Universidad Zaragoza
1542

Trabajo Fin de Grado en Ingeniería Informática

Cálculo de flujo óptico denso en imágenes de colonoscopias mediante aprendizaje no supervisado

Autor

Iván Gonzalo Laplaza

Directores

José María Martínez Montiel

Javier Morlana Ledesma

Escuela de Ingeniería y Arquitectura

2022-23

RESUMEN

Se ha evaluado un método de flujo óptico denso en imágenes de colonoscopia, que ha sido adaptado al dominio del colon mediante un método de entrenamiento no supervisado.

Para ello, primero se ha construido un dataset para *training*, *validation* y *test*, a partir de las secuencias del Endomapper dataset. Una vez se disponía de un conjunto de test, se ha evaluado el modelo proporcionado por los autores, que fue entrenado en el dataset de Megadepth, que contiene imágenes de monumentos alreadedor del mundo. Los resultados con este modelo son bastante positivos, a pesar de que la red no haya procesado ninguna imagen del colon previamente.

El modelo ha sido entrenado en el conjunto de *train* creado, obteniendo una versión adaptada al dominio del colon. Además de los datos, se han realizado una serie de ajustes necesarios para mejorar el funcionamiento del modelo. El modelo entrenado es capaz de calcular el flujo entre imágenes que presentan grandes rotaciones, y también es capaz de calcular el flujo de forma robusta bajo cambios de iluminación.

Las modificaciones realizadas al repositorio original están escritas en Python y están disponibles en un repositorio privado de Github.

Agradecimientos

Me gustaría agradecer a mis directores, José María Martínez Montiel y Javier Morlana Ledesma, su interés e implicación con este proyecto.

This project has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement No 863146

Este proyecto ha recibido una beca de colaboración del Ministerio de Educación y Formación Profesional

Índice

Capítulo 1

Introducción

Este trabajo se enmarca en el proyecto Endomapper [1], cuyo objetivo es conseguir un mapa en tiempo real del colon a partir de secuencias de colonoscopias. Uno de los procesos de vital relevancia en las tecnologías SLAM (Simultaneous Localization And Mapping) es la asociación de datos que, en el dominio de las imágenes, supone ser capaces de relacionar píxeles entre dos imágenes que corresponden al mismo lugar. Por ejemplo, la esquina de una ventana, vista desde dos imágenes diferentes.

Tradicionalmente, esto se ha hecho con características clásicas como ORB o SIFT. Estos métodos no funcionan bien en el colon, debido a los cambios en la apariencia de las imágenes.

Otra familia de métodos son los de flujo óptico, que tratan de obtener el movimiento de todos los píxeles de una imagen a otra. Estos métodos suelen depender de la consistencia fotométrica, es decir, que la intensidad de dos píxeles correspondientes en dos imágenes sea constante. Esta restricción no se cumple en el colon, debido a las variaciones de iluminación anteriormente mencionadas.

Los métodos de flujo más modernos utilizan métodos de *deep learning*, y han demostrado ser capaces de superar alghunas de las limitaciones de los métodos de flujo clásicos. Una ventaja de estos métodos de flujo es que consideran toda la información de la imagen, en lugar de utilizar información únicamente local como en el caso de las características locales. La desventaja de estos métodos es que confían en tener una supervisión ground truth a la hora de entrenar los modelos de estimación de flujo. Esto, nuevamente, es algo de gran dificultad en el dominio de las colonoscopias.

Debido a estas restricciones, se ha optado por un modelo de estimación de flujo que una utiliza una técnica de entrenamiento no supervisado. Esta técnica, llamada *warp consistency*, sólo necesita de pares de imágenes que observen el mismo lugar para realizar el entrenamiento de la red.

La elección de este sistema se sustenta en las dos cuestiones mencionadas previamente. Por un lado, el uso de un modelo de flujo basado en *deep learning* permite

calcular una transformación coherente para todos los píxeles de la imagen, al utilizar toda la información contenida en la imagen al mismo tiempo. Esto, a diferencia de los métodos de características locales, permite estimar una transformación entre imágenes aun cuando la textura local en la imagen sea baja. Por otro lado, *warp consistency* emplea un algoritmo no supervisado, lo cual es vital para el dominio del colon, donde no se dispone de supervisión confiable.

Además, aunque el resultado del flujo que se obtiene siguiendo la técnica de *warp consistency* es bastante bueno, este mejora al entrenar el modelo con las imágenes del Endomapper. Por lo que se trata de un método bastante eficaz para el cálculo del flujo óptico denso en colonoscopias.

Por último, cabe destacar que los autores de *warp consistency* proporcionan acceso al código de su proyecto así como documentación y ejemplos de uso.

1.1. Objetivos

Los objetivos de este proyecto son los siguientes:

- Puesta en marcha de *warp consistency*: partiendo del código de los autores originales, se instal el software y hacer los primeros cálculos de flujo en imágenes de colonoscopia usando el modelo out-of-the-box.
- Preparación del conjunto de datos: usando secuencias del Endomapper dataset, se obtendrá un set de *train*, *validation* y *test* con el que poder entrenar modelos y evaluar el funcionamiento.
- Entrenamiento en Endomapper: una vez que el sistema esté instalado, se entrenará el modelo WarpC+GLUNet con las imágenes del Endomapper Dataset [1]
- Validación experimental: Con el modelo entrenado, se compararán sus resultados obtenidos con respecto al modelo out-of-the-box.
- Ejecución en DGX: Se aprovechará la potencia de cálculo de una estación de trabajo para mejorar el rendimiento de *warp consistency*.

Capítulo 2

Cálculo del flujo óptico mediante aprendizaje no supervisado

El flujo óptico entre dos imágenes se puede definir como el movimiento que hay que aplicar a cada píxel de la primera imagen para que coincida con su correspondiente píxel en la segunda imagen. De esta forma, si a cada píxel de la primera imagen se le aplica su correspondiente valor de flujo, se obtiene una aproximación de la segunda imagen. Puede verse un ejemplo detallado en la Figura 2.1.

Los métodos clásicos están basados en consistencia fotométrica [2]. Los primeros métodos de aprendizaje empleaban aprendizaje supervisado [3, 4], mediante el uso de simulación, ya que es difícil encontrar en la práctica supervisión fiable para todos los píxeles. Además, existe el problema del salto que existe entre las imágenes reales y las obtenidas con simulación que pueden ser muy diferentes.

Por otra parte, están los métodos no supervisados. La diferencia principal entre ambos es que en los no supervisados no se requiere de un flujo ground-truth, el cuál se define como la estimación exacta y precisa del movimiento de todos los píxeles entre dos imágenes. En el contexto de las imágenes de colonoscopia es bastante difícil obtener el flujo ground-truth, ya que los métodos clásicos de estimación de flujo óptico no tienen un buen desempeño en este dominio. Además, se dispone exclusivamente de la información captada por una cámara, por lo que se dificulta todavía más el proceso de conseguir un flujo ground-truth confiable. De esta forma, se va a seguir una estrategia de aprendizaje no supervisado.

2.1. Flujo óptico auto-supervisado mediante warp consistency

La *warp consistency*, a la que nos referiremos de forma abreviada como WarpC [5], es una técnica para supervisar el cálculo del flujo óptico denso. Su funcionamiento se

