

CARRERA DE MAESTRÍA EN INGENIERÍA

**APLICACIÓN DE VELOCIMETRÍA POR IMÁGENES DE
PARTÍCULAS A FLUJOS SOBRE MEDIOS POROSOS**

Ivan Carovano
Maestrando

Nicolás Silin
Director

Miembros del jurado:

Dr. F.E. Teruel (Instituto Balseiro, Centro Atómico Bariloche)
Mgter. O.D. Osorio (Instituto Balseiro, Centro Atómico Bariloche)
Dr. Ing. R.G. Martino (CONICET, UNC)

Febrero 2023

Instituto Balseiro
Universidad Nacional de Cuyo
Comisión Nacional de Energía Atómica
Argentina

(Biblioteca Leo Falicov CAB-IB)

ÍNDICE

RESUMEN	3
ABSTRACT	4
I. INTRODUCCIÓN	5
I.1. MOTIVACIÓN.....	5
I.2. PERMEABILIDAD	6
I.3. VELOCIMETRÍA POR IMÁGENES DE PARTÍCULAS	7
I.4. ESTRUCTURA DEL TRABAJO	7
II. MÉTODO PIV	8
II.1. PREPROCESAMIENTO DE IMAGEN	9
II.2. ENMASCARAMIENTO	10
II.3. ESTABLECIMIENTO DEL TAMAÑO DE VENTANAS DE INTERROGACIÓN	12
II.4. EMPLEO DE FUNCIONES DE VENTANEO	14
II.5. ESTABLECIMIENTO DE LA GRILLA	15
II.6. CORRELACIÓN CRUZADA.....	15
II.6.1. <i>Correlación cruzada directa</i>	16
II.6.2. <i>Correlación cruzada estándar</i>	18
II.6.3. <i>Mínimos cuadrados</i>	19
II.6.4. <i>Correlación cruzada normalizada</i>	20
II.7. ESTIMACIÓN SUBPÍXEL	22
II.7.1. <i>Determinación del centroide</i>	22
II.7.2. <i>Interpolación lineal</i>	22
II.7.3. <i>Ajuste parabólico de tres puntos</i>	23
II.7.4. <i>Ajuste gaussiano de tres puntos</i>	24
II.7.5. <i>Ajuste gaussiano de cuatro puntos</i>	25
II.7.6. <i>Estimadores basados en el desarrollo en series de Taylor</i>	26
II.7.7. <i>Ajuste a superficie parabólica</i>	27
II.7.8. <i>Ajuste a superficie gaussiana por mínimos cuadrados no lineales</i>	28
II.7.9. <i>Ajuste a superficie gaussiana linealizada por mínimos cuadrados</i>	28
II.7.10. <i>Estimación por medio del flujo óptico</i>	30
II.7.11. <i>Ajuste de mínimos cuadrados sobre las ventanas de interrogación</i>	30
II.8. VALIDACIÓN	31
II.8.1. <i>Validación por medio de valor umbral</i>	32
II.8.2. <i>Detectores basados en la media y la mediana</i>	32
II.8.3. <i>Detección basada en redes neuronales</i>	34
II.8.4. <i>Bootstrapping</i>	37
II.8.5. <i>Método todo en uno</i>	38
II.9. REEMPLAZO DE VECTORES ESPURIOS.....	39
II.10. SUAVIZADO	40
II.11. PROCESAMIENTO DE PASOS MÚLTIPLES	40
II.11.1. <i>Desplazamiento de ventanas discreto</i>	41
II.11.2. <i>Deformación de imágenes</i>	42
III. SUPER RESOLUCIÓN	44
IV. SOFTWARE IMPLEMENTADO	47
V. ANÁLISIS DE ERRORES Y DESEMPEÑO	53
V.1. COMPARACIÓN ENTRE LA CORRELACIÓN DIRECTA Y LA CORRELACIÓN ESTÁNDAR	54
V.2. EFECTO DEL TAMAÑO DE VENTANA EN EL ERROR ALEATORIO.....	56
V.3. EFECTOS DEL GRADIENTE DE DESPLAZAMIENTO DENTRO DE LAS VENTANAS	58

