euler-Lagrange(E-L)–:

$$\mathbf{f}(x; \mathbf{u}) + \alpha \mathcal{A}[\mathbf{u}](x) = 0, \quad (3.13)$$

donde la notación empleada muestra de forma explícita la dependencia respecto a la variable espacial x .

Hay que tener en cuenta que la minimización (3.3) ha sido formulada sin considerar las condiciones de contorno explícitas para \mathbf{u} . Las condiciones de contorno implícitas del regularizador empleado surgirán de forma “natural”.

3.2. Registro por difusión

El registro por difusión fue presentado inicialmente por Fischer y Modersitzki 2002 [6]. El término difusión viene dado por la PDE de Euler-Lagrange asociada a la técnica que nos ocupa, ya que ésta puede verse como una ecuación de difusión del calor.

El regularizador del registro por difusión, que se muestra a continuación:

$$\mathcal{S}^{diff}[\mathbf{u}] = \frac{1}{2} a[\mathbf{u}, \mathbf{u}] := \frac{1}{2} \sum_{l=1}^d \int_{\Psi} \langle \nabla u_l, \nabla u_l \rangle_{\mathbb{R}^d} dx = \frac{1}{2} \sum_{l=1}^d \int_{\Psi} \|\nabla u_l\|^2 dx, \quad (3.14)$$

únicamente está motivado por el suavizado de la deformación, es decir, obtiene de forma preferente transformaciones suaves, mediante la minimización de las oscilaciones en las componentes del desplazamiento.

Resumiendo, cabe destacar que técnica de registro se basa en la minimización de la energía de las derivadas de primer orden de las componentes del desplazamiento, y por tanto el término de regularización no penalizará las traslaciones de tipo rígido.

El cálculo de la derivada *Gâteaux* de \mathcal{S}^{diff} proporciona el siguiente resultado:

$$\begin{aligned} \delta \mathcal{S}^{diff}[\mathbf{u}; \mathbf{z}] &= a[\mathbf{u}, \mathbf{z}] = \\ &= \sum_{l=1}^d \int_{\Psi} \langle \nabla u_l, \nabla z_l \rangle_{\mathbb{R}^d} dx = \sum_{l=1}^d \int_{\partial \Psi} z_l \langle \nabla u_l, \mathbf{b} \rangle_{\mathbb{R}^d} dx - \int_{\Psi} \langle \Delta \mathbf{u}, \mathbf{z} \rangle_{\mathbb{R}^d} dx, \end{aligned} \quad (3.15)$$

donde se ha empleado la primera identidad de Green para obtener la última igualdad.

Por último, se deduce la forma analítica del operador \mathcal{A}^{diff} de la ecuación en derivadas parciales de Euler-Lagrange:

$$\mathcal{A}^{diff}[\mathbf{u}](\mathbf{x}) = -\Delta \mathbf{u}(\mathbf{x}), \quad (3.16)$$

siendo la ecuación correspondiente en el dominio de Fourier:

$$\tilde{\mathcal{A}}_{ll}^{diff}(\boldsymbol{\omega}) = 2 \sum_{m=1}^d (1 - \cos \omega_m). \quad (3.17)$$

Nótese por último que en el caso $\tilde{\mathcal{A}}_{lm}^{diff} = \mathbf{0}$ (con $l \neq m$). En otras palabras, las componentes frecuenciales del desplazamiento están desacopladas, al igual que sucede en el dominio espacial.

3.3. Implementación frecuencial de las ecuaciones de Euler-Lagrange

Antes de centrarnos en la resolución de la ecuación de Euler-Lagrange, cabe destacar la completa necesidad de que se cumpla la siguiente ecuación:

$$\tilde{\mathbf{f}}(\boldsymbol{\omega}) + \alpha \tilde{\mathbf{A}}(\boldsymbol{\omega}) \tilde{\mathbf{u}}(\boldsymbol{\omega}) = \mathbf{0}. \quad (3.18)$$

Con objeto de resolver la ecuación de Euler-Lagrange formulada en el dominio de Fourier (ecuación 3.18), en esta sección se propone el uso de un esquema iterativo como los que habitualmente se emplean en el dominio espacial, centrándonos en este trabajo en los semi-implícitos de paso temporal. Partiendo de (3.18) la aplicación del esquema paso temporal resultaría en la siguiente expresión:

$$\tilde{\mathbf{u}}^{(\xi)}(\boldsymbol{\omega}) = (\mathbf{I} + \tau \alpha \tilde{\mathbf{A}}(\boldsymbol{\omega}))^{-1} \left(\tilde{\mathbf{u}}^{(\xi-1)}(\boldsymbol{\omega}) - \tau \tilde{\mathbf{f}}^{(\xi-1)}(\boldsymbol{\omega}) \right), \quad (3.19)$$

siendo \mathbf{I} una matriz identidad a bloques en el dominio $\boldsymbol{\Omega}$, y donde $\tilde{\mathbf{u}}^{(\xi)}$ se inicializa a cero, es decir, $\tilde{\mathbf{u}}^{(0)} = \mathbf{0}$. La inversión que aparece en la ecuación anterior se puede realizar sin considerar técnicas especiales.

La implementación eficiente que se deduce de (3.19) para cada una de las componentes espaciales discretizadas del campo de desplazamiento, viene dada por la siguiente expresión:

$$u_l^{(\xi)}[\mathbf{n}] = \text{IFFT} \left\{ \sum_{m=1}^d (\mathbf{I}_{lm} + \tau \alpha \tilde{\mathbf{A}}_{lm}[\mathbf{k}])^{-1} \odot \text{FFT} \left\{ u_l^{(\xi-1)}[\mathbf{n}] - \tau f_m^{(\xi-1)}[\mathbf{n}] \right\} \right\}, \quad (3.20)$$

donde la submatriz, o bloque, \mathbf{I}_{lm} , de tamaño $N_1 \times \dots \times N_d$, se define como

$$\mathbf{I}_{lm} = \begin{cases} \mathbf{1}, & l = m, \\ \mathbf{0}, & l \neq m. \end{cases} \quad (3.21)$$

4. Integración de correspondencias geométricas en un marco de registro de imagen variacional

En este capítulo nos centramos en la parte fundamental del proyecto. El objetivo principal es comprobar el funcionamiento de la integración de correspondencias geométricas (*landmarks*) en un marco de registro de imagen variacional. Para llevar a cabo este proceso utilizaremos dos métodos diferentes: el método de *resultante global* y el método de *parches locales*.

A lo largo de este capítulo explicaremos la obtención de las correspondencias geométricas, la formulación en general y cada uno de los métodos en particular, dejando la comparativa de resultados para el siguiente capítulo.

4.1. Detección de correspondencias geométricas

Como ya explicamos en el capítulo 2, para detectar las correspondencias geométricas utilizamos la función Surf, la cual nos devuelve el número de correspondencias indicado ordenadas de mayor a menor coincidencia.

Nuestra idea principal era usar únicamente Surf, sin embargo, en algunas imágenes monomodales (sobre todo en las de 3 dimensiones) y en todas las multimodales, esta función no nos proporcionaba las correspondencias adecuadamente, o directamente estas no eran influyentes. Debido a esto, decidimos poder detectar las correspondencias de forma manual mediante la función de Matlab “cpselct”.

A raíz de esto, hemos implementado el código de ambos métodos para que se puedan detectar los *landmarks* de las dos formas, o bien con la función Surf (siempre preferible) o bien de una manera manual. Hay que tener en cuenta que para una mayor velocidad de procesado, hemos detectado correspondencias geométricas únicamente antes de la primera iteración.

Por último, pondremos un ejemplo de cómo funciona la función SURF. Para ello utilizaremos una de las imágenes más empleadas en el registro de imagen, la fotografía de Lenna. En este caso hemos decidido obtener cinco correspondencias geométricas, como se muestra a continuación:



Figura 4.1: Detección de correspondencias geométricas mediante SURF

4.2. Formulación

En esta sección nos centraremos en la formulación que hemos empleado en este proyecto. Buscando en la literatura, la integración de correspondencias geométricas ya había sido formulada por Fischer y Modersitzki en [7] y también podemos encontrarla posteriormente en [9].

A continuación, introduciremos el término de energía que mide la disparidad (o distancia) entre las correspondencias geométricas detectadas \mathcal{D}^{LM} :

$$\mathcal{D}^{LM} = \frac{1}{2} \int_{\Psi} \sum_{j=1}^N \left(\mathbf{x}_{Rj} - \mathbf{u}(\mathbf{x}_{Tj}) \right)^2 dx, \quad (4.1)$$

donde \mathbf{x}_R y \mathbf{x}_T son las posiciones de las correspondencias geométricas en las imágenes referencia y objetivo, respectivamente, y N es el número total de correspondencias.

Tras esto, calcularemos la derivada parcial de \mathcal{D}^{LM} , para la obtención de la variable \mathbf{l} , la cual añadirá la resultante de las correspondencias geométricas a las fuerzas obtenidas en el capítulo anterior:

$$\begin{aligned} d\mathcal{D}^{LM} &= \frac{\partial}{\partial \epsilon} \mathcal{D}^{LM}(\mathbf{u} + \epsilon \mathbf{z}) \Big|_{\epsilon=0} = \frac{\partial}{\partial \epsilon} \frac{1}{2} \int_{\Psi} \sum_{j=1}^N \left(\mathbf{x}_{Rj} - \mathbf{u}(\mathbf{x}_{Tj}) - \epsilon \mathbf{z}(\mathbf{x}_{Tj}) \right)^2 dx \Big|_{\epsilon=0} \\ &= \frac{1}{2} \int_{\Psi} \sum_{j=1}^N 2 \left(\mathbf{x}_{Rj} - \mathbf{u}(\mathbf{x}_{Tj}) - \epsilon \mathbf{z}(\mathbf{x}_{Tj}) \right) \left(-\mathbf{z}(\mathbf{x}_{Tj}) \right) dx \Big|_{\epsilon=0} \\ &= \int_{\Psi} \sum_{j=1}^N \left(\mathbf{x}_{Rj} - \mathbf{u}(\mathbf{x}_{Tj}) \right) \left(-\mathbf{z}(\mathbf{x}_{Tj}) \right) dx = \int_{\Psi} \langle \mathbf{l}, \mathbf{z} \rangle dx \end{aligned} \quad (4.2)$$

Por lo tanto, de (4.2) podemos deducir por identificación, que:

$$\mathbf{l} = \sum_{j=1}^N \left(\mathbf{u}(\mathbf{x}_{Tj}) - \mathbf{x}_{Rj} \right) = \sum_{j=1}^N \left(\mathbf{x}_{Tu_j} - \mathbf{x}_{Rj} \right), \quad (4.3)$$

donde \mathbf{x}_{Tu} hace referencia a la posición de los *landmarks* en la imagen objetivo que vamos actualizando en cada iteración del registro.

Tras obtener la componente \mathbf{l} , nuestra componente final de fuerzas \mathbf{g} quedaría de la siguiente manera:

$$\mathbf{g}_l^\xi = \mathbf{f}_l^\xi + \beta \mathbf{l}_l^\xi = \mathbf{f}_l^\xi + \beta \sum_{j=1}^N \left(\mathbf{x}_{Tu_j}^\xi - \mathbf{x}_{Rj}^\xi \right), \quad (4.4)$$

donde \mathbf{g}_l^ξ es la combinación de ambas fuerzas (las de intensidad y las de distancia geométrica), β es el factor de escalado que evita que las fuerzas \mathbf{l}_l^ξ desplacen más de un pixel la imagen objetivo en cada iteración y \mathbf{l} es el campo de fuerzas externas que se deriva de la energía de disimilitud definida por la distancia euclídea de las correspondencias geométricas detectadas.

Por último, sólo tendríamos que hacer la transformada de Fourier para finalmente, tras un proceso de suavizado, obtener la solución de la ecuación de Euler-Lagrange:

$$\mathbf{G}_l^\xi = \mathcal{F} \{ \mathbf{g}_l^\xi \}, \quad (4.5)$$

$$u_l^{(\xi)}[\mathbf{n}] = \text{IFFT} \left\{ \sum_{m=1}^d (I_{lm} + \tau \alpha \tilde{\mathcal{A}}_{lm}[\mathbf{k}])^{-1} \odot \text{FFT} \{ u_l^{(\xi-1)}[\mathbf{n}] - \tau \mathbf{g}_l^{(\xi-1)}[\mathbf{n}] \} \right\}, \quad (4.6)$$

4.3. Método de resultante global

Este método, al cual hemos llamado “método de resultante global”, es el que podemos encontrar en la literatura (en [7], [8] y [9]). De todas maneras, la formulación que hemos empleado para llegar a este método sigue el mismo criterio que se emplea en [2].