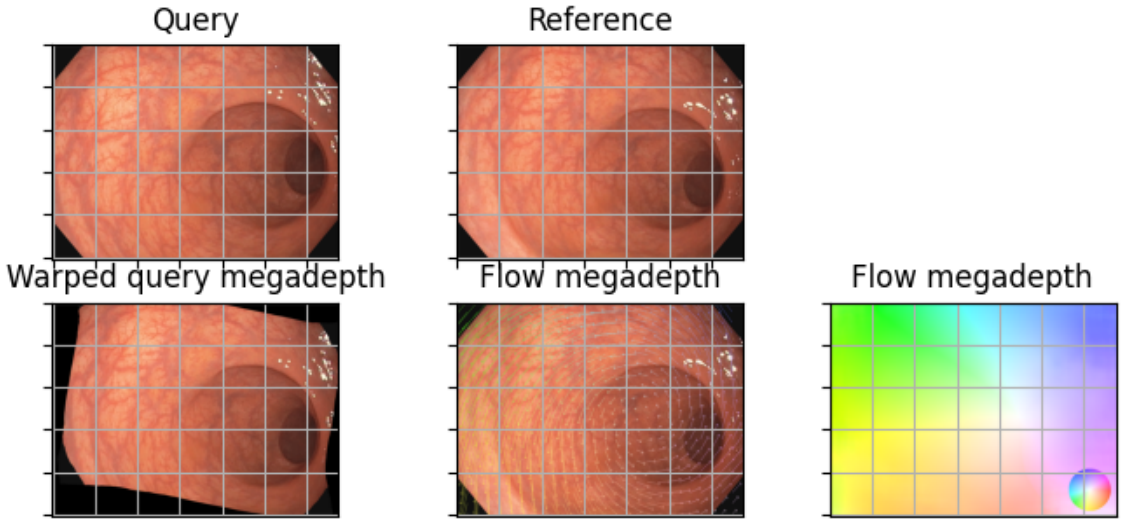


Figura 2.1: Ejemplo de estimación de flujo óptico utilizando un par de imágenes del Endomapper Dataset [1]. En la primera fila se muestran las imágenes de query y referencia (derecha). En la segunda fila se pueden ver la warped query (izquierda), que es el resultado de aplicar el flujo a la imagen de query para convertirla en la imagen de referencia; el flujo representado en ciertos puntos con flechas (centro); y el flujo evaluado en todos los píxeles de la imagen (derecha) debido a que es un flujo denso (Capítulo 2).

basa en el empleo de tripletas de imágenes. Para supervisar el flujo óptico (F) entre 2 imágenes I y J, se crea en primer lugar la imagen I', que es el resultado de aplicar una transformación conocida W a la imagen I (Figura 2.2). Estas transformaciones pueden ser homografías, transformaciones afines o transformaciones *thin plate spline* (tps).

Con las 3 imágenes, se calcula por un lado el flujo desde la imagen I a la J y por otro el de la I' a la J. Como se conoce el flujo de I a I' (W) y se tiene tanto el de I' a J como el de I a J, se puede evaluar la calidad de la estimación del flujo de I a J, porque debería ser la misma que de I a J pasando por I', a partir de esta evaluación de la calidad de la estimación de flujo se puede establecer la función de loss y por lo tanto la supervisión.

Utilizando una notación más formal, se puede definir W como el flujo de I' a J más el warping I' a J del flujo de J a I:

$$W = F_{I' \to J} + \Phi_{F_{I' \to J}}(F_{J \to I}) \tag{2.1}$$

donde el warping Φ_F de una función T se define como el flujo F que cumple:
$\Phi_F(T)(x) = T(x + F(x))$.

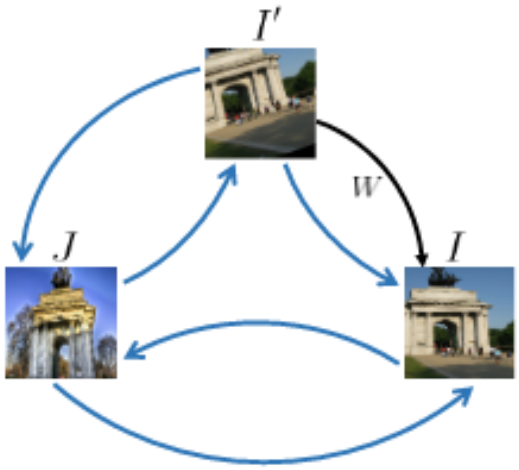


Figura 2.2: Grafo de flujos para ilustrar la supervisión del flujo mediante de warp consistency. Imagen obtenida de la Figura 3.c de [5]

2.2. GLUNet

La WarpC podría aplicarse para supervisar diferentes redes de estimación de flujo. En este trabajo proponemos emplear red de estimación de flujo GLUNet [6] siguiendo la propuesta de [5].

La arquitectura de GLUNet (Figura 2.3) está basada en capas de correlación global y rama local para estimar el flujo óptico denso entre imágenes. La red se divide en dos ramas: una rama global que captura características de alto nivel y una local que se enfoca en detalles más específicos. Estas ramas se fusionan mediante un módulo de fusión adaptativo que ajusta la resolución de la entrada global para adaptarse a la escala del detalle local. Además, con esta arquitectura es posible calcular con gran precisión y fiabilidad desplazamientos a largas distancias, incluso cuando hay cambios en la apariencia o el ángulo de observación.

2.3. Loss function

La función de pérdida (loss) utilizada en WarpC [5] se basa en la consistencia de flujo (warp consistency), que establece que el flujo óptico estimado entre dos imágenes debe ser consistente con el flujo óptico estimado entre una imagen y su versión deformada. Esto se detalla en la Sección 2.1. Teniendo esto en cuenta, la función de loss de WarpC se define como la suma ponderada de dos términos. Por un lado, un término de consistencia de flujo global, que mide la coherencia del flujo óptico en toda la imagen y, por otro lado, un término de flujo local que tiene en cuenta la consistencia del flujo en regiones pequeñas. Se puede ver un ejemplo de la evaluación de esta loss a lo largo

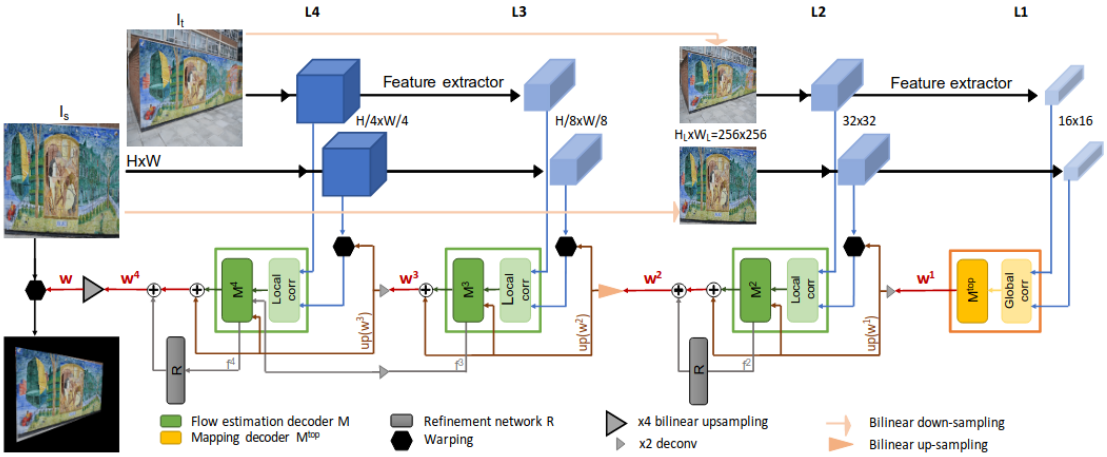


Figura 2.3: Arquitectura de GLUNet. En la parte de la izquierda se encuentra la H-Net que se corresponde con la rama local. A la derecha, se puede observar la L-Net, que se corresponde con la rama global (Sección 2.2). Imagen obtenida de la Figura 3 de [6]

del entrenamiento del modelo WarpC+GLUNet en la Figura 3.4.

Con la misma notación que la utilizada en Sección 2.1, los términos que componen la función de loss de WarpC son la loss W-bipath (Ecuación 2.2) y la loss warp (Ecuación 2.3). La primera se obtiene a partir de la consistencia del flujo entre I' y J más el warp de J a I con W (que es el flujo sintético de I' a I). Es decir, esta loss se minimiza cuando el flujo de I' a J más el warp de J a I es similar a W. Por otra parte, la segunda función de loss se obtiene de la restricción de consistencia del flujo de I' a I con respecto a W. Como W es el flujo de I a I', esta función se minizará si ambos flujos tienen un valor similar. Estás 2 funciones de loss se unifican (Ecuación 2.4) de tal forma que la loss final es la suma de la loss W-bipath con la loss warp multiplicada por un parámetro de regularización λ. Este parámetro λ se ajusta automáticamente tras cada iteración del entrenamiento según se indica en la Ecuación 2.5

$$L_W = \left\| \widehat{F}_{I' \to J} + \Phi_{\widehat{F}_{I' \to J}}(\widehat{F}_{J \to I}) - W \right\| \tag{2.2}$$

$$L_{warp} = \left\| \widehat{F}_{I' \to I} - W \right\| \tag{2.3}$$

$$L = L_W + \lambda L_{warp} \tag{2.4}$$

$$\lambda = L_W / L_{warp} \tag{2.5}$$

2.4. Warps

En la técnica de WarpConsistency [5] se utilizan transformaciones afines, homografías y transformaciones tps. A continuación, se va a explicar en qué consiste cada una de ellas. En primer lugar, una transformación afín es aquella que conserva las rectas paralelas y las relaciones de proporcionalidad entre las distancias. Las transformaciones afines también se pueden ver como una combinación de traslaciones, rotaciones y escalados. Por otra parte, una homografía es una transformación geométrica que mapea los puntos de una imagen en otra. Permite transformaciones más complejas que las que se pueden conseguir utilizando exclusivamente transformaciones afines, pero por contra, no conserva el paralelismo ni las relaciones existentes entre las distancias. Por último una transformación tps (Thin-Plate Spline) es una transformación no lineal que permite deformaciones más suaves que los otros dos tipos anteriores. Se compone de un conjunto de funciones lineales junto con otras dependientes de una base radial.