V.4.	EFFECTOS DEL RELLENADO CON CEROS Y EL VENTANEO EN EL ERROR SISTEMÁTICO Y ALEATORIO	59
V.5.	COMPARACIÓN ENTRE LA CORRELACIÓN CRUZADA ESTÁNDAR, LA CORRELACIÓN NORMALIZADA Y EL MÉTODO DE MÍNIMOS CUADRADOS	62
V.6.	ERROR DEBIDO AL EFECTO DE ENCLAVAMIENTO DE PICOS	64
V.7.	DIÁMETRO ÓPTIMO DE PARTÍCULAS CON DISTRIBUCIÓN GAUSSIANA PARA LA ESTIMACIÓN BASADA EN EL AJUSTE GAUSSIANO DE TRES PUNTOS	69
V.8.	COMPARACIÓN ENTRE LOS DISTINTOS MÉTODOS DE ESTIMACIÓN SUBPÍXEL.....	71
V.9.	ERROR ASOCIADO A LOS COMPONENTES ESTACIONARIOS DE LA IMAGEN	73
V.10.	DESEMPEÑO DEL MÉTODO SUPER RESOLUCIÓN	76
VI.	ANÁLISIS CASO PERMEABLE	80
VII.	CONCLUSIONES	98
VIII.	REFERENCIAS	100
IX.	ANEXO	105
IX.1.	TRANSFORMADA DE FOURIER DISCRETA EN EL TIEMPO	105
IX.2.	TRANSFORMADA DE FOURIER DISCRETA	105
IX.3.	NATURALEZA PERIÓDICA DE LA TRANSFORMADA DE FOURIER.....	106
IX.4.	TEOREMA DE LA CONVOLUCIÓN	107
IX.5.	TRANSFORMADA DE FOURIER EN DOS DIMENSIONES	107
IX.6.	EXPRESIÓN ANALÍTICA PARA LA FUNCIÓN DE CORRELACIÓN DE UNA PARTÍCULA CON DISTRIBUCIÓN GAUSSIANA	108
IX.7.	IMPLEMENTACIÓN DEL ALGORITMO DE LEVENBERG-MARQUARDT PARA EL AJUSTE A UNA SUPERFICIE GAUSSIANA	109
IX.8.	DERIVACIÓN DEL MÉTODO DE ESTIMACIÓN SUBPÍXEL BASADO EN EL FLUJO ÓPTICO.....	111
IX.9.	APLICACIÓN DEL MÉTODO DE VALIDACIÓN TODO EN UNO.....	113
IX.10.	GENERACIÓN DE IMÁGENES SINTÉTICAS Y MÉTODO DE MONTECARLO	117
IX.11.	ANÁLISIS DEL CÓDIGO PRANA.....	118

RESUMEN

En el presente trabajo fue abordada la problemática de la medición del campo de velocidades de un fluido que circula a través de un medio poroso, mediante la técnica de Velocimetría por Imágenes de Partículas (PIV) en un plano. Se llevó a cabo un estudio exhaustivo de esta técnica, con la finalidad de desarrollar un conjunto de herramientas de procesamiento, buscando optimizar la aplicación del método PIV al caso particular de flujo en medios porosos. Estas herramientas fueron incorporadas a un software con interfaz de usuario que también fue desarrollado en el marco del trabajo. Se llevaron a cabo ensayos por medio de imágenes sintéticas y el método de Montecarlo para caracterizar el desempeño de los distintos algoritmos implementados. También se realizaron mediciones experimentales aplicando las herramientas desarrolladas. Los medios porosos empleados en los experimentos fueron producidos mediante impresión 3D. Estos medios fueron caracterizados a partir del coeficiente de permeabilidad de Darcy. Los resultados obtenidos demuestran la aplicabilidad del modelo de Darcy en las condiciones estudiadas. A su vez, se demuestra la factibilidad para la medición del campo de velocidades dentro de los poros del medio. Se comprobó que el uso de técnicas como la aplicación de máscaras, técnica de mascara digital y el método PIV Super Resolución permiten resolver algunas de las dificultades que surgen en la determinación del campo de velocidades dentro de los poros.

ABSTRACT

In the present work, the problem of measuring the velocity field of a fluid that circulates through a porous medium was addressed, using the Particle Imaging Velocimetry (PIV) technique in a plane. An exhaustive study of this technique was carried out, in order to develop a set of processing tools, looking forward to the optimization of the PIV method application to the particular case of flow in porous media. These tools were incorporated into a software with a user interface that was also developed in the framework of the work. Tests were carried out using synthetic images and the Monte Carlo method to characterize the performance of the different implemented algorithms. Experimental measurements were also carried out applying the developed tools. The porous media used in the experiments were produced by 3D printing. These were characterized using the Darcy coefficient of permeability. The results obtained demonstrate the applicability of the Darcy model in the studied conditions. In turn, the feasibility for the measurement of the velocity field within the pores of the medium is demonstrated. It was verified that the use of techniques such as the application of masks, the digital mask technique and the Super Resolution PIV method allows to solve some of the difficulties that arise in the determination of the velocity field inside the pores.