El método se caracteriza porque el parámetro l es una constante, que sale de la resolución del sumatorio que podemos ver en (4.3).

Vamos a mostrar gráficamente el funcionamiento de este método, para ello utilizaremos la Figura 4.2:

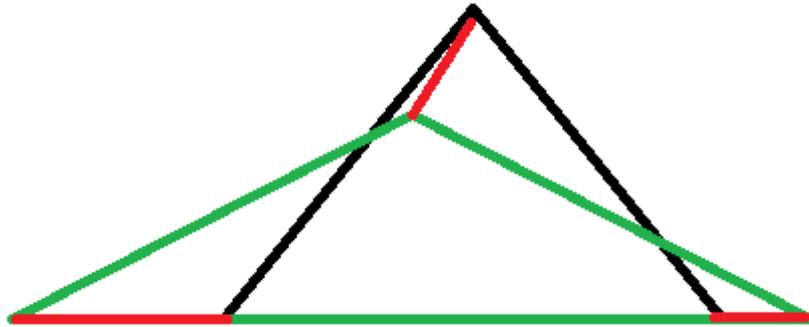


Figura 4.2: Ejemplo funcionamiento método resultante global (I)

En ella tenemos como imagen referencia (R) el triángulo negro, como imagen objetivo (T) el triángulo rojo, y las marcas rojas indican las fuerzas de desplazamiento que introducen las correspondencias geométricas (para simplificar la explicación hemos elegido los vértices como correspondencias).

Con estas tres correspondencias geométricas, obtendríamos una resultante global como la que podemos observar en la Figura 4.3. Como podemos observar en el vértice inferior derecho, en esta iteración, lejos de acercarse a su posición deseada se aleja de esta.

El principal problema que presenta este método es que al utilizar una constante para todas las correspondencias, lo más probable es que en muchos casos haya píxeles o vóxeles de la imagen objetivo que estemos alejando de la posición óptima en lugar de acercarlos. Claro, al trabajar con correspondencias geométricas fijas, estas pueden salirse de la matriz y eliminarse, lo que da lugar a error.

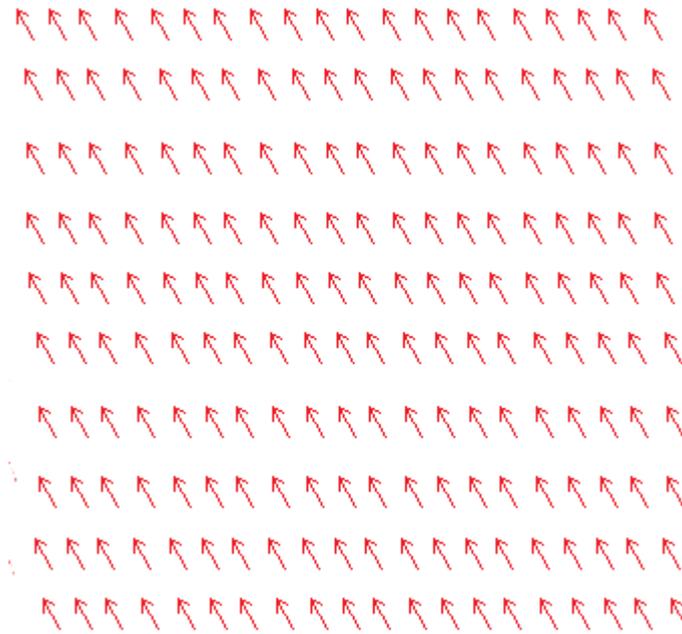


Figura 4.3: Ejemplo funcionamiento método resultante global (II)

Además como comprobaremos en el siguiente capítulo, este método presenta muchas oscilaciones antes de llegar a una solución, que en la mayoría de casos no será óptima.

4.4. Método de parches locales

Aunque matemáticamente está demostrado, a la hora de aplicarlo a imágenes reales el método de resultante global podría no ser muy efectivo. Debido a esto decidimos crear nuestro propio método de registro.

De esta manera se nos ocurrió que si aplicáramos unos parches locales a cada correspondencia geométrica, alcanzaríamos la solución de una manera más óptima, ya que estaríamos corrigiendo a la vez los N *landmarks*, y no sólo el que más peso tuviera.

Uno de los posibles inconvenientes que se nos plantearon era la posibilidad de que se “rompiera” la imagen, ya que estábamos aplicando desplazamientos en diferentes direcciones, pero al realizar al final un filtrado pasabajo, estos desplazamientos se ven suavizados, evitando así que la imagen se “rompiera”.

Una vez que teníamos esto claro, procedimos a la creación de los parches. Para ello decidimos que la mejor solución era que estos fueran unos rectángulos. El tamaño de estos lo marcan las correspondencias, de tal manera que si tenemos una correspondencia en la posición (4,7) en la imagen R y (2,5) en la imagen T_w , el parche local de dicha correspondencia será el rectángulo (2:4,5:7).

Para explicar esto de una manera gráfica, si partimos de la Figura 4.2, la resultante obtenida con el método de parches locales quedaría como se muestra en la Figura 4.4:



Figura 4.4: Ejemplo funcionamiento método parches locales

Por lo tanto, en el método de parches locales, la componente l será una matriz del tamaño de la imagen, a diferencia que en el otro método. Esta matriz estará vacía exceptuando los parches locales (Figura4.4). Además, cabe resaltar que en el caso de que dos o más parches se solaparan se respeta siempre el primer parche que se haya introducido en la matriz, es decir, el parche que corresponde al *landmark* con mayor importancia.

Como habíamos previsto y como mostraremos en el siguiente capítulo, este método mejora al método de resultante global.

5. Resultados

En esta sección presentaremos los principales resultados obtenidos con la integración de características geométricas en un marco de registro de imagen variacional. A su vez, realizaremos una comparación de los mismos con los obtenidos con un método de registro variacional de imagen.

Básicamente, procederemos a comparar los resultados obtenidos con el método presentado en este proyecto, con los resultados obtenidos con el método de [2]. Además, también utilizaremos esta sección para comprobar qué método de los expuestos en este proyecto proporciona mejores resultados.

5.1. Resultados 2D

En este apartado se exponen resultados obtenidos sobre varios pares de imágenes, es decir $d=2$, procedentes de diferentes escenarios de aplicación, aunque haciendo de nuevo especial énfasis en el ámbito de la imagen médica. Dividiremos la sección en cuatro apartados, primero expondremos los resultados obtenidos con cada método. Al final haremos una comparativa de los resultados obtenidos con ambos métodos, al igual que con el método explicado en el capítulo 3.

5.1.1. Método de resultante global

Para comenzar este apartado utilizaremos una de las imágenes más utilizadas en la literatura del registro de imagen, la fotografía de Lenna. Para localizar las correspondencias geométricas hemos utilizado la función Surf obteniendo el siguiente resultado:

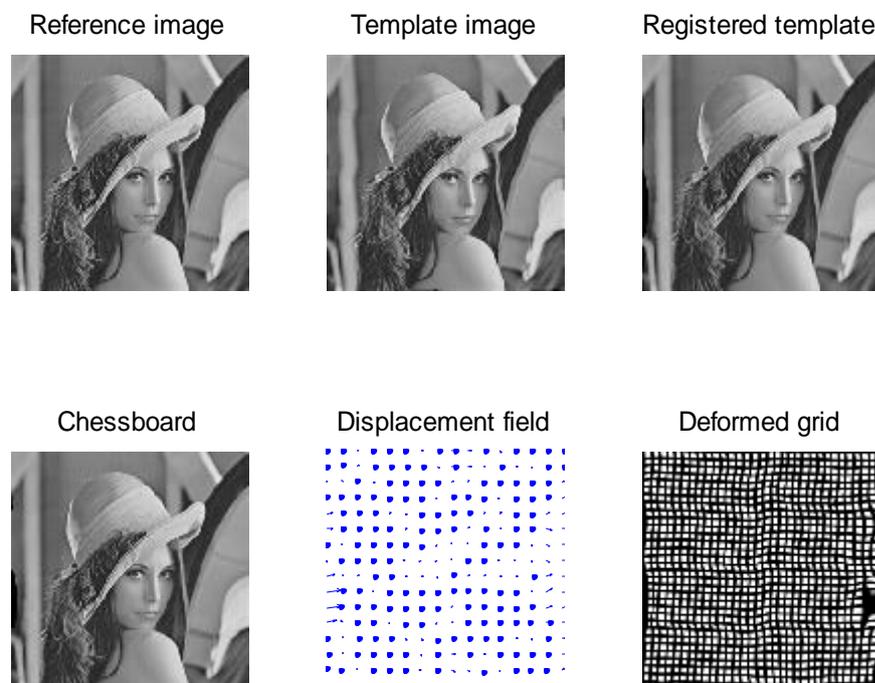


Figura 5.1: Registro imagen Lenna mediante método de resultante global

Se alcanza la solución óptima con un PSNR=25.97 dB en la iteración $\xi=17$. Hemos utilizado los siguientes valores para los parámetros: $\alpha=10$, $\tau=1$ y $\sigma=1$.

A continuación, utilizaremos una imagen creada por nosotros para comprobar si realmente la integración de correspondencias geométricas podría ayudar a realizar un registro que únicamente con un marco variacional no se puede realizar:

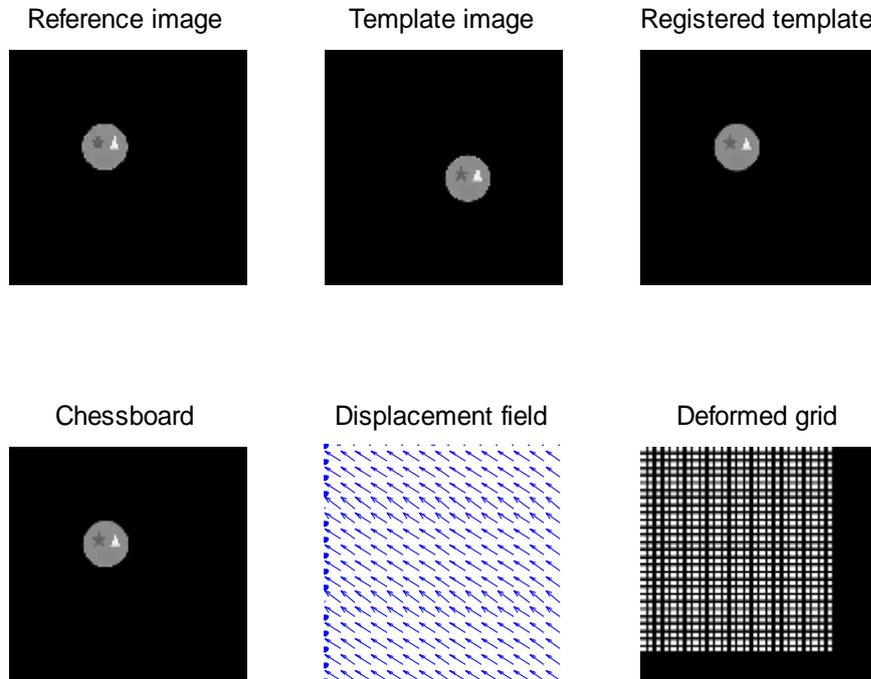


Figura 5.2: Registro imagen creada mediante método de resultante global

Se alcanza la solución óptima con un PSNR=33.54 dB en la iteración $\xi=27$. Hemos utilizado los siguientes valores para los parámetros: $\alpha=10000$, $\tau=1$ y $\sigma=1$.

Tras las dos primeras imágenes, procederemos al empleo de imágenes médicas. Comenzaremos con el corte de una caja torácica (WB128corte). Al tratarse de una imagen multimodal haremos uso de la función “cpselect” para detectar correspondencias geométricas. El resultado obtenido se muestra en la Figura 5.3.