Capítulo 3

Adaptación al dominio de las colonoscopias

El modelo WarpC+GLUNet (que se explicará con más detalle en la Sección 4.3) se ha diseñado para procesar el dataset de Megadepth [7]. Para poder utilizarlo con el dataset de EndoMapper [1], fue necesario modificar el formato del dataset para que coincidiera con el utilizado en Megadepth.

3.1. Datos entrenamiento y validación

Los datos que se van a utilizar para entrenar el modelo WarpC+GLUNet propuesto por [5] consisten en 8 secuencias del Endomapper Dataset [1]. Estas imágenes han sido procesadas con COLMAP [8], obteniendo una serie de clústers de ellas. Un clúster es un conjunto de imágenes covisibles, es decir, que observan el mismo lugar. Se define que dos imágenes son covisibles si observan puntos en común. Se pueden visualizar algunos ejemplos de los clústers utilizados en la Figura 3.1. Cada secuencia contiene 24.63 clústers de media (obtenidos mediante [8]), teniendo cada clúster una media de 125.15 imágenes. Se pueden ver más detalles sobre los conjuntos de entrenamiento, validación y test en la Tabla 3.1.

Por su parte, los datos de validación se componen de 2 secuencias distintas a las 8 de entrenamiento pero que siguen su misma estructura. Además, con todas las imágenes de cada clúster se puede construir su matriz de solapamiento. Esta es la matriz que representa el nivel de covisibilidad de cada par de imágenes del clúster. Cada uno de los elementos de dicha matriz se obtiene a partir de cada par de imágenes, más concretamente del resultado de dividir el número de puntos que tienen en común con el número de puntos de cada una de las imágenes. En la Figura 3.2 se pueden ver ejemplos de un par de imágenes que tienen un alto nivel de covisibilidad (0.61) y otro cuyo nivel de covisibilidad es bajo (0.0017).

	Entrenamiento	Validación	Test
Número de secuencias	8	2	1
Número total de imágenes	24654	7599	974
Número total de clústers	197	51	9
Número medio de clústers por secuencia	24.63	25.5	9
Número medio de imágenes por clúster	125.15	149	108.22
Pares de imágenes utilizados	80x300=24000	80x25=2000	79

Tabla 3.1: Estadísticas de los datos de entrenamiento, validación y test. En el caso de los pares de imágenes utilizados durante el entrenamiento y la validación, se ha incluido el número de épocas y el número de pares de imágenes por época (Sección 3.3)

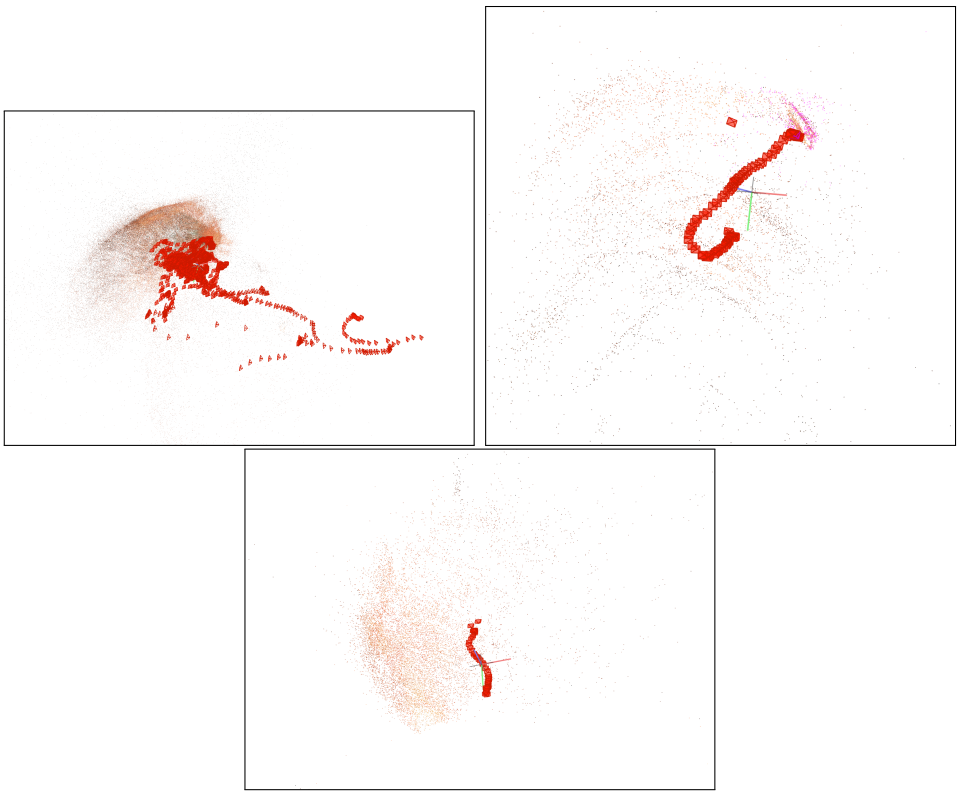


Figura 3.1: Ejemplo de clústers (Sección 3.1) utilizados para el entrenamiento del modelo "endomapper" (Sección 4.3). En cada uno están representados en color naranja los puntos obtenidos mediante SfM [8] y en color rojo la posición estimada de la cámara

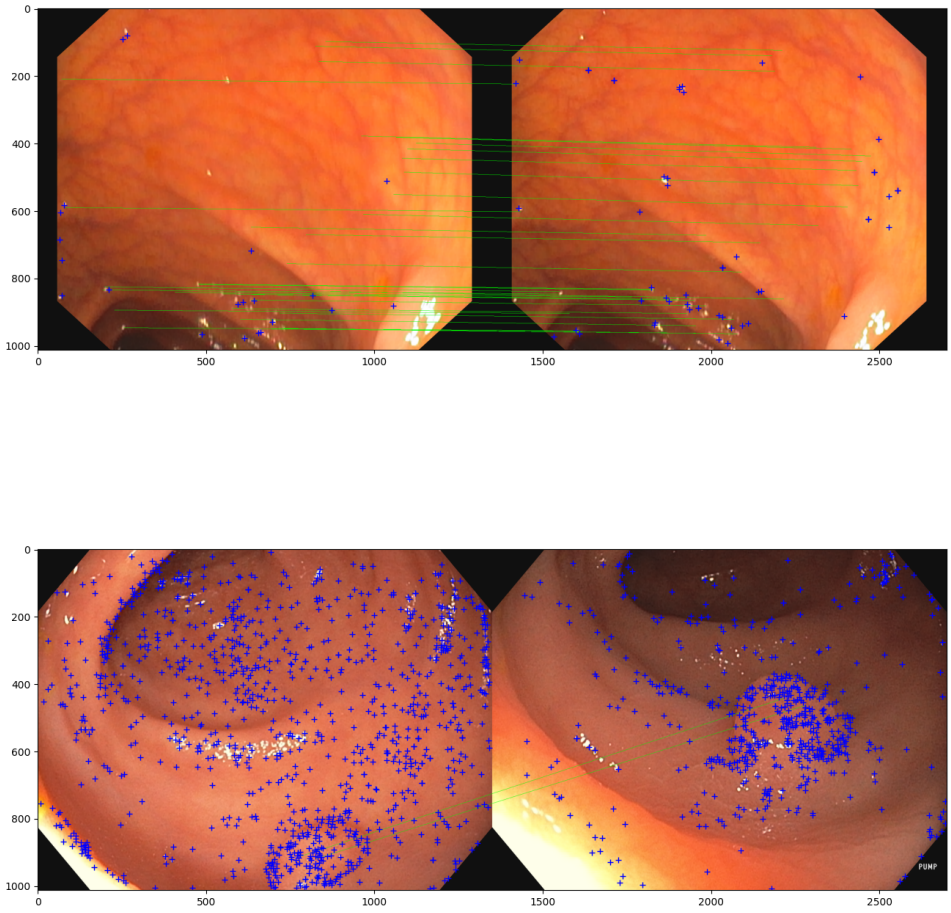


Figura 3.2: Ejemplos de covisibilidad alta (arriba) y baja (abajo). En verde están dibujados las líneas que unen los puntos en común de ambas imágenes. En azul están señalados los puntos exclusivos de cada imagen. El par de imágenes superior tiene una separación de 1 frame y tiene un nivel de covisibilidad de 0.61, mientras que en el inferior, con una separación de 200 frames, la covisibilidad es de 0.0017