I. INTRODUCCIÓN

En la naturaleza y la ingeniería se encuentran diversos materiales que presentan porosidad y que son permeables a los fluidos, y que en el contexto de la Mecánica de Fluidos se denominan Medios Porosos. Estos materiales constituyen un caso particular de estudio ya que, si bien el movimiento del fluido está bien descrito por la ecuación de Navier Stokes, por lo general el análisis detallado del movimiento en los poros no es de interés y excede la capacidad de análisis de las técnicas disponibles. De esta forma el comportamiento se suele modelar considerando propiedades del flujo promediadas espacial y temporalmente. En el presente trabajo se buscará desarrollar la herramienta experimental de Velocimetría por Imágenes de Partículas (o PIV por sus siglas en inglés), de forma que permita analizar los flujos complejos que involucran tanto las zonas porosas como de flujo libre. El objetivo es que esta herramienta permita estudiar este tipo de medios y sus efectos sobre las propiedades del flujo, en pos de explorar sus potenciales aplicaciones tecnológicas.

I.1. MOTIVACIÓN

Uno de los contextos de la ingeniería en donde se ha explorado el uso de medios porosos es en el transporte de fluidos a través de conductos. El transporte de fluidos en conductos es una necesidad común en el ámbito de la ingeniería y la industria, y el diseño eficiente de estos sistemas de transporte permite reducir el costo energético, asociado a los sistemas de bombeo requeridos para impulsar estos fluidos y compensar las pérdidas de presión. Las pérdidas de presión a lo largo de una cañería son originadas por la interacción del fluido con las paredes del canal y sus accesorios, y por los efectos de disipación viscosa, relacionados con la turbulencia del flujo. Esto ha motivado a distintos investigadores a estudiar el flujo de fluidos sobre superficies con distintas características, con el objetivo de determinar los efectos de estas superficies sobre el flujo, e inspirar el diseño de superficies novedosas que constituyan mejoras tecnológicas al modificar favorablemente el comportamiento del flujo cerca de las paredes. En estudios recientes (Rosti, Brandt, & Pinelli, 2018) se ha demostrado por medio de simulaciones numéricas, que la fricción para el flujo en una cañería puede ser reducida si la pared del conducto es revestida con un material poroso anisotrópico. En el presente se han realizado algunos esfuerzos por explorar este hecho en forma experimental, pero no se ha observado aún una mejora según las predicciones de los modelos teóricos (Suga, Okazaki, Matsuo, Taneo, & Kuwata, 2019; Chavarin, Efstathiou, Vijay, & Luhar, 2020). El principal objetivo del presente trabajo es continuar en esa línea de investigación, buscando validar experimentalmente las observaciones mencionadas. Para tal fin, existe una línea de trabajo que propone llevar a cabo una serie de experimentos utilizando un canal revestido con un medio poroso anisotrópico. El efecto de los medios sobre el flujo se caracterizará por medio de mediciones mediante la técnica de Velocimetría por Imágenes de Partículas (PIV). Sin embargo, la aplicación de esta técnica en geometrías altamente complejas, como las que involucran medios porosos, presentan un gran desafío. El objetivo del presente trabajo es realizar un

análisis de la técnica de PIV enfocado en lograr minimizar los errores de medición del campo de velocidades en flujos con presencia de medios porosos.

I.2. PERMEABILIDAD

La permeabilidad puede definirse como la propiedad de los medios porosos que caracteriza su capacidad para permitir el flujo de un fluido a través de él, cuando es aplicado un gradiente de presión (Mitra, de Castro, & El Mansori, 2019). Para el modelado de la permeabilidad a través de medios porosos con fluidos newtonianos en régimen laminar estacionario, la ley de Darcy es ampliamente utilizada. Esta ley establece que el caudal volumétrico que fluye a través de un medio poroso es directamente proporcional a la sección transversal al flujo y al gradiente de presión existente a través del medio poroso, e inversamente proporcional a la viscosidad dinámica del fluido, lo cual se puede expresar a través de la ecuación

$$Q = k \frac{A \Delta P}{\mu L},$$

donde Q es el caudal volumétrico, A es la sección transversal al flujo, ΔP es la caída de presión a través del medio poroso, μ es la viscosidad dinámica del fluido, L es la longitud del medio poroso en la dirección principal del flujo, y finalmente, la constante de proporcionalidad k es el coeficiente de permeabilidad. Esta ley fue primero formulada por Darcy (1856), basándose en el resultado de sus experimentos sobre el flujo de agua en lechos de arena, y posteriormente derivada de forma teórica a partir de las ecuaciones de Navier-Stokes (Irmay, 1958; Neuman, 1977; Whitaker, 1986).

En el análisis de medios porosos con baja permeabilidad, muchas veces es suficiente esta ecuación. Sin embargo, en casos en que existe alta permeabilidad, un análisis más completo requiere considerar otras tensiones viscosas, términos inerciales y la fracción de volumen ocupada por el sólido. Como resultado de considerar estos fenómenos se obtiene la ecuación de Brinkman (Durlinsky & Brady, 1987), una ecuación similar a la de Navier Stokes, pero modificada para considerar la porosidad del medio en los términos inerciales y para incluir una fuerza volumétrica lineal con la velocidad equivalente a la ecuación de Darcy.