Con esta imagen se alcanza la solución con un CR=85.41% en la iteración $\xi=715$. Hemos utilizado los siguientes valores para los parámetros: $\alpha=100$, $\tau=1$ y $\sigma=1$.

Ahora pasaremos a usar una imagen craneal monomodal. A pesar de esto la función SURF no funciona correctamente con estas imágenes, por lo que tenemos que utilizar la función “cpselect”. La solución obtenida se muestra en la Figura 5.4.

Se alcanza la solución con un PSNR=25.41 dB en la iteración $\xi=300$. Hemos utilizado los siguientes valores para los parámetros: $\alpha=100$, $\tau=1$ y $\sigma=1$.

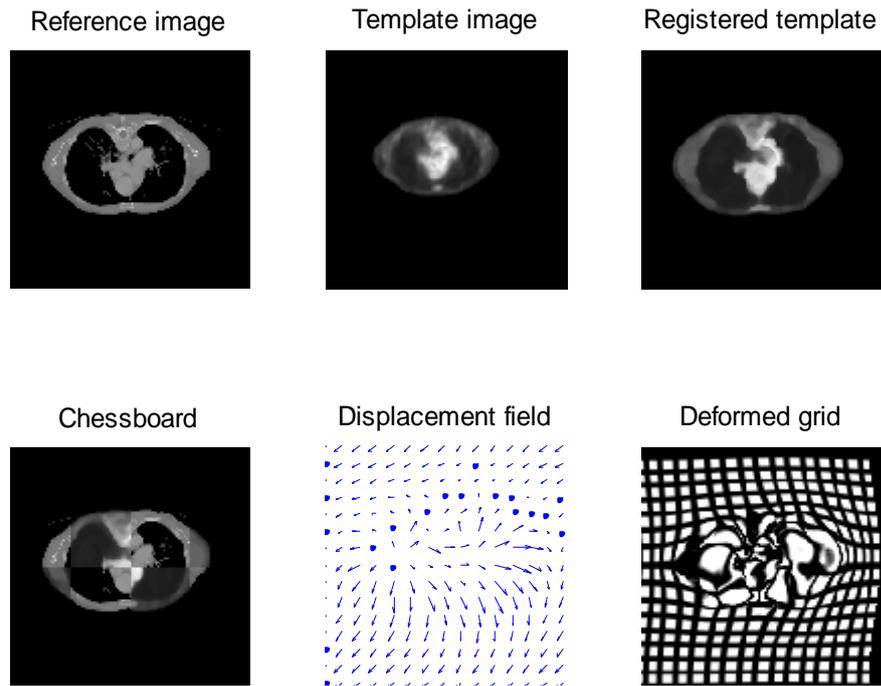


Figura 5.3: Registro imagen WB128corte mediante método de resultante global

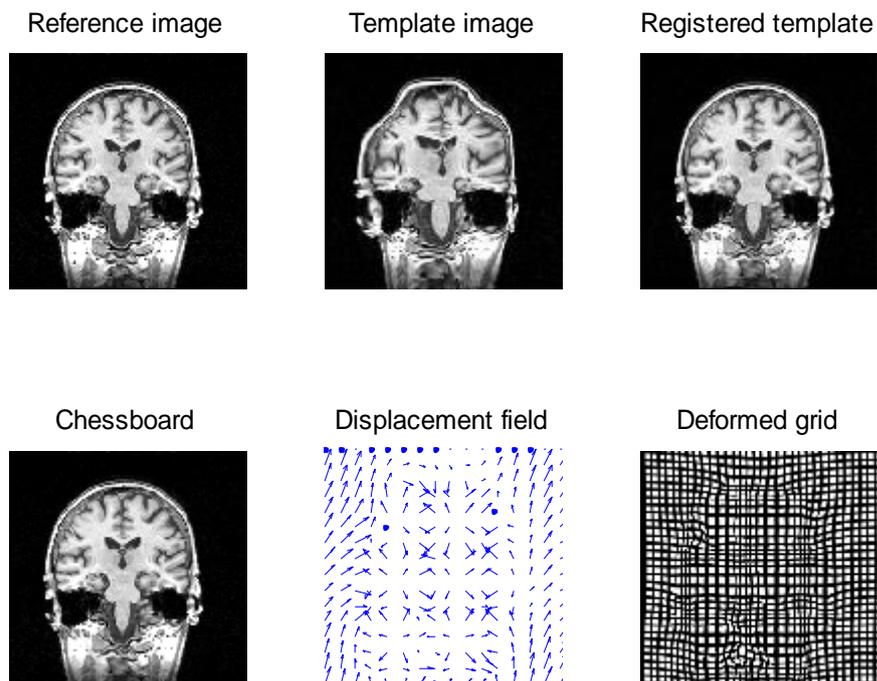


Figura 5.4: Registro imagen WBA3s mediante método de resultante global

A continuación procederemos al registro de un TAC, para el cual recurriremos otra vez a la función “cpselect” debido a que es un conjunto multimodal, obteniendo:

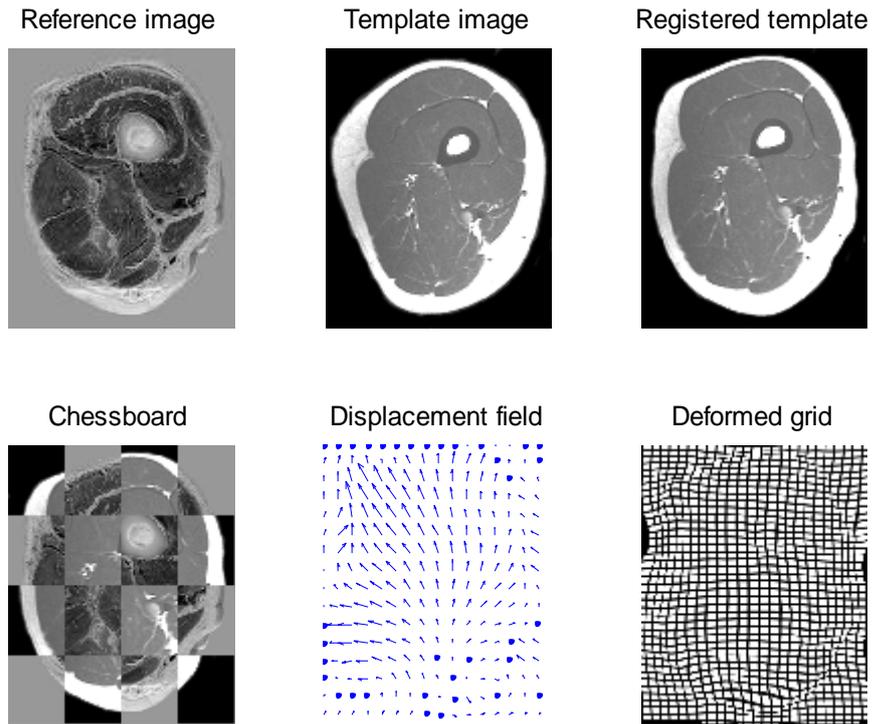


Figura 5.5: Registro imagen TAC_atlas mediante método de resultante global

Se alcanza la solución óptima con un CR=79.12% en la iteración $\xi=225$. Hemos utilizado los siguientes valores para los parámetros: $\alpha=350$, $\tau=1$ y $\sigma=1$.

5.1.2. Método de parches locales

Empezaremos, al igual que en el apartado anterior, con la imagen de Lenna. Podemos ver el resultado obtenido en la Figura 5.6.

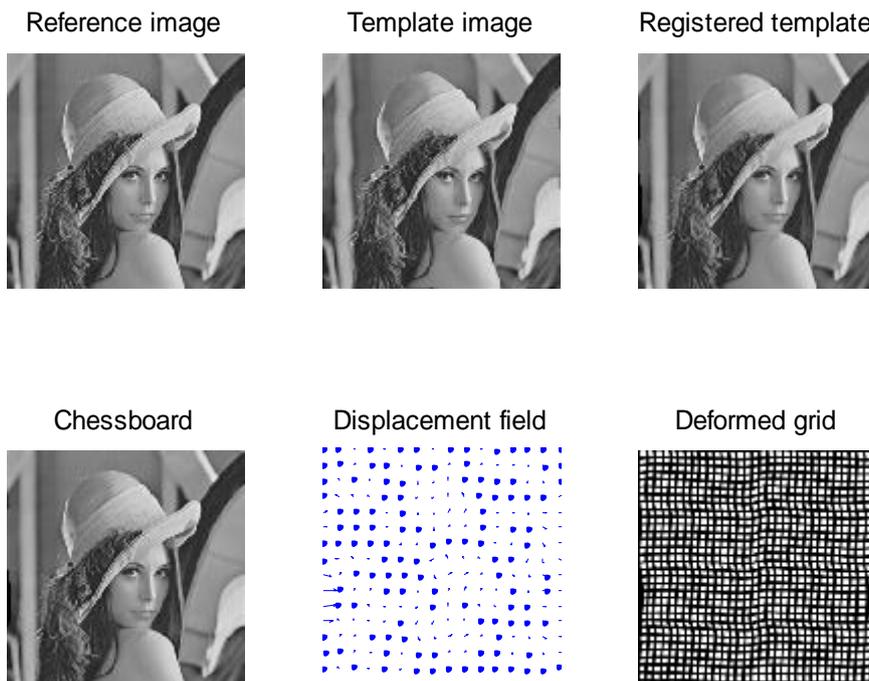


Figura 5.6: Registro imagen Lenna mediante método de parches locales

Se alcanza la solución óptima con un PSNR=26.40 dB en la iteración $\xi=32$. Hemos utilizado los siguientes valores para los parámetros: $\alpha=10$, $\tau=1$ y $\sigma=1$.

Ahora utilizaremos la imagen creada por nosotros. Tras la simulación correspondiente, los resultados se muestran en la Figura 5.7.

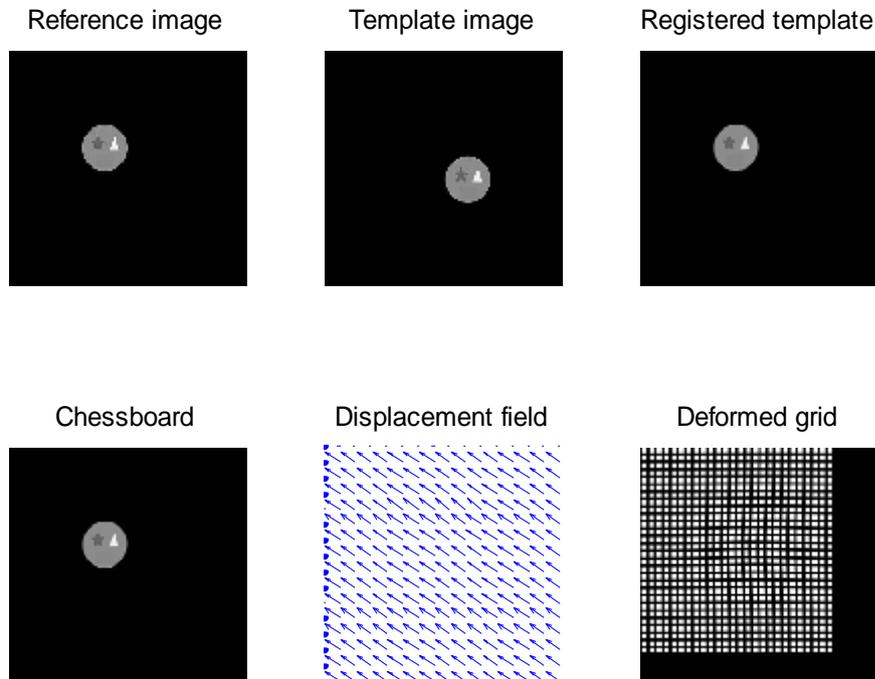


Figura 5.7: Registro imagen creada mediante método de parches locales

En este caso se alcanza la solución óptima con un PSNR=37.89 dB en la iteración $\xi=45$. Hemos utilizado los siguientes valores para los parámetros: $\alpha=10000$, $\tau=1$ y $\sigma=1$.