3.2. Datos Test

De cara a evaluar el desempeño del modelo WarpC+GLUNet, se ha diseñado un sistema de evaluación basado en flujo óptico *sparse*. Esto significa que no se dispone del valor de flujo para todos los píxeles, sino para un subconjunto de ellos. Para ello, a partir de una secuencia diferente de las de entrenamiento y validación, se han obtenido un conjunto de clústers siguiendo el mismo sistema que la Sección 3.1. De estos clústers, se han obtenido pares de imágenes con distinto nivel de covisibilidad (ver Figura 3.2) y se han ordenado en función de su flujo medio ground-truth. El flujo ground-truth utilizado se trata de un flujo disperso o sparse debido a que sólo se conoce su valor para

un grupo reducido de píxeles de cada par de imágenes. De esta forma, se han generado un conjunto de 79 pares de imágenes. En la Figura 3.3 se muestran tres tests (el más fácil, el más difícil y el que se encuentra en la mitad) del conjunto total de tests junto con su flujo ground-truth. Estos son los tests 1, 39 y 79 y tienen un valor de flujo medio de 6.40 px, 112.99 px y 412.77 px respectivamente.

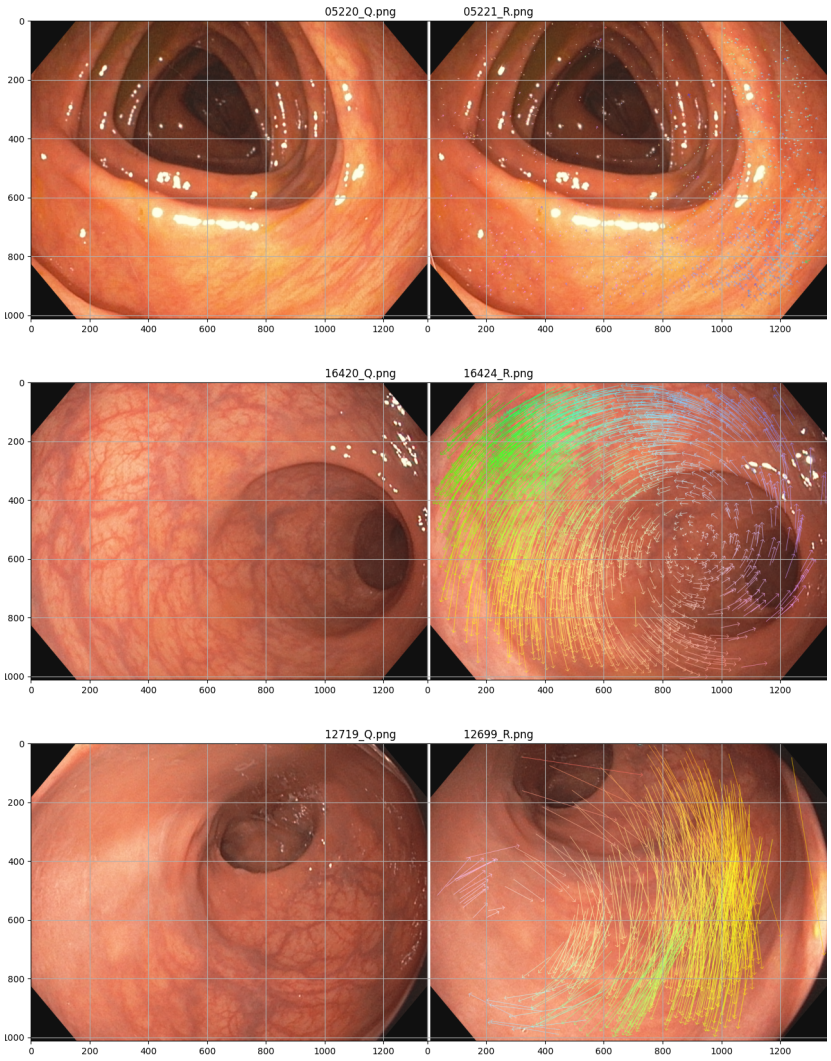


Figura 3.3: Ejemplos de pares de imágenes utilizados para test. Se muestra para cada uno la imagen de query (primera columna) y la de referencia (segunda columna). Además, se muestra el flujo disperso ground-truth sobre las imágenes de referencia

3.3. Metodología de entrenamiento

El entrenamiento del modelo WarpC+GLUNet [5] se realiza se la siguiente manera. En primer lugar, se seleccionan los pares de entrenamiento y validación a utilizar. Después, se filtran dichos pares utilizando dos umbrales mínimo y máximo para los valores de la matriz de solapamiento (esto se detallará en la Sección 4.3). Una vez

filtrados los pares, se aplica un redimensionado y un recorte a todas las imágenes (ver Sección 4.3). Una vez que las imágenes tienen el tamaño adecuado, se inicia el entrenamiento. El entrenamiento está configurado para ejecutarse durante 80 épocas. Para cada una de ellas, se eligen 300 pares de forma aleatoria con los que se entrena la GLUNet (Sección 2.2) utilizando (Sección 2.1) y 25 pares, también escogidos de forma aleatoria, para la validación. En la Figura 3.4 se muestra la función de loss de la red (Sección 2.3) en cada una de las 80 épocas.

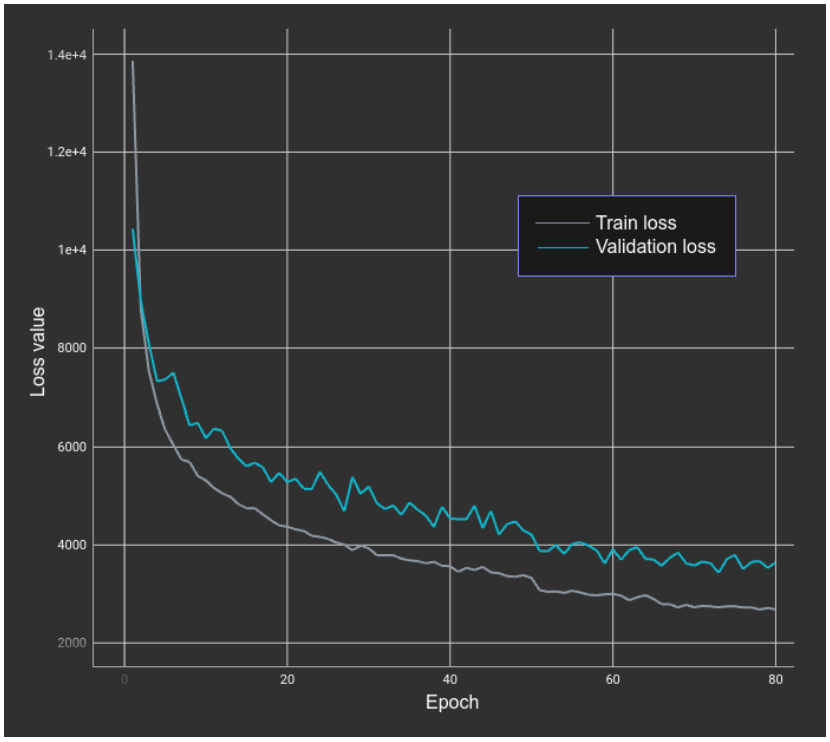


Figura 3.4: Función loss resultante de entrenar el modelo WarpC+GLUNet según se describe en la Sección 3.3. Para cada época se muestran los valores de loss de entrenamiento (gris) y de validación (azul).