La motivación de este trabajo es generar las herramientas experimentales para poder analizar flujos complejos que involucran tanto zonas porosas como de flujo libre, con el propósito de evaluar el uso de modelos sencillos como el de Darcy, así como más complejos como el de la ecuación de Brinkman. En principio, tal como se mostrará más adelante, es posible determinar el coeficiente de permeabilidad a partir del conocimiento del campo de velocidades. Se propuso implementar la técnica experimental conocida como Velocimetría por Imágenes de Partículas para la estimación de este campo.

I.3. VELOCIMETRÍA POR IMÁGENES DE PARTÍCULAS

La Velocimetría por Imágenes de Partículas o PIV (del inglés, Particle Image Velocimetry) es una técnica ampliamente utilizada, y no intrusiva, que permite estimar la velocidad del fluido en distintos puntos con buena resolución espacial y temporal (Raffel, y otros, 2018; Willert & Gharib, 1991; Westerweel, 1995; Grant, 1997; Westerweel, 1997). La técnica consiste en medir la velocidad del fluido de forma indirecta a partir de la medición de la velocidad de partículas previamente sembradas en el flujo. Esta velocidad es determinada utilizando una técnica por imágenes. Las partículas en el flujo son iluminadas, y una serie de imágenes son capturadas a intervalos cortos de tiempo. El campo de velocidades es determinado entonces teniendo en cuenta el desplazamiento relativo de partículas entre cuadros sucesivos. Una descripción más detallada del método se presenta más adelante.

Debido al desafío adicional que presenta la medición de velocidades dentro del medio poroso, en el que se desarrollan zonas con pobre iluminación y depleción de partículas, se planteó la necesidad de realizar un estudio exhaustivo de la técnica PIV, buscado aplicar los algoritmos y técnicas de procesamiento más modernas, a fines de obtener los resultados con el mayor nivel de fidelidad posible. A su vez, se propone el estudio de la técnica conocida como PIV Super Resolución, como método para resolver las posibles dificultades en la estimación de las velocidades dentro del medio poroso. A lo largo del trabajo se exponen las distintas técnicas estudiadas, y se presentan resultados comparativos obtenidos por ensayos de Montecarlo, que sirven para identificar los métodos óptimos para cada etapa del procesamiento.

I.4. ESTRUCTURA DEL TRABAJO

El trabajo se estructura de la siguiente manera. En la sección II se desarrolla en detalle el método de Velocimetría por Imágenes de Partículas, que incluye una descripción de cada una de las etapas del procesamiento, y los algoritmos propuestos en la bibliografía para cada una de estas etapas. En la sección III se describe el método de Super Resolución. En la sección IV se describe el software implementado para llevar a cabo el procesamiento en este trabajo. En la sección V se reportan una serie de resultados obtenidos por simulaciones de Montecarlo a partir de imágenes sintéticas, donde se ponen a prueba y comparan los distintos algoritmos de procesamiento PIV descritos en la sección V. En la sección VI se detalla el experimento realizado y se reportan los resultados obtenidos. Finalmente, en la sección VII se comparten las conclusiones.

II. MÉTODO PIV

La Velocimetría por imágenes de partículas o PIV (del inglés, Particle Image Velocimetry) es una técnica experimental cuantitativa no intrusiva para la determinación del campo de velocidades de flujos de fluidos, que permite efectuar mediciones con una gran resolución espacial y temporal (Raffel, y otros, 2018; Willert & Gharib, 1991; Westerweel, 1995; Grant, 1997; Westerweel, 1997). La técnica consiste en procesar una serie de imágenes de partículas microscópicas, las cuales son previamente sembradas en el fluido para su visualización. En la implementación más sencilla de PIV, conocida como PIV plano o 2D2C-PIV, las partículas son iluminadas por medio de un haz de luz plano, emitido en forma pulsada, y las imágenes son capturadas por medio de una cámara de alta velocidad, apuntando de forma normal al plano de luz. Efectuadas las capturas de esta forma, las partículas solo se observan en el plano iluminado. Finalmente, a partir de dos imágenes consecutivas separadas por un pequeño intervalo de tiempo, y haciendo uso del desplazamiento relativo de las partículas, es posible inferir el campo de velocidades proyectado en dicho plano.

Para determinar el campo de velocidades, se emplea un método estadístico basado en la estimación del desplazamiento de las imágenes de un grupo de partículas. En primer lugar, se define una grilla regular que determina los puntos del plano de la imagen en los cuales se evaluarán los vectores de desplazamiento. A continuación, se toma un par de capturas consecutivas, las cuales se subdividen en áreas más pequeñas conocidas como ventanas de interrogación, centradas en los puntos de grilla. Las ventanas de interrogación correspondientes al mismo punto de grilla en la primera y la segunda imagen son procesadas por medio de una técnica de correlación cruzada (sección II.6), obteniéndose la función de correlación, la cual refleja el grado de coherencia entre el contenido de las ventanas para distintos desplazamientos de una respecto a la otra. El desplazamiento entre partículas más probable es estimado a partir del máximo de esta función. El proceso se repite para cada punto de grilla para conformar el campo de desplazamientos. El campo de velocidades se puede derivar a partir de este último, conociendo el intervalo de tiempo Δt entre los pulsos de luz y el factor de magnificación de la imagen.