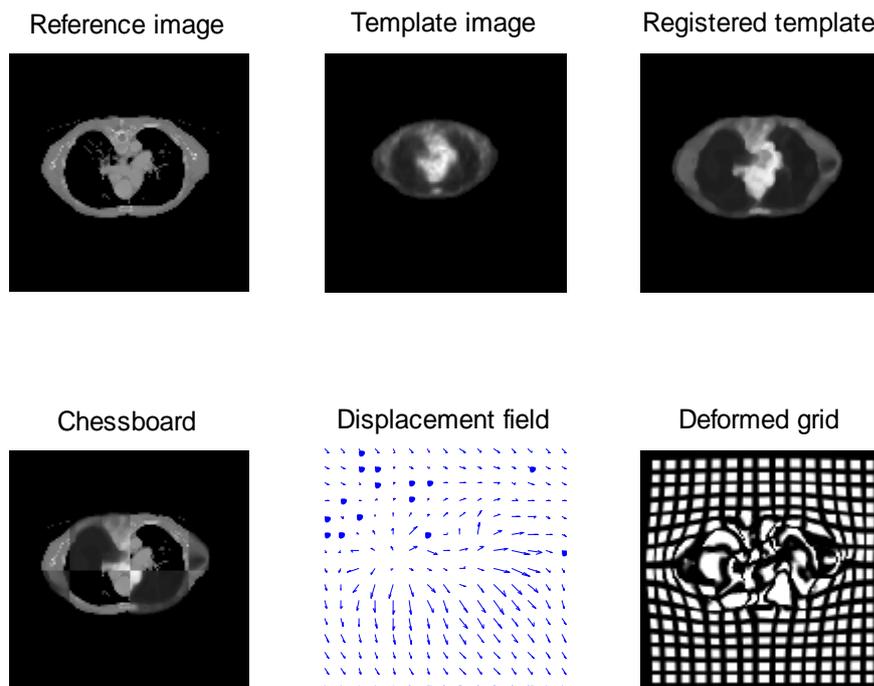


Figura 5.8: Registro imagen WB128corte mediante método de parches locales

Aplicamos ahora el método de parches locales a la imagen WB128corte, cuya simulación podemos ver en la Figura 5.8.

Se alcanza la solución con un CR=84.45% en la iteración $\xi=564$. Hemos utilizado los siguientes valores para los parámetros: $\alpha=100$, $\tau=1$ y $\sigma=1$.

Ahora utilizaremos la imagen craneal, que la llamamos WBA3s, tras la cual obtenemos:

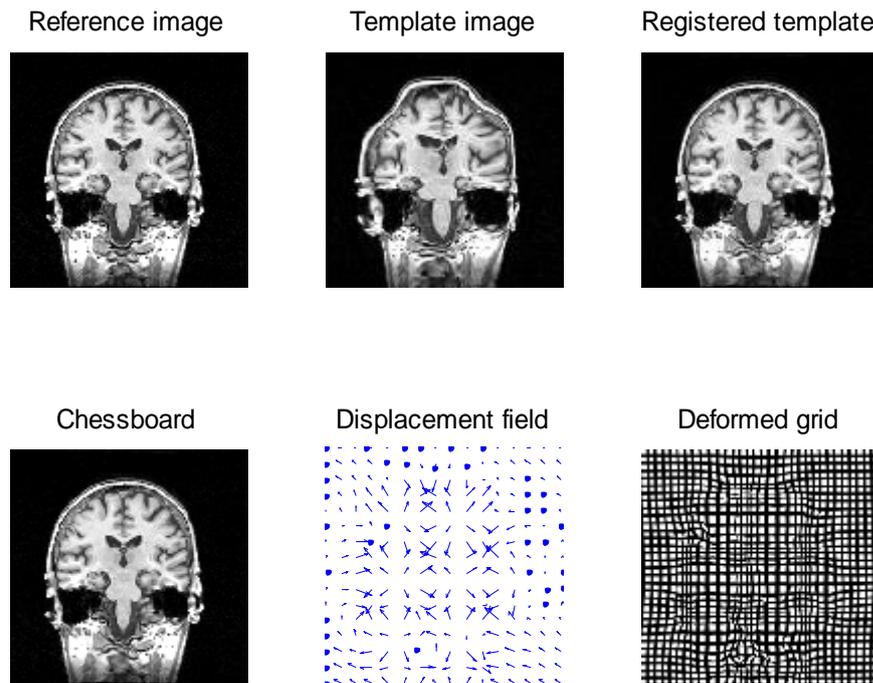


Figura 5.9: Registro imagen WBA3s mediante método de parches locales

Se alcanza la solución con un PSNR=25.41 dB en la iteración $\xi=300$. Hemos utilizado los siguientes valores para los parámetros: $\alpha=100$, $\tau=1$ y $\sigma=1$.

Con las imágenes del atlas obtenemos al aplicar el método de parches locales. Los resultados obtenidos con este método los encontramos en la Figura 5.10.

Se alcanza la solución óptima con un CR=84.21% en la iteración $\xi=460$. Hemos utilizado los siguientes valores para los parámetros: $\alpha=350$, $\tau=1$ y $\sigma=1$.

5.1.3. Registro con marco variacional

En este apartado comprobaremos los resultados obtenidos sin la integración de correspondencias geométricas en el marco variacional del registro de imagen.

Empezaremos con la imagen de Lenna. El resultado obtenido lo mostramos en la Figura 5.11.

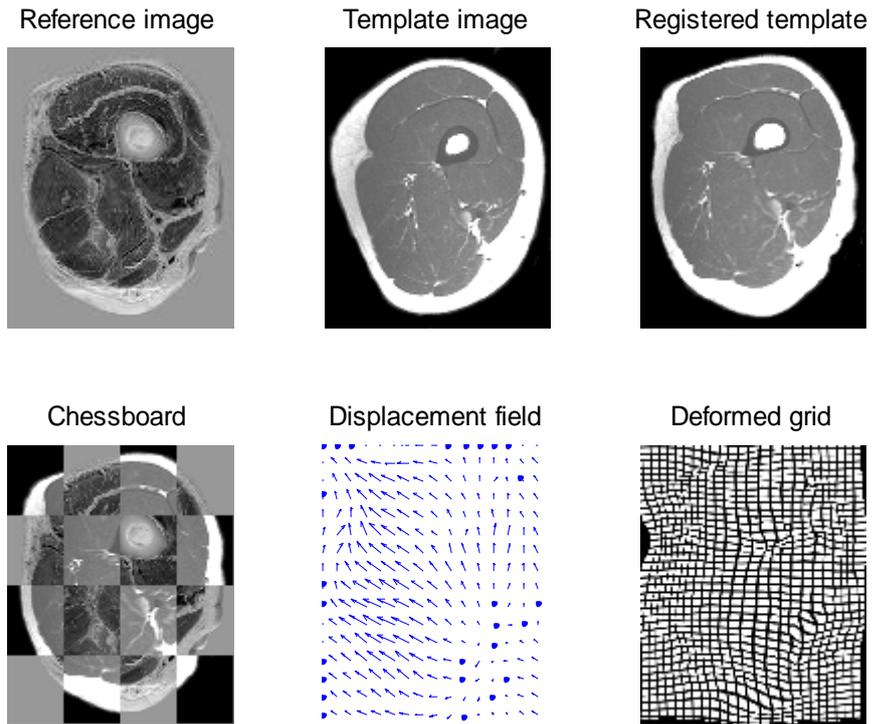


Figura 5.10: Registro imagen TAC_atlas mediante método de parches locales

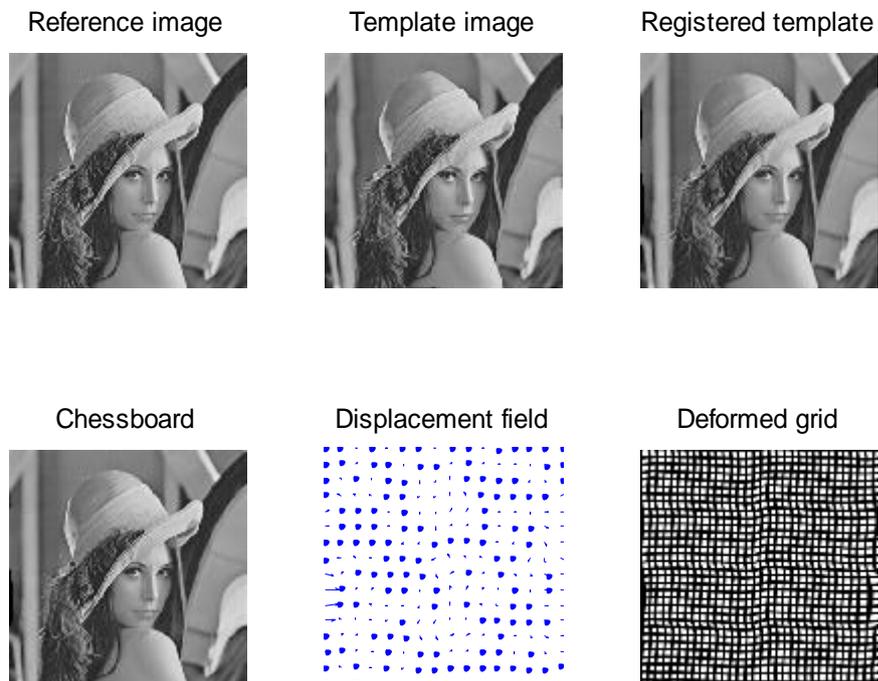


Figura 5.11: Registro imagen Lenna en un marco variacional

Se alcanza la solución óptima con un PSNR=26.39 dB en la iteración $\xi=32$. Hemos utilizado los siguientes valores para los parámetros: $\alpha=10$, $\tau=1$ y $\sigma=1$.

Ahora utilizaremos la imagen creada por nosotros, obteniendo:

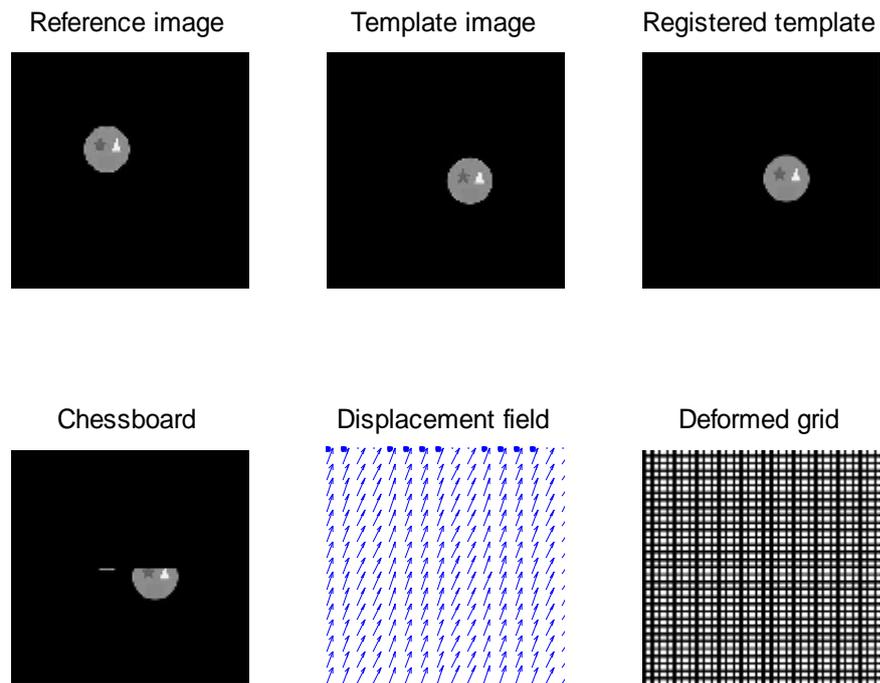


Figura 5.12: Registro imagen creada en un marco variacional

Se alcanza la solución óptima con un PSNR=17.36 dB en la iteración $\xi=100$. Hemos utilizado los siguientes valores para los parámetros: $\alpha=10000$, $\tau=1$ y $\sigma=1$.