Capítulo 4

Experimentos

El objetivo de esta sección es evaluar el desempeño del entrenamiento auto-supervisado mediante WarpC+GLUNet [5]. Por una parte, se evalúa el sistema out-of-the box, esto es, entrenado en Megadepth [7] y con test en EndoMapper [1]. Posteriormente, se evalúa como mejora el desempeño al entrenar con el EndoMapper dataset [1]. Finalmente, se muestran ejemplos de la capacidad del WarpC+GLUNet para calcular correspondencias entre imágenes reales del colon.

4.1. Métricas de evaluación

Para evaluar los distintos modelos que se van a utilizar, se va a hacer uso de dos métricas: el AEPE y el histograma de error acumulado.

En primer lugar se va a definir el concepto de EPE (End Point Error), que es el error existente entre la posición estimada de un píxel de una imagen y su posición real según el ground truth. De esta forma, el AEPE (Average End Point Error) es la media de los EPE de una imagen. Por su parte, el histograma de error acumulado muestra la distribución de los errores de todos los puntos medidos en cada par de imágenes.

4.2. Sistema out-of-the-box

En esta sección se van a describir las pruebas realizadas con el modelo WarpC+GLUNet propuesto por [5]. Este modelo se utilizará con el nombre "megadepth" de cara a los diferentes tests.

El modelo "megadepth" está construido sobre la arquitectura de la red GLUNet (Sección 2.2) utilizando la técnica de WarpConsistency (Sección 2.1) y ha sido entrenado utilizando el Megadepth dataset [7] que consta de imágenes de edificios de 196 localizaciones junto con valores de flujo sparse obtenidos mediante [8].

4.2.1. Toleracia a cambios de iluminación

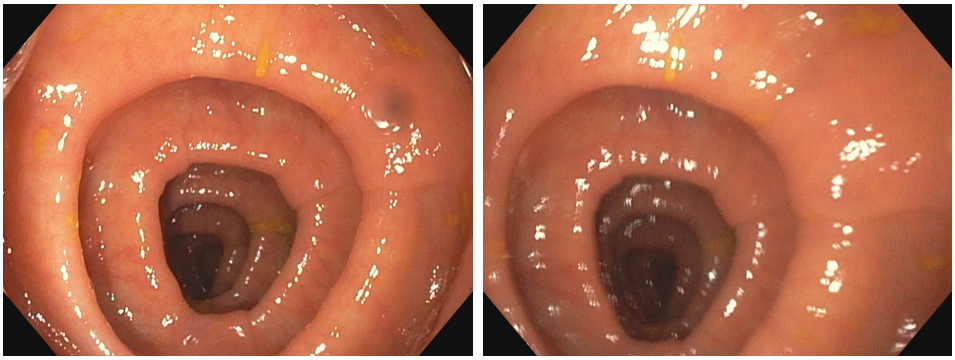


Figura 4.1: Imágenes de query (izda.) y referencia (dcha.) para los tests de cambio de brillo

En los pares de imágenes del Endomapper Dataset [1], la iluminación puede llegar a tener un alto nivel de variabilidad. Con el objetivo de comprobar si esta variación afectaba a la eficacia de los modelos anteriores, se diseñó un test en el que se aplicó una variación artificial de la iluminación mediante la modificación del brillo de las imágenes. En concreto, se han realizado un conjunto de 4 tests alterando los valores de brillo con factores 0.25, 0.5, 1.5, 2.0. En todos los tests de brillo se utilizaron las imágenes descritas en la Figura 4.1.

Como se puede apreciar en la Figura 4.2, cerca del 80 % de los errores son inferiores a 10 píxeles y el error en la mediana está próximo a los 2.5 píxeles. También puede apreciarse como el patrón general del flujo se mantiene. Se muestra que el modelo escogido, WarpC+GLUNet, es capaz de seguir produciendo flujos válidos ante cambios bruscos de iluminación, mostrando así su robustez.

4.3. Sistema entrenado en EndoMapper

En el entrenamiento es posible modificar una serie de parámetros que influyen en el modelo generado. A continuación, se describen aquellos que han sido modificados para obtener el modelo final, el cuál en los tests se utilizará con el nombre “endomapper”.

Factores de reescalado y recorte

Durante la fase de carga de los datos de entrenamiento, se realiza un reescalado de cada imagen a 750×750 para, posteriormente, realizar un recorte de la zona central de tamaño 520×520. De esta forma, se eliminan los posibles bordes negros de la imagen. Sin embargo, dado que las imágenes de [1] son rectangulares de tamaño 1350×1012, este procedimiento causa un cambio del píxel aspect ratio en dichas imágenes que se

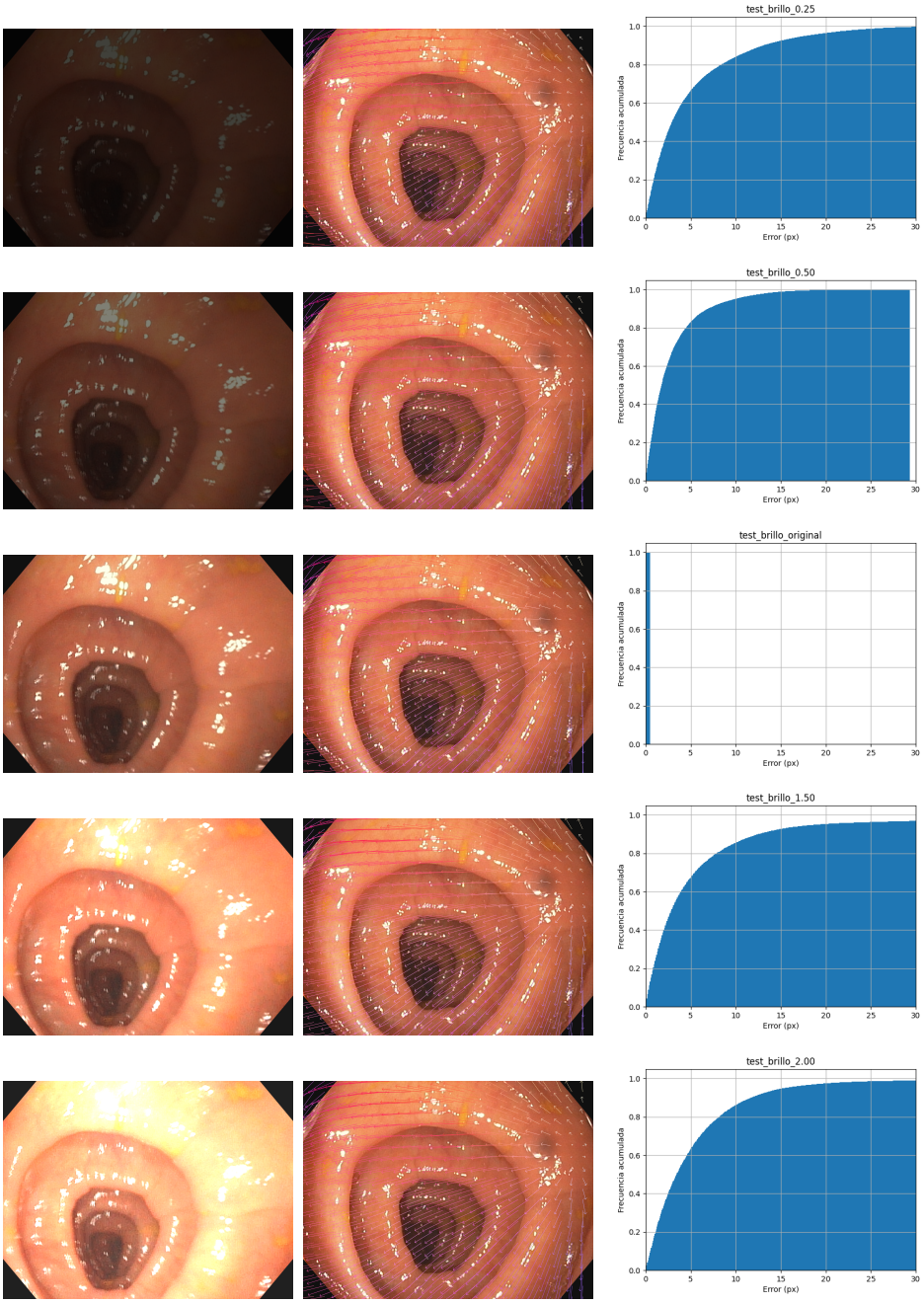


Figura 4.2: Test de robustez frente al cambio de brillo. Cada fila representa los cambios de brillo de factor 0.25, 0.5, 1 (original), 1.5 y 2. En cada columna se pueden ver la imagen de query con el nivel de brillo modificado, la imagen de referencia con el flujo dibujado con flechas y el histograma acumulado de la diferencia de flujo en valor absoluto respecto del flujo del par original (sin modificación de brillo)

traduce en una menor capacidad para estimar correctamente el flujo. Para solucionar este inconveniente, se modificó el factor de reescalado para que fuera rectangular, conservando la relación de aspecto de las imágenes. El recorte por su parte, se configuró para capturar el mayor área posible de cada imagen, de forma que, la forma de la imagen recortada siguiera siendo cuadrada como se puede ver en la Figura 4.3.