La función de correlación es calculada para desplazamientos discretos, por lo que el desplazamiento obtenido a partir del máximo de esta función tiene una precisión de $\pm 1/2$ píxeles. No obstante, una estimación con mayor precisión puede realizarse a partir del empleo de la información de los valores de la función de correlación en los puntos vecinos al pico máximo. El procedimiento empleado para tal fin se conoce como estimación subpíxel (sección II.7).

Para un procesamiento satisfactorio de las imágenes, estas deben poseer una calidad adecuada. Las características deseables en las imágenes para la aplicación de PIV son principalmente el buen contraste entre las imágenes de partículas y el fondo, una iluminación homogénea y una buena población de partículas (alrededor de 10 partículas por ventana de interrogación). En la práctica, sin embargo, la calidad de las imágenes

puede verse negativamente afectada por diversos factores locales. Algunos de estos son las zonas con densidad de partículas deficiente, los movimientos fuera del plano, los cambios de iluminación entre cuadros, y las zonas saturadas por reflexiones de luz en límites sólidos, entre otros. Cabe destacar que en el caso de PIV en medios porosos, la existencia de sólidos en el campo de la imagen causa que estos problemas se vean acrecentados. Debido a estos factores, puede ocurrir que la evaluación en determinados puntos de la grilla resulte en planos de correlación con una menor relación señal-ruido, o con picos espurios. Estos son picos adicionales que aparecen en el plano de correlación, con intensidades semejantes a la intensidad del pico correspondiente al desplazamiento medio en dicho punto. Si la intensidad de alguno de estos picos espurios supera a la intensidad del pico correspondiente al desplazamiento medio en la región, la estimación del desplazamiento basada en el pico máximo resultará en un valor incorrecto. Estos vectores incorrectos, denominados vectores espurios, son un problema en pasos posteriores de procesamiento como en el cálculo de cantidades derivadas, y pueden generar propagación de errores en el caso de los métodos de pasos múltiples (sección II.11). Por lo tanto, estos deben ser identificados y reemplazados. Para ello, luego de estimar el campo de desplazamientos, este debe ser verificado mediante un algoritmo de validación (sección II.8). Posteriormente, se emplea alguna técnica para reemplazar los vectores espurios (sección II.9). Finalmente, suele aplicarse algún procedimiento de suavizado para atenuar el nivel de ruido presente en el campo de desplazamientos resultante debido al error aleatorio en las estimaciones (sección II.10).

La resolución espacial alcanzable mediante PIV está limitada por el tamaño de las ventanas de interrogación, el cual a su vez se encuentra limitado por una serie de factores que tienen que ver principalmente con la amplitud de los desplazamientos, los gradientes de desplazamiento y la densidad de partículas (esto se discute en la sección II.3). No obstante, una mayor resolución espacial puede alcanzarse mediante el empleo de los métodos de pasos múltiples (sección II.11). En estos esquemas, se implementan dos o más pasos PIV sucesivos, tomando el campo de desplazamientos estimado en el paso previo como entrada para aplicar, ya sea un desplazamiento discreto en las ventanas, o bien una deformación en las imágenes. Esto permite utilizar ventanas de interrogación más pequeñas en los pasos siguientes sin sufrir un aumento en la cantidad de vectores espurios, logrando una mayor resolución espacial. En las secciones siguientes se analizan cada una de las etapas de procesamiento PIV.

II.1. PREPROCESAMIENTO DE IMAGEN

Uno de los aspectos que influye en la precisión de los resultados PIV es la calidad de las imágenes, las cuales constituyen la principal información de entrada del procesamiento. Idealmente, las imágenes para aplicaciones PIV deberán consistir en partículas brillantes con un diámetro de 2 a 3 píxeles (Raffel, y otros, 2018), impuestas en un fondo oscuro. En contraste, si las imágenes contienen iluminación proveniente de objetos estacionarios, esta se correlacionará consigo misma durante el procesamiento,

resultando en picos de correlación con un sesgo hacia el desplazamiento nulo. Una técnica satisfactoria para corregir este fenómeno es la sustracción de fondo.