Aplicamos ahora el método de parches locales a la imagen WB128corte:

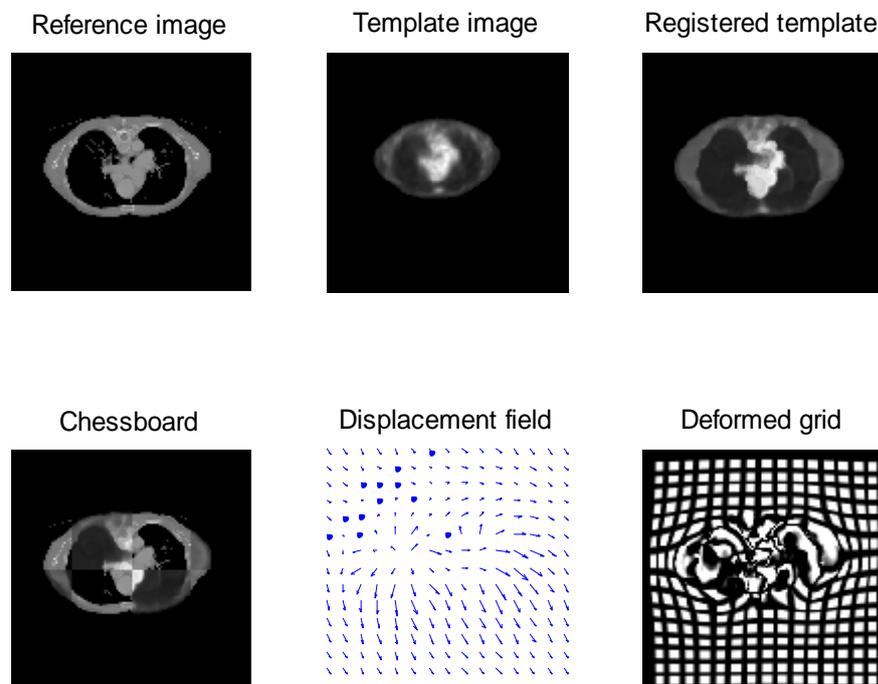


Figura 5.13: Registro imagen WB128corte en un marco variacional

Se alcanza la solución con un CR=84.42% en la iteración $\xi=1000$. Hemos utilizado los siguientes valores para los parámetros: $\alpha=100$, $\tau=1$ y $\sigma=1$.

Ahora utilizaremos la imagen craneal, que la llamamos WBA3s, tras la cual obtenemos:

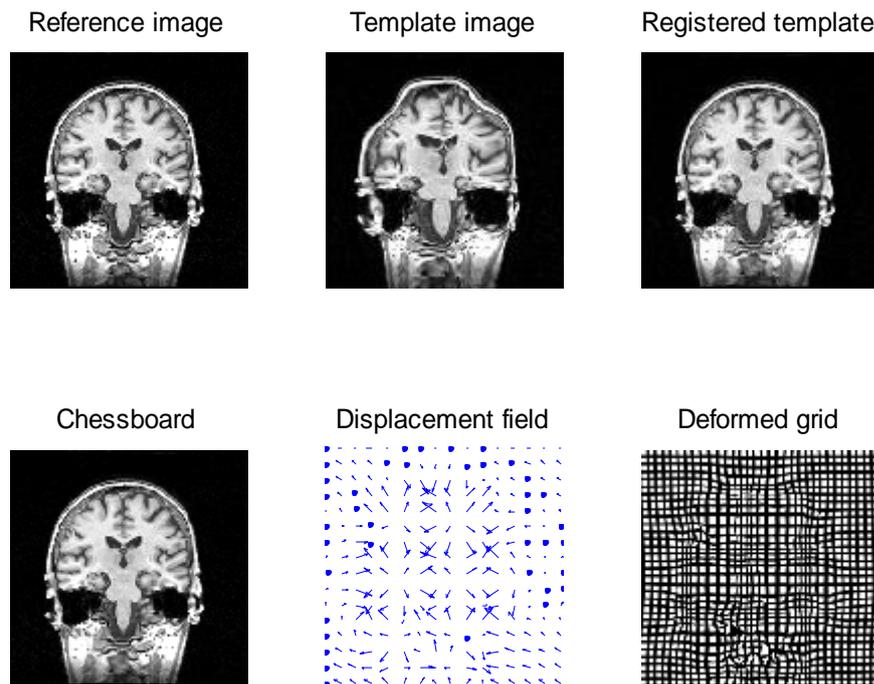


Figura 5.14: Registro imagen WBA3s en un marco variacional

Se alcanza la solución con un PSNR=24.65 dB en la iteración $\xi=300$. Hemos utilizado los siguientes valores para los parámetros: $\alpha=100$, $\tau=1$ y $\sigma=1$.

Con las imágenes del atlas obtenemos al aplicar un registro en un marco variacional:

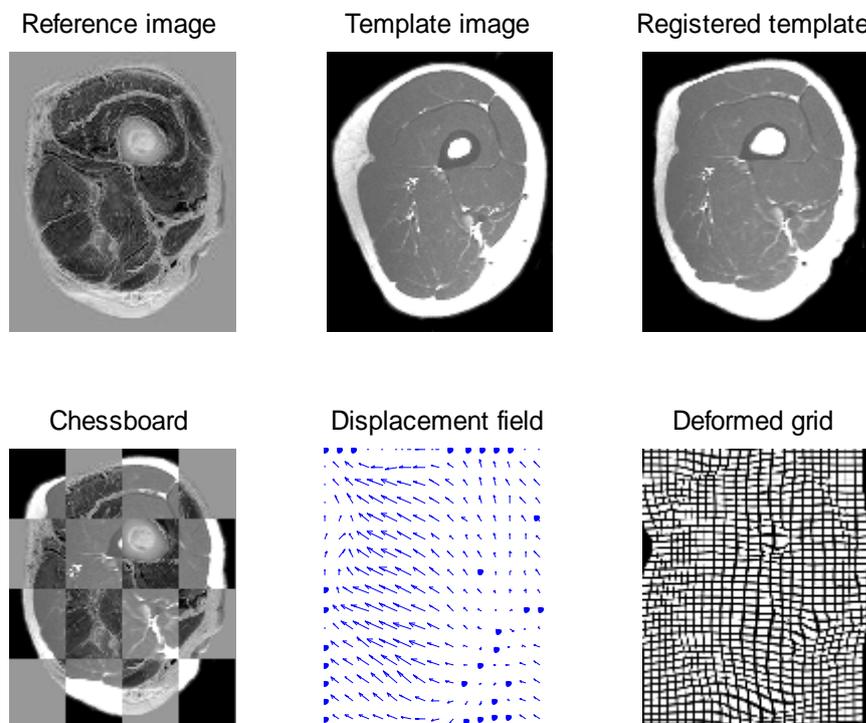


Figura 5.15: Registro imagen TAC_atlas en un marco variacional

Se alcanza la solución óptima con un CR=84.36% en la iteración $\xi=500$. Hemos utilizado los siguientes valores para los parámetros: $\alpha=350$, $\tau=1$ y $\sigma=1$.

5.1.4. Comparación de resultados

En este apartado reuniremos todos los resultados obtenidos con los métodos anteriores para cada conjunto de imágenes en unas tablas para poder comparar los resultados de una forma más simple y cómoda.

Como vamos a seguir el orden utilizado en los apartados anteriores, empezaremos con la imagen de Lenna:

| Método | Parámetro α | Iteraciones ξ | PSNR (dB) |
|-------------------|--------------------|-------------------|-----------|
| Resultante Global | 10 | 17 | 25.97 |
| Parches locales | 10 | 32 | 26.40 |
| Variacional | 10 | 32 | 26.39 |

Tabla 5.1: Comparación resultado imagen Lenna

Se puede apreciar que al ser una imagen que no requiere un registro previo, no hay apenas diferencias entre el método de parches locales y el registro variacional. Sin embargo se puede ver como el método de resultante global no alcanza el mismo nivel de registro.

A continuación mostraremos de la comparación de la imagen creada por nosotros:

| Método | Parámetro α | Iteraciones ξ | PSNR (dB) |
|-------------------|--------------------|-------------------|-----------|
| Resultante Global | 10.000 | 27 | 33.54 |
| Parches locales | 10.000 | 45 | 37.89 |
| Variacional | 10.000 | 100 | 17.36 |

Tabla 5.2: Comparación resultados imagen creada

En esta ocasión hemos elegido un parámetro de restricción α bastante elevado para evitar que se produzcan deformaciones en la imagen. En este ejemplo se puede apreciar claramente como la introducción de *landmarks* ayuda a poder realizar el registro.

Se puede apreciar como el registro en un marco variacional no se puede producir, ya que, necesitaría un pre registro. Sin embargo al introducir correspondencias geométricas se puede conseguir un registro muy satisfactorio. En esta ejemplo se puede apreciar también cómo el método de parches locales obtiene resultados mucho mejores que el método de resultante global, con una diferencia final superior a los 4 dB.

Empezando con las imágenes médicas, primero compararemos los resultados obtenidos para WB128corte:

| Método | Parámetro α | Iteraciones ξ | CR (%) |
|-------------------|--------------------|-------------------|--------|
| Resultante Global | 100 | 715 | 85.41 |
| Parches locales | 100 | 564 | 84.45 |
| Variacional | 100 | 1000 | 84.42 |

Tabla 5.3: Comparación resultados imagen WB120corte

En este ejemplo podemos ver cómo se alcanza el registro de una manera más rápida con la integración de correspondencias. Tras 500 iteraciones, sin correspondencias geométricas, el *Ratio de Correlación* era de 75.32 %.

Ahora compararemos los resultados de la imagen craneal WBA3s:

| Método | Parámetro α | Iteraciones ξ | PSNR (dB) |
|-------------------|--------------------|-------------------|-----------|
| Resultante Global | 100 | 300 | 25.41 |
| Parches locales | 100 | 300 | 25.41 |
| Variacional | 100 | 300 | 24.65 |

Tabla 5.4: Comparación resultados imagen WBA3s

En este caso se puede comprobar también que la ayuda que proporcionan las correspondencias geométricas conlleva a un mejor registro.

Para terminar reuniremos los datos obtenidos del registro del TAC de la imagen correspondiente al atlas:

| Método | Parámetro α | Iteraciones ξ | CR (%) |
|-------------------|--------------------|-------------------|--------|
| Resultante Global | 350 | 225 | 79.12 |
| Parches locales | 100 | 460 | 84.21 |
| Variacional | 100 | 500 | 84.36 |

Tabla 5.5: Comparación resultados imagen TAC_atlas

En esta última tabla se puede apreciar que apenas hay diferencia entre introducir o no correspondencias geométricas. Es más, que no haya mejora se puede deber a que al tener que introducirlas manualmente haya podido haber algún fallo de precisión. Sin embargo, lo que queda claro es que el método de parches locales proporciona mejores resultados que el método de resultante global.

5.2. Resultados 3D

Ahora procederemos a exponer los resultados obtenidos sobre varios conjuntos de datos volumétricos, con $d=3$, procedentes a su vez de varios escenarios de aplicación, aunque haciendo de nuevo especial énfasis en el ámbito de la imagen médica. Al igual que en el caso de 2 dimensiones, dividiremos la sección en cuatro apartados, exponiendo los resultados obtenidos con cada método y luego haciendo una comparativa de los resultados obtenidos con ambos métodos, al igual que con el método explicado en el capítulo 3.

Cabe destacar que al no disponer de la función SURF adaptada para ser usada en 3 dimensiones, hemos tenido que introducir manualmente todas las correspondencias mediante la función de MATLAB "cpselect".

5.2.1. Método de resultante global

Comenzaremos utilizando el volumen WB128 del cual habíamos utilizado un corte en la sección anterior.

Debido a la inestabilidad de este método, en una de las oscilaciones saca las correspondencias seleccionadas en un principio de la matriz, por lo que no se puede llegar a alcanzar ningún registro.