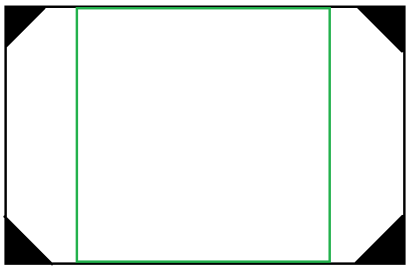


Figura 4.3: Recorte de forma cuadrada que maximiza el área capturada de las imágenenes del Endomapper Dataset [1]

Factor de giro

Este parámetro controla el valor del máximo ángulo que podían tener las rotaciones generadas por el generador de warps para la aumentación de los datos durante el entrenamiento de la red. Originalmente, este valor estaba establecido en $\pi/12$ rad (que se corresponden con 15°). No obstante, debido a que en los pares de imágenes utilizados existen rotaciones mayores, se ha incrementado este valor a $\pi/4$ rad (correspondiente con 45°).

Umbrales de la matriz de solapamiento

Este parámetro controla los valores mínimo y máximo a la hora de seleccionar las imágenes en función de su nivel de covisibilidad (ver Sección 3.1). De esta forma, los pares de imágenes que tengan un valor inferior al umbral mínimo o superior al umbral máximo serán descartados y no se tendrán en cuenta para el entrenamiento. Estos valores eran inicialmente 0.3 y 1.0, pero como se puede apreciar en la Figura 4.4, existía un número nada despreciable de pares que podrían utilizarse a pesar de su bajo grado de covisibilidad. Por ello, se decidió reducir el umbral mínimo a 0.1. Asimismo y con objeto de evitar utilizar pares demasiado sencillos, se modificó el umbral máximo a 0.99.

Entrenamiento del extractor de características

En el modelo original de WarpC+GLUNet se optó por utilizar los pesos preentrenados de ImageNet [9] puesto que los resultados que se obtienen tanto si se entrena toda la red como si no, son muy parecidos. En el caso del Endomapper Dataset [1] el dominio de las imágenes utilizadas (colonoscopias) difiere respecto de ImageNet, por lo que se ha elegido entrenar la red completa.

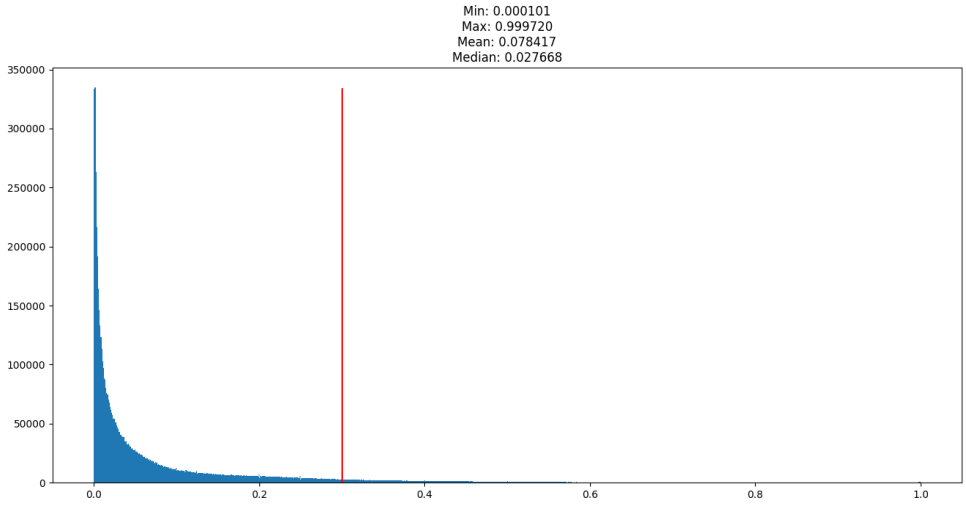


Figura 4.4: Histograma de los valores de todas las matrices de solapamiento de todos los pares utilizados para test. La barra vertical roja señala el umbral mínimo utilizado por defecto en el entrenamiento

4.4. Evaluación experimental

De cara a comparar el modelo out-of-the-box con el mejor modelo entrenado con el Endomapper Dataset [1], se han generado 79 pares de imágenes según la Sección 3.2 y se han ordenado en función de su flujo medio ground-truth. Después, se han separado en tres categorías de forma equitativa: fáciles, medios y difíciles. Para cada test, se ha estimado su flujo mediante el modelo descrito en la Sección 4.3 y se han calculado sus EPE (ver Sección 4.1) a partir de los puntos de los cuales se conoce el flujo sparse ground-truth. A partir de los EPE de los tests, se han calculado los histogramas acumulativos del EPE por cada categoría. Estos pueden ser visualizados en la Figura 4.5. Como se puede ver, el modelo "endomapper" supera ligeramente al modelo "megadepth" tanto en los tests fáciles como en los intermedios. En los tests difíciles, "endomapper" está por encima de "megadepth" excepto entre los EPEs 75 y 180.

Por otra parte, se han elegido varios tests, de diferentes niveles de dificultad para mostrar el funcionamiento del sistema. Se ha elegido el test que se encuentra en la mediana de cada categoría i.e fácil mediano, medio mediano y difícil mediano) para evaluar cada uno de los dos modelos. También se ha elegido como test el más fácil y el más difícil. Estos se muestran ordenados en la Figura 4.7.

Se observa que en todos los tests, exceptuando el medio mediano, el modelo entrenado con endomapper tiene un mejor desempeño pues el histograma acumulado está siempre por encima del megadepth. Se ha conseguido incluso captar aspectos como el giro de las imágenes. Además, se aprecia que los errores máximos son bastante elevados. Esto estará causado por la presencia de espurios en los clústers generados para

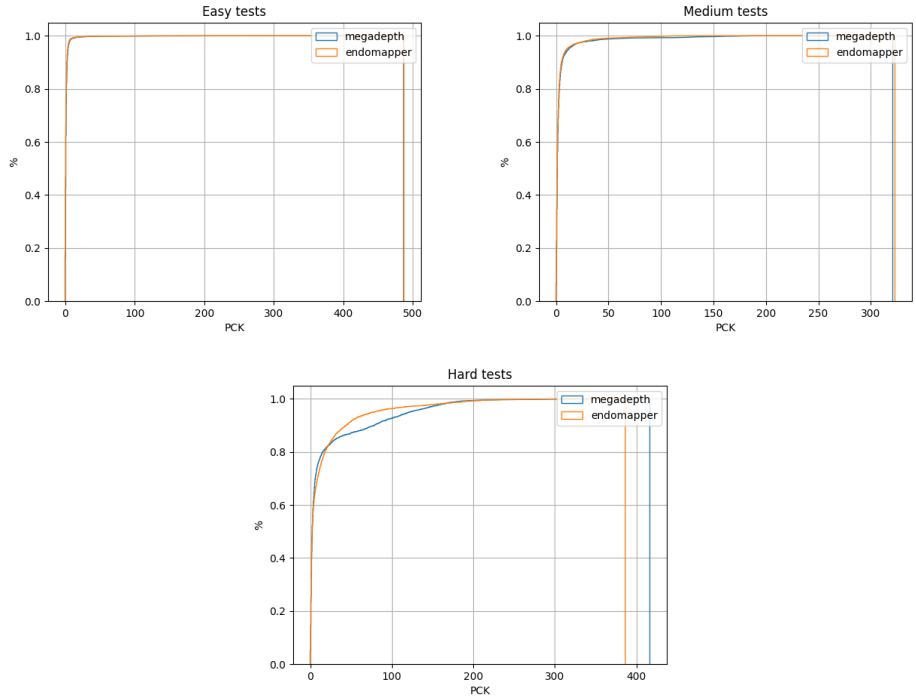


Figura 4.5: Curvas de los histogramas acumulados de cada categoría de test. En cada uno se muestra para cada valor de PCK el correspondiente porcentaje de puntos

test (Ver Sección 3.2).