La sustracción de fondo consiste en eliminar la información de las imágenes correspondiente a los componentes estacionarios (fondo) de las mismas, con el objetivo de obtener unas imágenes que posean solo la información de las partículas. Una forma de sustracción de fondo es la nivelación, la cual se basa en restar de cada imagen PIV un mapa de intensidades de referencia, representativo del fondo. La imagen de fondo puede ser obtenida a partir de capturas sobre el flujo iluminado sin partículas. También puede ser construida aplicando un filtro pasa bajos espacial sobre las imágenes (Willert, 1997), o tomando cada imagen de la secuencia como la imagen de referencia de la siguiente (Honkanen & Nobach, 2005; Mejia-Alvarez & Christensen, 2013). Si se dispone de un gran número de capturas, la imagen de fondo puede ser generada a partir de la media temporal (Wereley, Gui, & Meinhart, 2002), o del mínimo histórico (Gui, 1997) de toda la serie. Alternativamente, las imágenes de partículas pueden ser discriminadas a partir de considerar la distribución de probabilidad de intensidades (Stitou & Riethmuller, 2001). Por último, la información de las partículas puede ser extraída aplicando sobre la serie de imágenes un filtro pasa altos en el dominio temporal (Sciacchitano & Scarano, 2014), o aplicando Descomposición Ortogonal Propia (Mendez, y otros, 2017).

II.2. ENMASCARAMIENTO

En toda serie de capturas PIV existen regiones de las imágenes que no aportan información útil, ya sea porque representan zonas que no son de interés, o bien porque no contienen partículas (paredes del canal, cuerpos sólidos, regiones externas al flujo, etc.). También podría ocurrir que ciertas regiones sean propensas a producir vectores espurios, ya sea por contener baja densidad de partículas, pobre iluminación (regiones sombreadas), o por ser regiones que están saturadas en intensidad (reflexión de la luz en objetos sólidos). En estos casos, realizar el procesamiento PIV en los puntos de la grilla dentro de estas regiones solo implicaría un mayor costo computacional sin ningún beneficio. Resulta conveniente entonces enmascarar estas regiones para evitar el procesamiento en dichas zonas.

Una máscara es una imagen binaria, cuyas dimensiones coinciden con las dimensiones de las imágenes de la serie PIV a procesar, y que contiene ceros en los píxeles correspondientes a las regiones de no interés antes mencionadas, y unos en los píxeles restantes. Esta máscara se puede usar entonces como entrada adicional en el procesamiento PIV para indicar sobre qué puntos de grilla realizar el procesamiento. En métodos de procesamiento avanzado, la máscara también puede aplicarse en la etapa de correlación cruzada para mejorar la estimación del desplazamiento en lo que se conoce como técnica de máscara digital (Gui, Merzkirch, & Fei, 2000; Gui, Wereley, & Kim, 2003). Esta técnica se detalla en la sección II.6.

En el caso de imágenes con objetos móviles, se requerirá una máscara diferente para cada imagen de la serie (enmascaramiento dinámico). Pero, en secuencias de imágenes donde las zonas a enmascarar permanecen estáticas, basta con una sola máscara para toda la serie (enmascaramiento estático). Este último tipo de enmascaramiento es el de interés en este trabajo.

Una máscara estática puede generarse de forma manual. También existen métodos de procesamiento que permiten generar una máscara utilizando la serie de imágenes como entrada. Entre las técnicas propuestas en la bibliografía, se encontró conveniente el empleo de un método basado en el análisis estadístico de cada píxel en toda la serie de imágenes (Masullo & Theunissen, 2017). Esta técnica se fundamenta en el hecho de que los píxeles correspondientes a las regiones de la imagen que no presentan flujo de partículas poseen una distribución de intensidades diferente (variaciones debidas principalmente al ruido de la cámara, histograma con distribución normal) que aquellos píxeles pertenecientes a las regiones con flujo de partículas (fuertes cambios debido al paso de las partículas, histograma con gran asimetría). Esto se aprecia en los histogramas de la Figura 1, realizados muestreando píxeles sobre zonas con y sin flujo de partículas en una secuencia de imágenes experimentales, posterior al procedimiento de sustracción de fondo. Haciendo uso de este rasgo diferencial en las distribuciones, y aplicando una prueba de normalidad sobre la historia temporal de cada píxel, es posible discriminar entre estos dos tipos de zonas.