A continuación hemos utilizado una imagen médica de un hígado para la cual hemos obtenido:

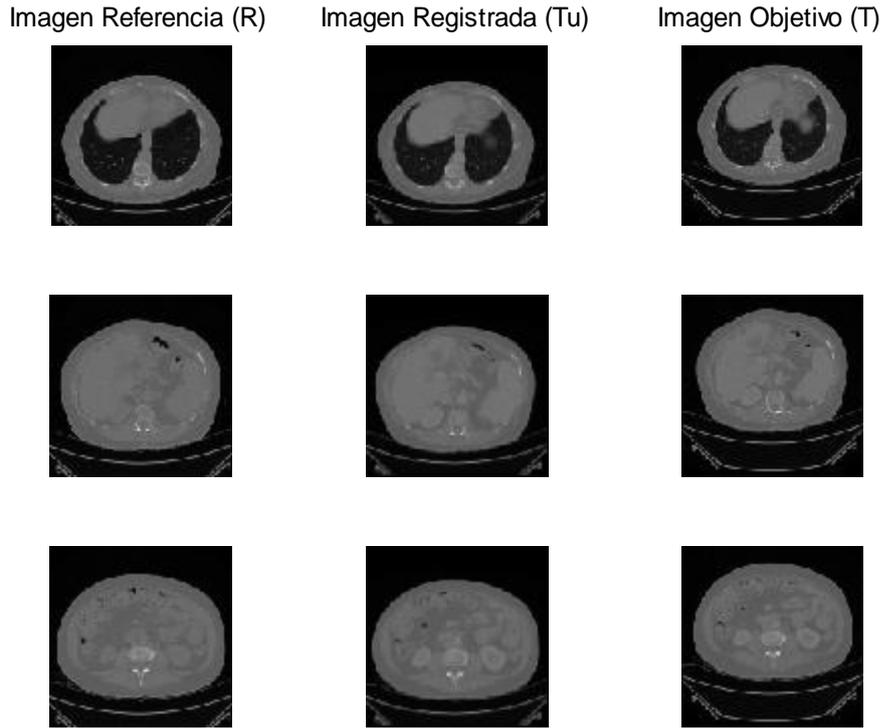


Figura 5.16: Registro volumen Hígado mediante método de resultante global

Se alcanza la solución óptima con un $CR=80.48\%$ en la iteración $\xi=439$. Hemos utilizado los siguientes valores para los parámetros: $\alpha=850$, $\tau=1$ y $\sigma=1$.

En el siguiente ejemplo, hemos procedido a registrar un volumen médico de una resonancia magnética con otro al que hemos denominado BOLD, obteniendo:

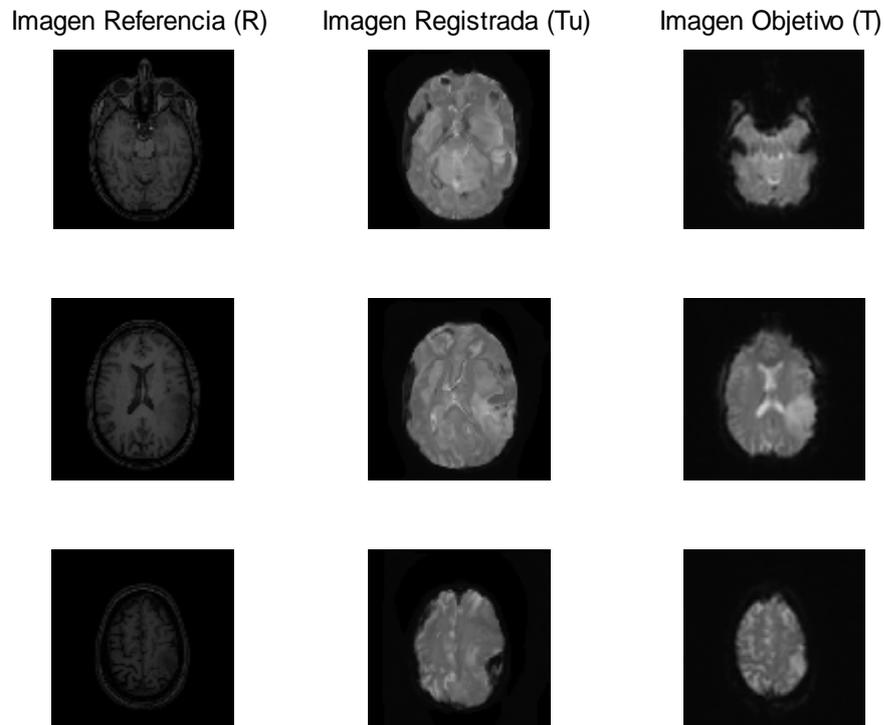


Figura 5.17: Registro volumen BOLD mediante método de resultante global

Se alcanza la solución óptima con un CR=78.43% en la iteración $\xi=130$. Hemos utilizado los siguientes valores para los parámetros: $\alpha=20$, $\tau=1$ y $\sigma=1$.

Ahora registraremos el mismo volumen de la resonancia magnética con otro volumen de un PET. Debido a lo mismo que ha sucedido con WB128, estos volúmenes no pueden registrarse con este método.

Por último registraremos los volúmenes de un paciente, pero tampoco podrá realizarse un registro adecuado con este método.

5.2.2. Método de parches locales

Al igual que en el método anterior comenzaremos utilizando el volumen WB128. Tras el registro obtenemos:

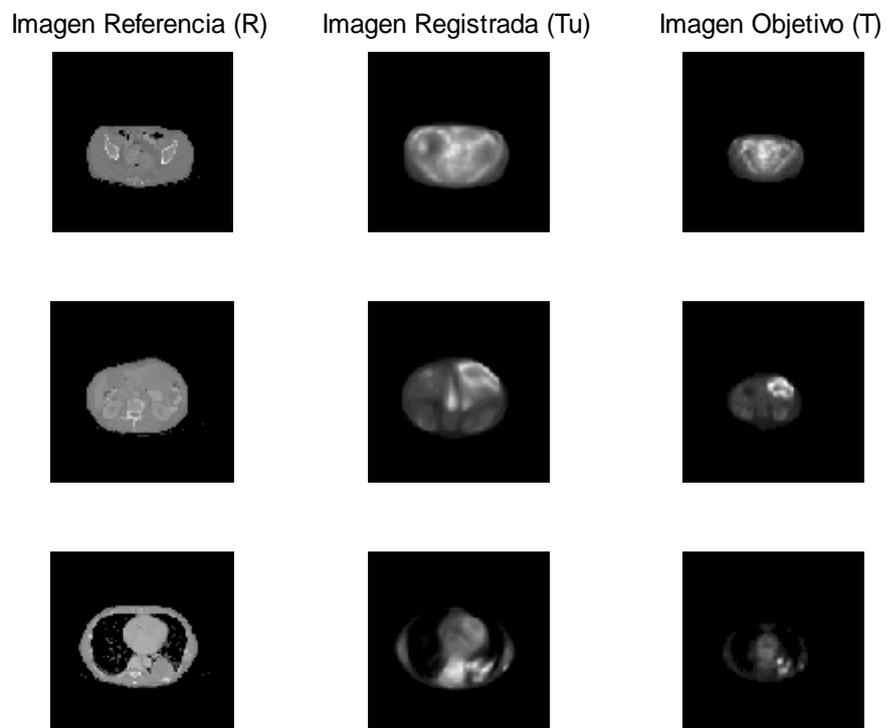


Figura 5.18: Registro volumen WB128 mediante método de parches locales

Se alcanza la solución óptima con un CR=70.67% en la iteración $\xi=2000$. Hemos utilizado los siguientes valores para los parámetros: $\alpha=1000$, $\tau=1$ y $\sigma=1$.

Ahora vamos a registrar el volumen del hígado, obteniendo lo mostrado en la Figura 5.19.

Se alcanza la solución óptima con un CR=79.59 % en la iteración $\xi=489$. Hemos utilizado los siguientes valores para los parámetros: $\alpha=850$, $\tau=1$ y $\sigma=1$.

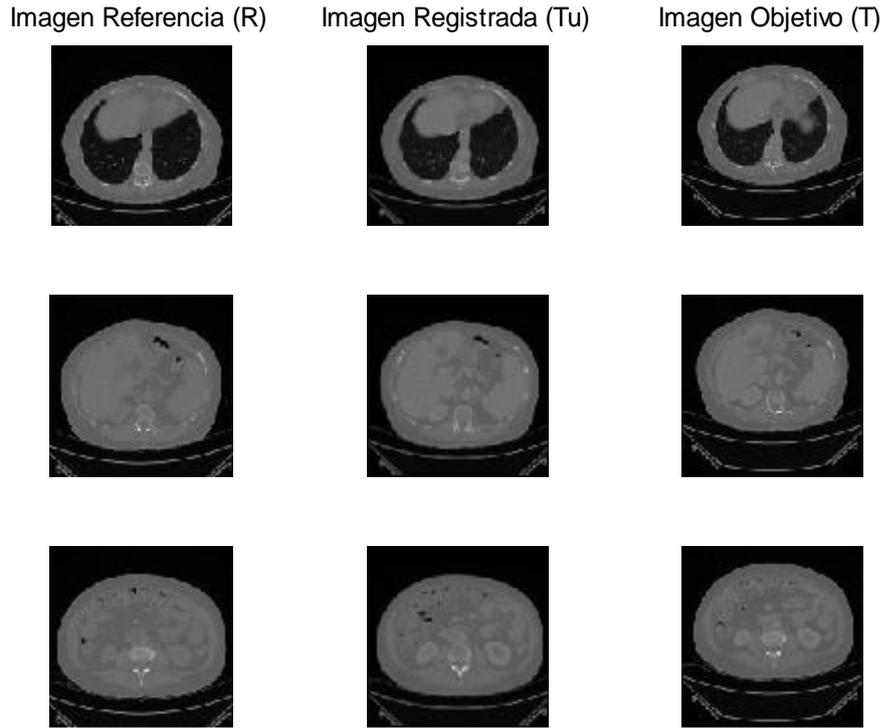


Figura 5.19: Registro volumen Hígado mediante método de parches locales

Ahora procedemos a registrar el volumen denominado BOLD:

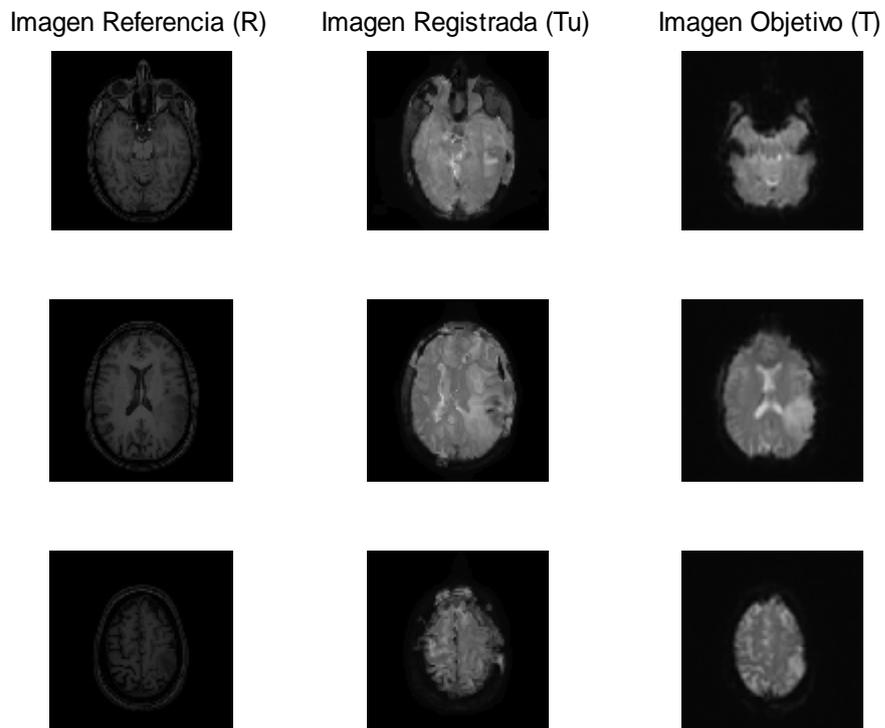


Figura 5.20: Registro volumen BOLD mediante método de parches locales

Se alcanza la solución óptima con un CR=82.54% en la iteración $\xi=150$. Hemos utilizado los siguientes valores para los parámetros: $\alpha=20$, $\tau=1$ y $\sigma=1$.