4.5. Coste computacional

Con el objetivo de conocer el coste computacional de WarpC+GLUNet, se han utilizado 2 máquinas en las que se ha ejecutado tanto el entrenamiento del modelo "endomapper" (Sección 4.3) como la inferencia del modelo una vez entrenado. Las características principales de las 2 máquinas son las siguientes:

- Ordenador robot-19
  CPU: Intel i7-9700K, 8 cores, 8 hilos
  RAM: 32 GB
  GPU: 1x NVIDIA TITAN V con GRAM de 12 GB

- Estación de trabajo DGX
  CPU: Intel(R) Xeon(R) CPU E5-2698 v4, 40 cores, 80 hilos
  RAM: 500 GB
  GPU: 8x Tesla V100-SXM2 con GRAM de 32 GB

El tiempo de ejecución del entrenamiento del modelo "endomapper" fue de 3 días en robot-19 (utilizando su única GPU) y 2.5 días en DGX (utilizando 2 de sus 8 GPUs).

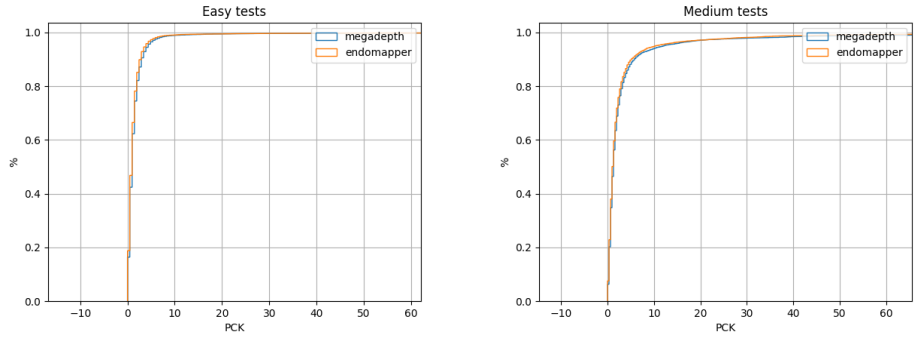
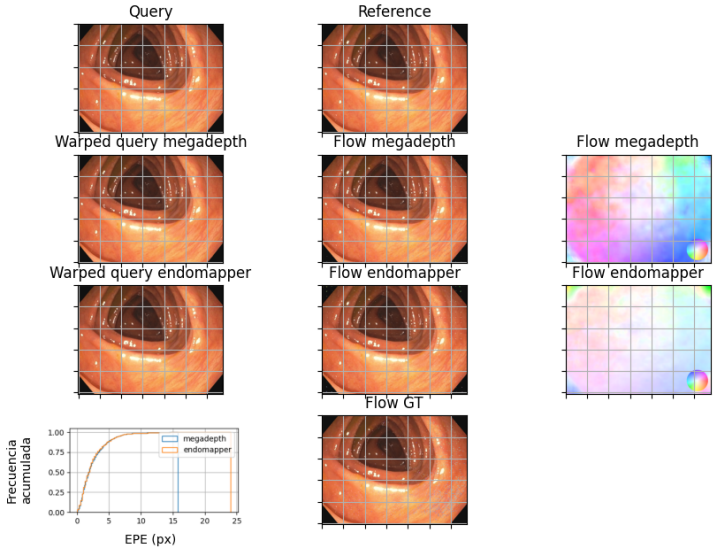
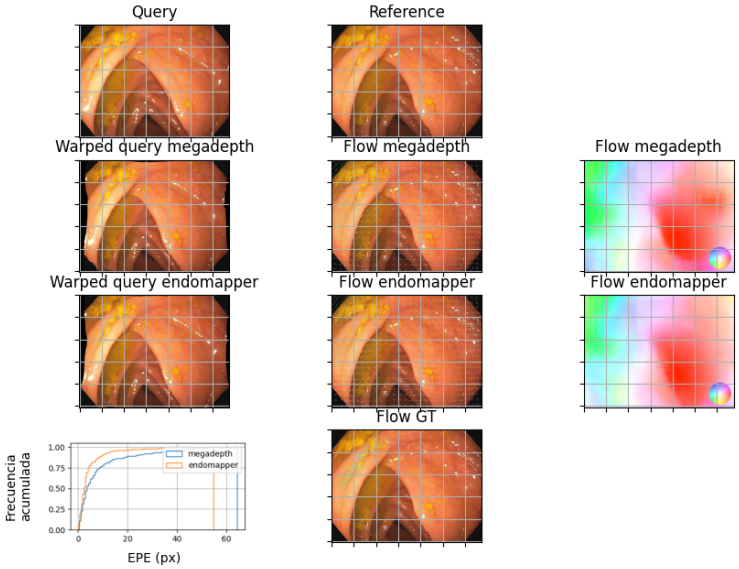


Figura 4.6: Curvas de los histogramas acumulados para las categorías de test fácil (izquierda) y medio (derecha) con la región de PCKs menores que 60 ampliada

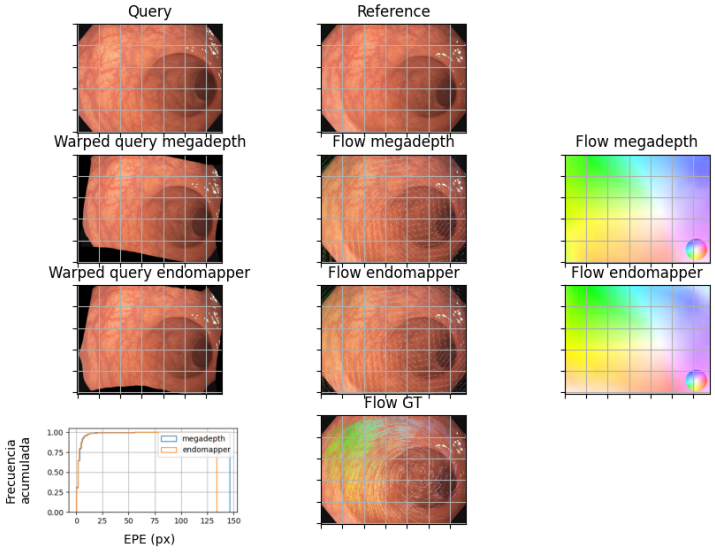
No obstante, dado que se pueden utilizar las 8 GPUs de la máquina DGX de forma independiente, se pueden realizar 4 entrenamientos de forma simultánea. Por otra parte, el tiempo de inferencia es el mismo tanto para la máquina robot-19 como para DGX. Para calcular el flujo entre 2 imágenes del Endomapper Dataset [1] (las cuales tienen un tamaño de 1350x1012 píxeles), tarda 2 segundos en cargar el modelo entrenado y otros 2 segundos en calcular el flujo. Sin embargo, si se calculan más flujos de forma sucesiva, el tiempo de estimación de los mismos será de 1 segundo (en lugar de 2 segundos). Esto significa que es más eficiente calcular el flujo de varias imágenes de forma consecutiva en lugar de calcularlo de forma aislada. Además, se ha comprobado que el tamaño de las imágenes afecta al tiempo de estimación del flujo. Al reducir las imágenes a la mitad de tamaño (675x506 píxeles), el tiempo de carga del modelo no se vio afectado. Sin embargo, el tiempo de cálculo del primer flujo se redujo de 2 segundos a 1 segundo y los tiempos de cálculo de flujos consecutivos se redujeron de 1 segundo a 0.5 segundos.