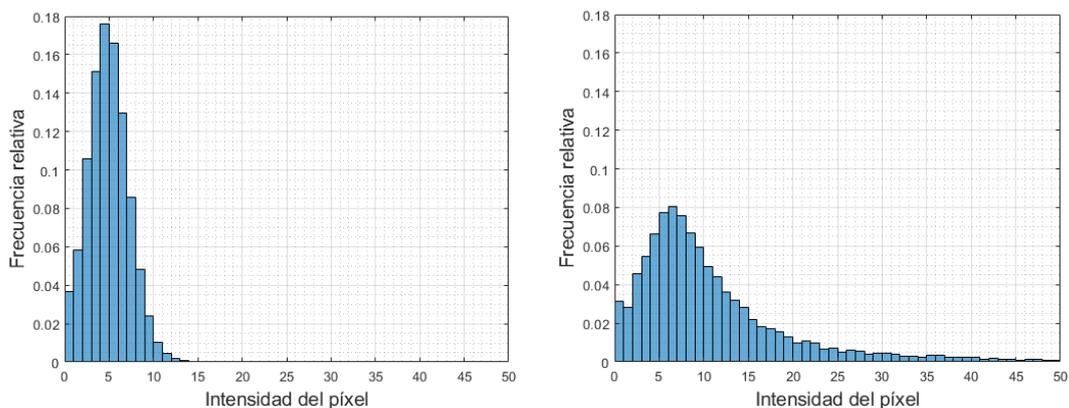


Figura 1 – Comparación de la distribución de intensidades de los píxeles en zonas sin flujo de partículas (izquierda) y zonas con flujo de partículas (derecha), basada en el análisis de una serie de imágenes experimentales.

Una de las principales limitaciones de esta técnica es que las regiones correspondientes a superficies sólidas que reflejan la luz de las partículas y generan imágenes reflejadas (partículas fantasmas), son interpretadas por el algoritmo como regiones de flujo y no son enmascaradas. Una posible solución a este problema es

realizar un ajuste manual posterior sobre la máscara para cubrir las zonas con imágenes reflejadas.

II.3. ESTABLECIMIENTO DEL TAMAÑO DE VENTANAS DE INTERROGACIÓN

Las ventanas de interrogación son las áreas en las que se subdividen las imágenes para efectuar la estimación del desplazamiento en distintos puntos del plano de la imagen. La correlación cruzada basada en el algoritmo de transformada rápida de Fourier (sección II.6) demanda que el par de ventanas sean del mismo tamaño, con forma rectangular y dimensiones que sean potencias de 2. Sin embargo, el empleo de funciones de ventaneo y rellenado con ceros (sección II.6) permite definir ventanas con forma y tamaño efectivo arbitrarios (Gui & Merzkirch, 1998; Gui, Merzkirch, & Fei, 2000; Eckstein & Vlachos, 2009).

Para la elección adecuada del tamaño efectivo de las ventanas se deben considerar una serie de factores. En primer lugar, la aplicación de la correlación cruzada seguida de un procedimiento de localización de máximo puede interpretarse como un operador que toma como entrada un par de ventanas de interrogación y devuelve una estimación del desplazamiento medio de las imágenes de partículas contenidas en el par. Para una densidad de partículas uniforme y relativamente alta (más de 10 partículas por ventana de interrogación), y diferencias en el desplazamiento de las partículas dentro de la ventana de interrogación menores al diámetro de estas, este operador puede ser considerado como un filtro pasa bajos espacial aplicado sobre el campo de desplazamientos (Willert & Gharib, 1991). Esto significa que, si existen fluctuaciones en el campo de desplazamientos con longitudes de onda menores al tamaño de las ventanas de interrogación, estas serán atenuadas. Por lo tanto, existe un error asociado con este efecto de filtrado espacial, el cual se ha analizado de forma teórica y a través de simulaciones numéricas por parte de Scarano (2003). Por otro lado, si las diferencias en el desplazamiento dentro de la ventana son mayores al diámetro medio de la imagen de las partículas, el emparejamiento de las partículas se pierde y la estimación a partir de la correlación cruzada produce un vector espurio.

A partir de lo mencionado precedentemente, podría concluirse que emplear ventanas de menor tamaño tendría un efecto beneficioso sobre la precisión y resolución de los resultados. Sin embargo, esto también trae como consecuencia una reducción de la cantidad de partículas dentro de las ventanas y del número de pares de partículas emparejadas entre el par de ventanas, resultando en un aumento en la incertidumbre (Willert & Gharib, 1991). Distintos autores han estudiado el efecto de la densidad de partículas sobre la precisión de los resultados (Raffel, y otros, 2018; Westerweel, 1994), y si bien no hay un consenso general en cuanto a la densidad de partículas óptima y esta depende de los algoritmos de interrogación empleados, se podría considerar que se encuentra comprendida entre 4 y 10 partículas por ventana de interrogación (Theunissen, Scarano, & Riethmuller, 2006). A pesar de que el cumplimiento de este criterio puede asegurarse sembrando más partículas, aún debe considerarse el efecto de pérdida de pares debido al desplazamiento fuera de la ventana. Como criterio para

esto se encuentra la ampliamente aplicada regla de un cuarto (Keane & Adrian, 1990), la cual establece que el desplazamiento medio de las imágenes de partículas dentro de la ventana no puede ser mayor a un cuarto del tamaño de las ventanas de interrogación.