Ahora vamos a registrar el volumen denominado PET:

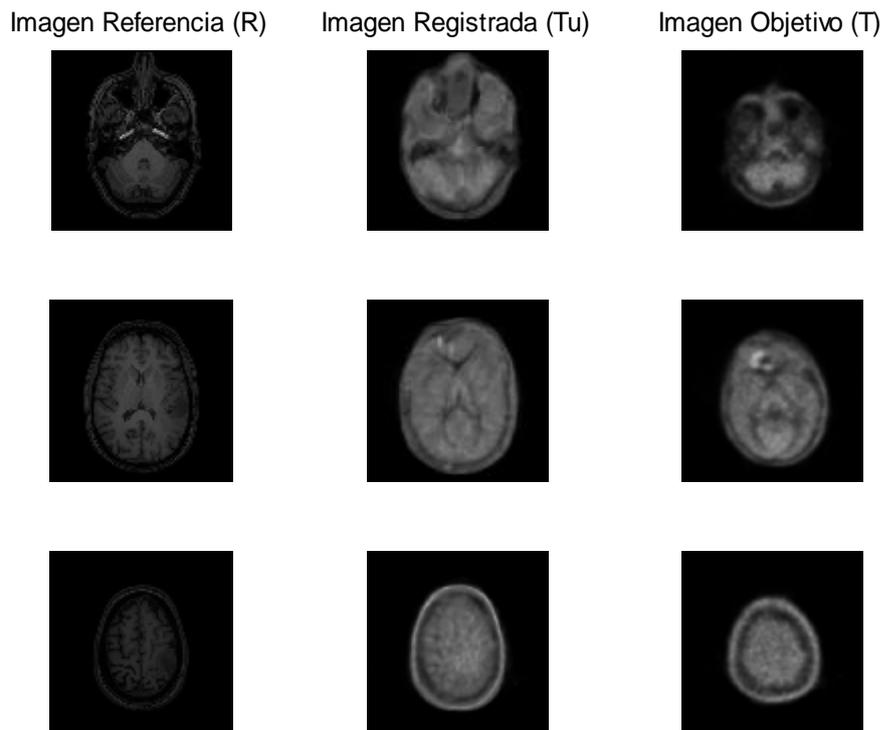


Figura 5.21: Registro volumen PET mediante método de parches locales

Se alcanza la solución óptima con un CR=90.22% en la iteración $\xi=200$. Hemos utilizado los siguientes valores para los parámetros: $\alpha=50$, $\tau=1$ y $\sigma=1$.

Por último realizaremos el registro del cerebro de un paciente:

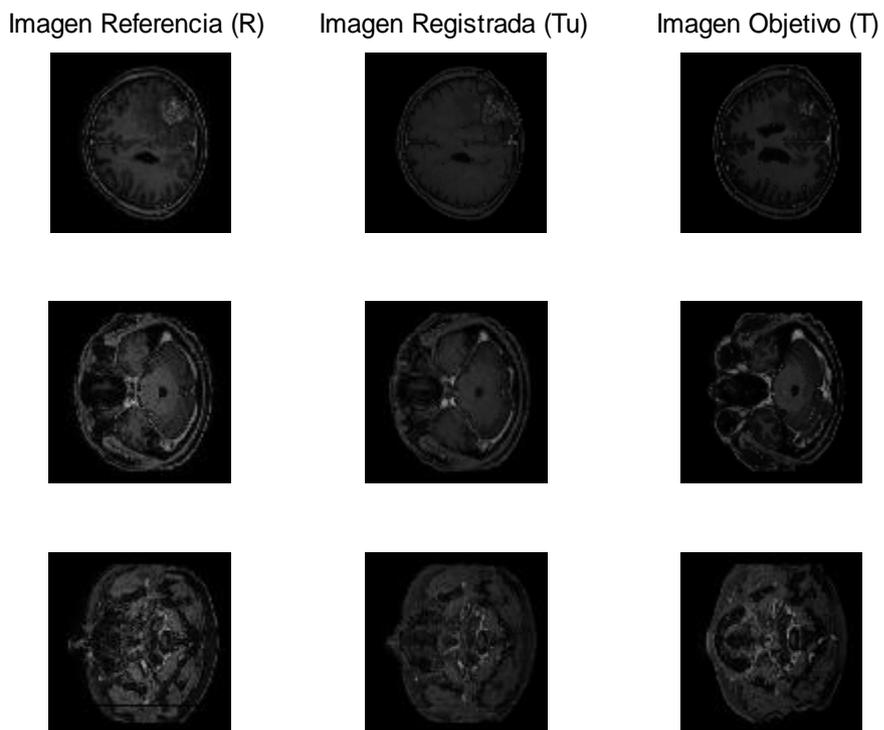


Figura 5.22: Registro volumen Paciente mediante método de parches locales

Se alcanza la solución óptima con un CR=87.40% en la iteración $\xi=600$. Hemos utilizado los siguientes valores para los parámetros: $\alpha=20$, $\tau=1$ y $\sigma=1$.

5.2.3. Registro con marco variacional

Como en el método anterior comenzaremos utilizando el volumen WB128. Tras el registro obtenemos:

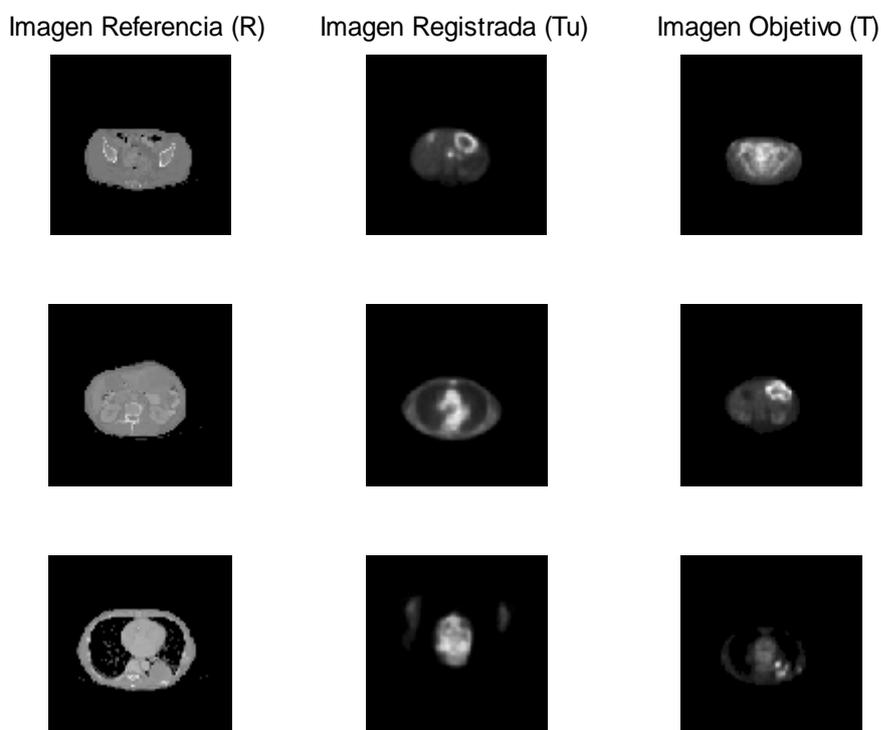


Figura 5.23: Registro volumen WB128 en un marco variacional

Se alcanza la solución óptima con un CR=36.72% en la iteración $\xi=282$. Hemos utilizado los siguientes valores para los parámetros: $\alpha=1000$, $\tau=1$ y $\sigma=1$.

Utilizaremos a continuación el volumen del hígado, cuyos resultados se muestran en la Figura 5.24.

Se alcanza la solución óptima con un CR=78.66% en la iteración $\xi=320$. Hemos utilizado los siguientes valores para los parámetros: $\alpha=850$, $\tau=1$ y $\sigma=1$.

Ahora procedemos a registrar el volumen denominado BOLD (Figura 5.25).

Se alcanza la solución óptima con un CR=88.74% en la iteración $\xi=100$. Hemos utilizado los siguientes valores para los parámetros: $\alpha=20$, $\tau=1$ y $\sigma=1$.

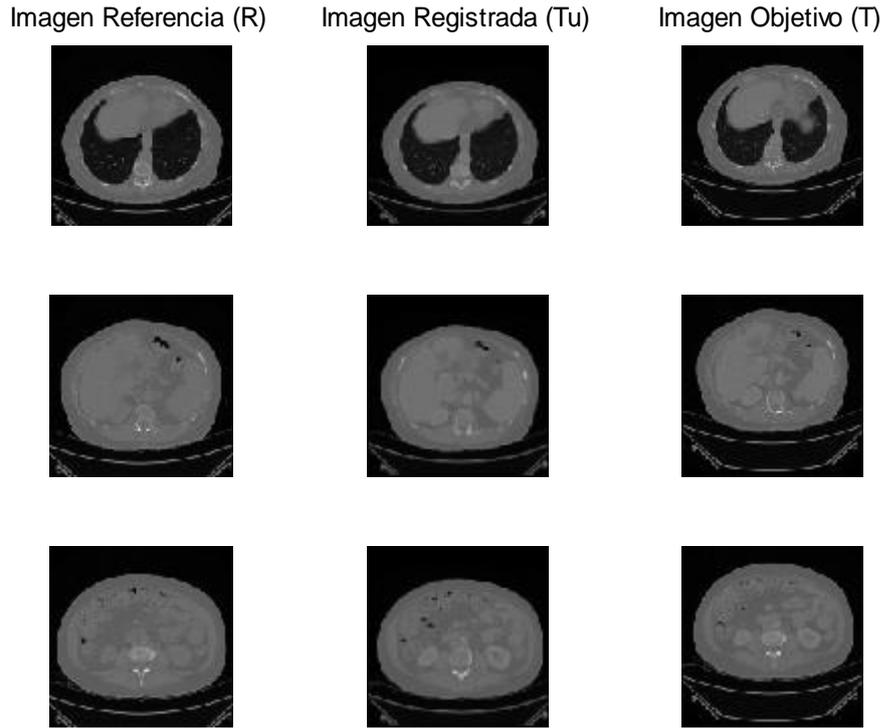


Figura 5.24: Registro volumen Hígado en un marco variacional

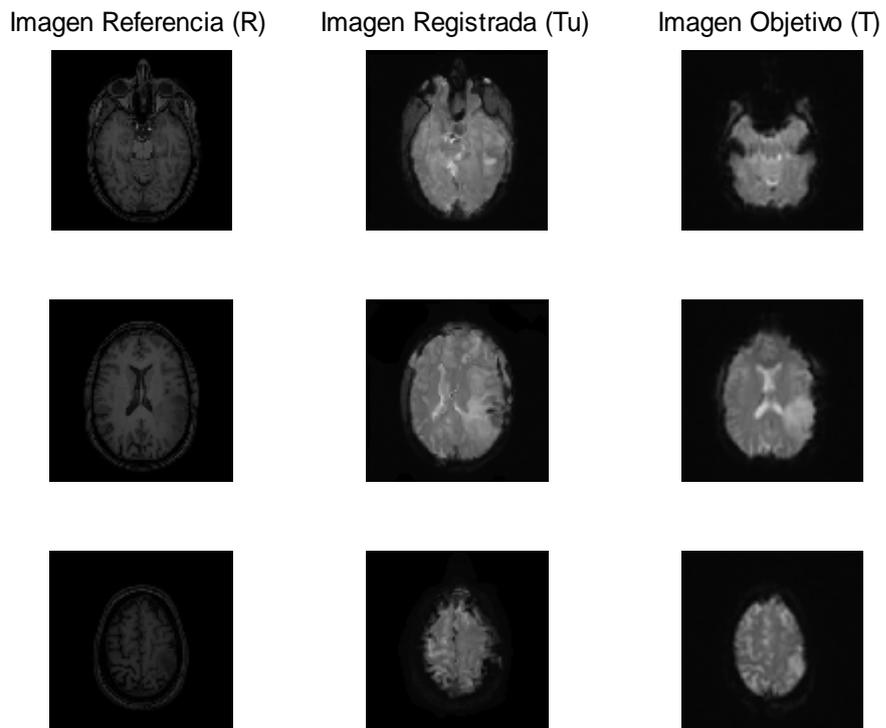


Figura 5.25: Registro volumen BOLD en un marco variacional

Tras este, procedemos a registrar el volumen denominado PET:

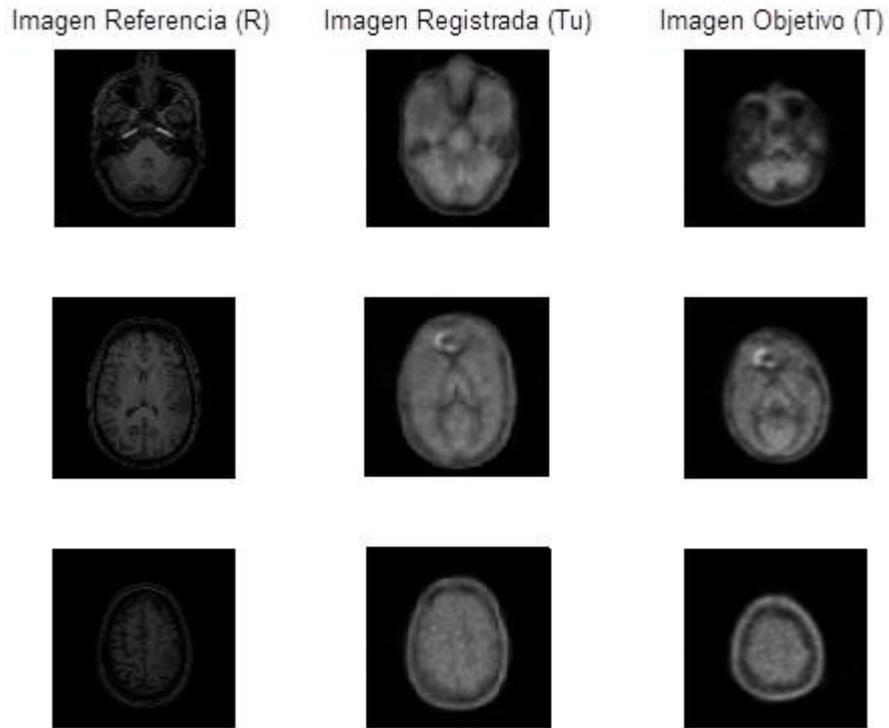


Figura 5.26: Registro volumen PET en un marco variacional

Se alcanza la solución óptima con un CR=88.51% en la iteración $\xi=150$. Hemos utilizado los siguientes valores para los parámetros: $\alpha=50$, $\tau=1$ y $\sigma=1$.

Por último realizaremos el registro del cerebro de un paciente:

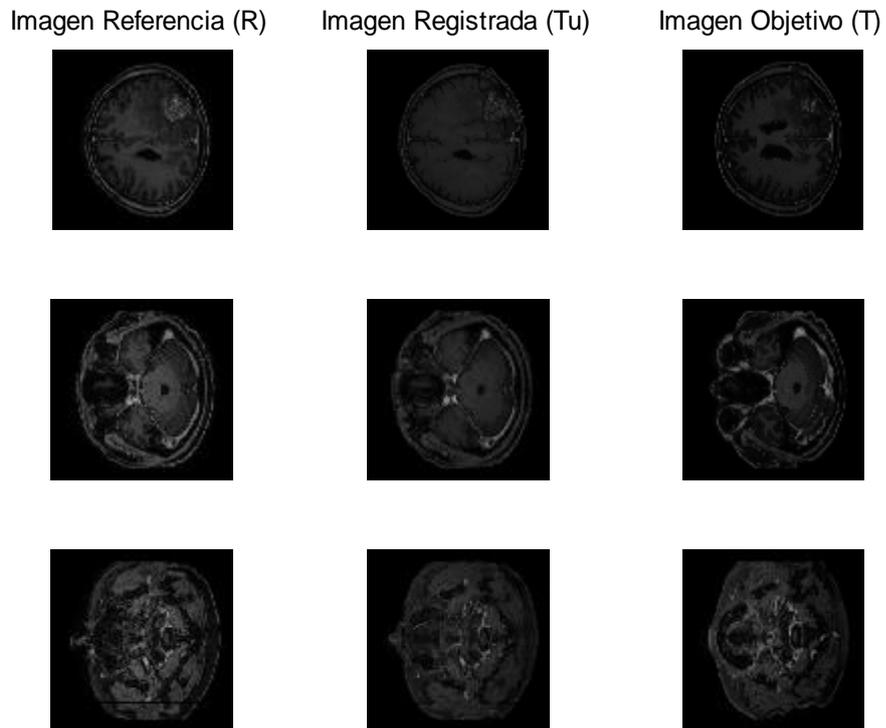


Figura 5.27: Registro volumen Paciente en un marco variacional

Se alcanza la solución óptima con un CR=87.66% en la iteración $\xi=600$. Hemos utilizado los siguientes valores para los parámetros: $\alpha=20$, $\tau=1$ y $\sigma=1$.

5.2.4. Comparación de resultados

En este apartado reuniremos todos los resultados obtenidos con los métodos anteriores para cada conjunto de volúmenes en unas tablas para poder comparar los resultados de una manera sencilla.

Como vamos a seguir el orden utilizado en los apartados anteriores, empezaremos con el volumen WB128:

| Método | Parámetro α | Iteraciones ξ | CR (%) |
|-------------------|--------------------|-------------------|--------|
| Resultante Global | 1000 | - | - |
| Parches locales | 1000 | 2000 | 70.67 |
| Variacional | 1000 | 282 | 36.72 |

Tabla 5.6: Comparación resultado volumen WB128

Este es el ejemplo más claro de la influencia de las correspondencias geométricas. Como se puede observar, con un registro variacional no se puede registrar este volumen. Sin embargo, al introducir correspondencias geométricas se puede comprobar que con el método de parches locales se consigue registrar. Aunque el valor numérico no es del todo bueno, si aumentásemos más el número de iteraciones se conseguiría un aumento del ratio de correlación.

Por otro lado, como ya se explicó anteriormente, la inestabilidad del método de resultante global impide el registro de este volumen.

Ahora mostraremos los resultados obtenidos con el conjunto de volúmenes correspondientes al hígado:

| Método | Parámetro α | Iteraciones ξ | CR (%) |
|-------------------|--------------------|-------------------|--------|
| Resultante Global | 850 | 439 | 80.47 |
| Parches locales | 850 | 489 | 79.59 |
| Variacional | 850 | 320 | 78.66 |

Tabla 5.7: Comparación resultado volumen Hígado

En este ejemplo se comprueba cómo con las correspondencias geométricas se consigue una pequeña mejora. En este caso además se obtiene un mejor resultado con el método de resultante global.

A continuación mostramos la tabla del conjunto formado por la resonancia magnética y el denominado BOLD:

| Método | Parámetro α | Iteraciones ξ | CR (%) |
|-------------------|--------------------|-------------------|--------|
| Resultante Global | 20 | 130 | 78.43 |
| Parches locales | 20 | 150 | 82.54 |
| Variacional | 20 | 100 | 88.74 |

Tabla 5.8: Comparación resultado volumen BOLD

En este caso podemos observar que las correspondencias geométricas han empeorado el registro obtenido con el método variacional. Esto se debe a la dificultad de encontrar correspondencias geométricas correctas de forma manual.

Ahora mostraremos los resultados obtenidos registrando la resonancia magnética con el PET:

| Método | Parámetro α | Iteraciones ξ | CR (%) |
|-------------------|--------------------|-------------------|--------|
| Resultante Global | 50 | - | - |
| Parches locales | 50 | 200 | 90.22 |
| Variacional | 50 | 150 | 88.51 |

Tabla 5.9: Comparación resultado volumen PET

Aquí si hemos encontrado una mejora con la integración de *landmarks*, pero únicamente con el método de parches locales, ya que con el método de la resultante global no se ha podido obtener un registro adecuado.

Por último analizaremos el conjunto del paciente:

| Método | Parámetro α | Iteraciones ξ | CR (%) |
|-------------------|--------------------|-------------------|--------|
| Resultante Global | 20 | - | - |
| Parches locales | 20 | 600 | 87.40 |
| Variacional | 20 | 600 | 87.66 |

Tabla 5.10: Comparación resultado volumen Paciente

En este ejemplo apenas se aprecia diferencia entre un método y otro. Lo único que cabe destacar es que otra vez ha sido imposible registrar mediante el método de resultante global.

6. Conclusiones y líneas futuras

Con este capítulo se pretende sacar las conclusiones principales de este proyecto y finalmente comentar cuáles podrían ser las líneas futuras de trabajo para seguir mejorando lo planteado aquí.

La motivación de este proyecto era comprobar si se podía mejorar el registro en un marco variacional integrando correspondencias geométricas. Para encontrar las correspondencias geométricas hemos utilizado dos procedimientos diferentes. Siempre que ha sido posible hemos utilizado la función SURF, sin embargo, cuando esta era incapaz de encontrar correspondencias geométricas o las que encontraba no eran correctas, hemos tenido que detectarlas manualmente con la función “cpselect” de MATLAB.

Como hemos comentado a lo largo de este trabajo, hemos empleado dos métodos que integran correspondencias geométricas: el método de *resultante global* y el método de *parches locales*.

Antes de concluir la utilidad de las correspondencias geométricas, parece necesario hacer un balance de estos dos métodos y sacar conclusiones de cada uno de ellos.

Empezaremos con el método de *resultante global*. Como hemos podido comprobar en el capítulo 5, cuando hemos tratado con imágenes (2 dimensiones) se consigue en la mayoría de los ejemplos un registro bueno, mejorando en algunos casos lo que se obtiene sin correspondencias geométricas. El problema aparece cuando trabajamos con volúmenes (3 dimensiones) puesto que debido a la inestabilidad del método y a la dificultad de localizar correspondencias geométricas en la mayoría de los casos no se ha podido realizar el registro, y sólo en uno de los ejemplos que se ha podido ha mejorado a los otros métodos.

Por otro lado tenemos el método de *parches locales*. Este método ha sido introducido por nosotros e implementado según se explica en el capítulo 4. A diferencia del método anterior, con este método se consigue un registro bueno tanto con imágenes como con volúmenes.

A raíz de esto se llega a la conclusión de que el método de *parches locales* funciona mejor que el método de *resultante global* de forma general tanto para 2 dimensiones como para 3 dimensiones.

Tras esta comparación, ahora pasaremos a analizar lo fundamental de este proyecto, es decir, si la introducción de correspondencias geométricas mejora el registro de imagen en un marco variacional. Como se puede comprobar, claramente, tanto en la imagen creada por nosotros (sección 5.1) como en el volumen WB128 (sección 5.2) sin correspondencias geométricas no se puede realizar el registro, por lo que la integración de estas sí que produce una mejora.

En el resto de ejemplos se produce una pequeña mejora en la mayoría de ellos, aunque no tan significativa como en esos dos ejemplos. Opinamos que esto se debe a la forma en que hemos tenido que detectar las correspondencias, ya que si hubiéramos sido más precisos se hubiera obtenido una mejora mayor, mejorando los registros obtenidos.

Esto nos lleva a las líneas futuras de trabajo, ya que pensamos que la mejor manera de mejorar el funcionamiento de los métodos implementados en este proyecto es con una correcta detección de las correspondencias geométricas, y estas no deberían detectarse

manualmente, sino que deberían ser detectadas mediante una función auxiliar, por ejemplo SURF.

Debido a esto, la mejor manera de avanzar sería implementando SURF para que pueda trabajar tanto con imágenes multimodales como con volúmenes (3 dimensiones).

7. Bibliografía

- [1] Bernd Fischer and Jan Modersitzki: *Ill-posed medicine—an introduction to image registration*, 2003.
- [2] Jorge Larrey: *Formulación Variacional del registro de imagen en el dominio de Fourier, y técnicas para su implementación optimizada*, 2008.
- [3] Jacob Toft Pedersen: *SURF: Features detection and description*, 2011.
- [4] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool ETH Zurich: *SURF: Speeded Up Robust Features*, Katholieke Universiteit Leuven.
- [5] D. G. Lowe: *Distinctive image features from scale invariant keypoints*, International Journal of Computer Vision, 2004.
- [6] Fischer, B. y J. Modersitzki: *Fast diffusion registration*. M.Z. Nashed, O. Scherzer (eds), Contemporary Mathematics 313, Inverse Problems, Image Analysis, and Medical Imaging AMS, páginas 11-129, 2002.
- [7] Fischer, B. y J. Modersitzki: *Combining landmark and intensity driven registrations*, 2003.
- [8] Fischer, B. y J. Modersitzki: *Large scale problems arising from image registration*, 2004.
- [9] M. Urschler, C. Zach, H. Ditt, H. Bischof: *Automatic Point Landmark Matching for Regularizing Nonlinear Intensity Registration: Application to Thoracic CT Images*, 2006.