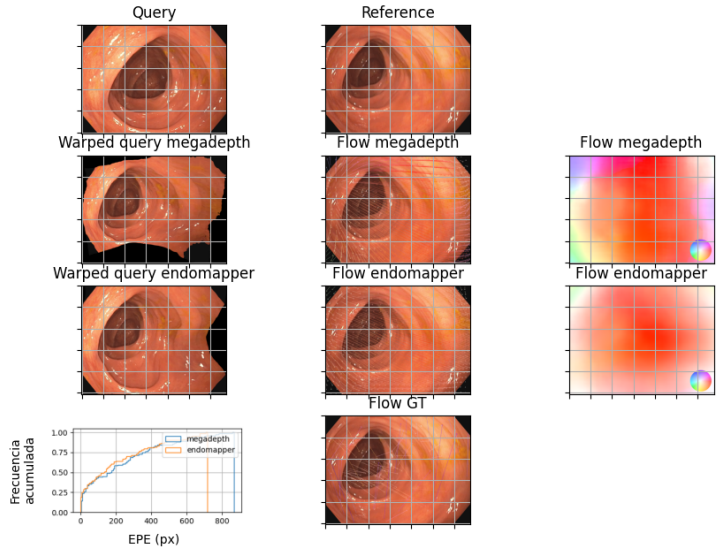
(a) Test más fácil



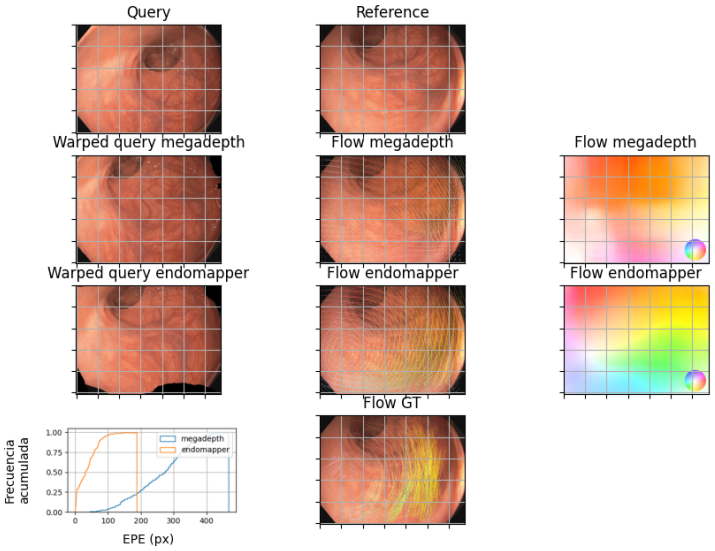
(b) Test fácil mediano



(c) Test medio mediano



(a) Test difícil mediano



(b) Test más difícil

Figura 4.7: Tests realizados para comparar los modelos "megadepth" (Sección 4.2) y "endomapper" (Sección 4.3). Para cada test, se muestra en la primera fila, su imagen de query y su imagen de referencia; en la segunda fila, la warped query tanto, el flujo con flechas y el flujo con mapa de color) tanto para el modelo "megadepth" como con el modelo "endomapper"; y en la última fila, se muestran la curva del histograma acumulado del EPE de cada modelo y el flujo sparse ground-truth

Capítulo 5

Conclusiones

Este trabajo supone una de las primeras aproximaciones a la aplicación de técnicas de flujo no supervisado en el dominio de las colonoscopias. Se ha evaluado el método de entrenamiento WarpC [5] y la red de estimación de flujo GLUNet [6] en secuencias del Endomapper Dataset [1], mostrando un rendimiento sorprendente para relacionar imágenes mediante flujo denso.

El método tiene gran potencial para encontrar correspondencias en colosnocopias, donde otros métodos fallan por completo. Primero se ha evaluado el modelo que había sido entrenado previamente en megadepth (Sección 4.2), mostrando un desempeño excelente para flujos pequeños y medios. En el trabajo, se ha propuesto un entrenamiento que, usando datos de colonoscopias y una serie de modificaciones, permiten mejorar el funcionamiento ligeramente, especialmente en el caso de que el flujo entre las imágenes sea elevado, donde la mejora es más significativa (Figura 4.7)..

5.1. Trabajo futuro

Este trabajo es un primer paso en la dirección para obtener correspondencias densas, y por tanto, todavía tiene varias áreas de mejora:

- Podría esperarse una mayor mejora tras hacer la adapción al dominio del colon respecto del modelo entrenado en Megadepth. Se podría estudiar el utilizar más secuencias de entrenamiento, así como variar los hiperparámetros de la red hasta obtener la mejor configuración.

- No se han considerado explícitamente los retos que presentan las secuencias de colonoscopia, como las especularidades o deformaciones. Añadir términos a la *loss* que tengan esto en cuenta podría mejorar los resultados.

Por otra parte, debido al hecho de que el entrenamiento del modelo WarpC+GLUNet es paralelizable, se podría diseñar un sistema distribuido para reducir el tiempo de en-

trenamiento. De esta forma, se podría realizar un mayor número de entrenamientos en el mismo periodo de tiempo.

Bibliografía

[1] Pablo Azagra, Carlos Sostres, Ángel Ferrandez, Luis Riazuelo, Clara Tomasini, Oscar León Barbed, Javier Morlana, David Recasens, Victor M. Batlle, Juan J. Gómez-Rodríguez, Richard Elvira, Julia López, Cristina Oriol, Javier Civera, Juan D. Tardós, Ana Cristina Murillo, Angel Lanas, and José M. M. Montiel. EndoMapper dataset of complete calibrated endoscopy procedures. apr 2022.

[2] Denis Fortun, Patrick Bouthemy, and Charles Kervrann. Optical flow modeling and computation: A survey. *Computer Vision and Image Understanding*, 134:1–21, may 2015.

[3] Eddy Ilg, Nikolaus Mayer, Tonmoy Saikia, Margret Keuper, Alexey Dosovitskiy, and Thomas Brox. FlowNet 2.0: Evolution of Optical Flow Estimation with Deep Networks. *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, 2017-January:1647–1655, nov 2017.

[4] Zachary Teed and Jia Deng. RAFT: Recurrent All-Pairs Field Transforms for Optical Flow. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 12347 LNCS:402–419, 2020.

[5] Prune Truong, Martin Danelljan, Fisher Yu, and Luc Van Gool. Warp Consistency for Unsupervised Learning of Dense Correspondences. *Proceedings of the IEEE International Conference on Computer Vision*, pages 10326–10336, apr 2021.

[6] Prune Truong, Martin Danelljan, and Radu Timofte. GLU-Net: Global-Local Universal Network for Dense Flow and Correspondences, 2020.

[7] Zhengqi Li and Noah Snavely. MegaDepth: Learning Single-View Depth Prediction From Internet Photos, 2018.

[8] Johannes L. Schonberger and Jan-Michael Frahm. Structure-From-Motion Revisited, 2016.

[9] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. ImageNet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, may 2017.

[10] Alexey Dosovitskiy, Philipp Fischer, Eddy Ilg, P. Häusser, C. Hazırbaş, V. Golkov, P. Smagt, D. Cremers, and Thomas Brox. FlowNet: Learning Optical Flow with Convolutional Networks, 2015.

[11] Simon Baker and Iain Matthews. Lucas-Kanade 20 years on: A unifying framework. *International Journal of Computer Vision*, 56(3):221–255, feb 2004.

[12] Jerome Revaud, Philippe Weinzaepfel, Zaid Harchaoui, and Cordelia Schmid A A Inria. EpicFlow: Edge-Preserving Interpolation of Correspondences for Optical Flow, 2015.

Lista de Figuras

Lista de Tablas

Anexos A

Dedicación al proyecto

El tiempo dedicado al proyecto se puede encontrar en la Tabla A.1.
Por otra parte, el código del mismo está disponible en el siguiente repositorio de GitHub: https://github.com/UZ-SLAMLab/WarpC-GLUNet.

Tiempo dedicado	
Tarea	Horas
Instalación WarpC+GLUNet	40
Pruebas iniciales con el modelo “megadepth”	40
Adaptación al dominio de Endomapper (generación de datos de train y validación)	70
Diseño del entrenamiento del sistema	70
Entrenamiento en la estación de trabajo DGX con 8 GPUs	60
Evaluación del sistema (generación de datos de test)	60
Análisis de mejoras	60
Elaboración de la memoria	50
Total	450

Tabla A.1: Tiempo dedicado a cada una de las tareas del proyecto