Teniendo en cuenta los factores mencionados, es posible establecer un tamaño efectivo de ventana apropiado. Si bien la resolución espacial obtenida se encuentra limitada por este tamaño de ventanas, un posterior incremento de la resolución espacial puede alcanzarse mediante el empleo de las técnicas de pasos múltiples (sección II.11). Estas técnicas consisten en efectuar sucesivos pasos de procesamiento PIV, utilizando el campo de desplazamientos obtenido en un paso de procesamiento PIV previo como entrada en un paso de procesamiento posterior. En el método basado en el desplazamiento discreto de ventanas (sección II.11.1), las ventanas de interrogación se toman con su centro desplazado desde el punto de grilla, una distancia que es función del desplazamiento estimado en el paso PIV anterior, de forma de que el desplazamiento medio de las partículas entre el par de ventanas queda comprendido en el rango de $[-\frac{1}{2}, \frac{1}{2}]$ píxeles. Esto elimina la restricción impuesta por la regla de un cuarto, y permite emplear ventanas más chicas y una grilla más fina, logrando un aumento en la resolución espacial.

El otro método que también permite aumentar la resolución espacial es la técnica basada en deformación de imágenes (sección II.11.2). Esta aplica una deformación sobre el par de imágenes que están siendo procesadas, según el campo de desplazamientos obtenido en el paso previo, y efectúa el procesamiento PIV sobre el par de imágenes deformadas. Esto no solo elimina la restricción impuesta por la regla de un cuarto, sino que también reduce el gradiente de desplazamiento dentro de las ventanas, aumentando la relación señal-ruido del pico de correlación. También reduce el efecto de promediado espacial producido por el tamaño de las ventanas de interrogación.

Una estrategia para reducir la limitación introducida por la regla de un cuarto sobre el tamaño de las ventanas para el primer paso de una implementación de PIV de pasos múltiples, es emplear ventanas de distinto tamaño. Si se utiliza un tamaño más grande para la segunda ventana de interrogación, de tal forma de conformar un área de búsqueda que abarque las imágenes de partículas de la primera ventana, entonces se elimina el error introducido por el efecto de pérdida de pares, sin reducir la resolución espacial, puesto que esta depende del tamaño de la primera ventana. Esta técnica usualmente es referida en la bibliografía como seguimiento de partículas basada en correlación cruzada, y fue descrita y aplicada por Huang y otros (1993), Kemmerich & Rath (1994), y Fincham & Spedding (1997). El desempeño de esta técnica en la reducción de los errores se analiza en la sección V.4 mediante simulaciones numéricas.

Una alternativa más sofisticada a los métodos de pasos múltiples previamente mencionados incluye la utilización de una grilla irregular, en donde los puntos de la grilla, y el tamaño de ventanas a emplear en cada punto, se seleccionan de forma dinámica en función de los gradientes de velocidad y la densidad de partículas a nivel local (Theunissen, Scarano, & Riethmuller, 2006). Esta técnica fue desestimada en el presente

trabajo ya que su implementación es más compleja, y se observó que los métodos de pasos múltiples son suficientes para obtener resultados satisfactorios en los casos analizados.

II.4. EMPLEO DE FUNCIONES DE VENTANEO

En la práctica, para realizar la correlación cruzada (sección II.6), comúnmente se emplea la transformada de Fourier discreta (anexo IX.2), con el algoritmo de Transformada Rápida de Fourier o FFT (del inglés, Fast Fourier Transform). Esto es posible a partir del teorema de convolución (anexo IX.4), y permite un cálculo más rápido de esta función, en contraste con el método basado en correlación directa. No obstante, la función de correlación obtenida por medio de este método equivale a calcular la correlación directa considerando que las ventanas de interrogación son periódicas, con periodicidad dada por el tamaño de estas (anexo IX.3). Este efecto periódico produce una discrepancia entre el plano de correlación obtenido por correlación directa y el obtenido por el método basado en transformada de Fourier, contribuyendo a aumentar el error aleatorio en la estimación del desplazamiento (secciones V.1 y V.4).

Una técnica que permite eliminar el efecto periódico producido por el cálculo basado en la transformada de Fourier es el relleno con ceros (Raffel, y otros, 2018), que consiste en duplicar el tamaño de las ventanas de interrogación relleno estas perimetralmente con ceros. Esto desplaza la condición de periodicidad hacia afuera de los bordes del área de interrogación original, y la función de correlación obtenida de esta forma por el método de transformada de Fourier se vuelve equivalente a la obtenida por correlación directa.

Raffel y otros (2018) desaconsejan el uso del relleno con ceros bajo el argumento de que esto introduce discontinuidades entre los bordes de la ventana original y el área rellena con ceros, que se traduce en adición de errores por fuga espectral. No obstante, esta discontinuidad introducida en los bordes puede ser parcialmente atenuada mediante el empleo de funciones de ventaneo. Estas son funciones especialmente diseñadas para proveer una atenuación de los efectos de borde en las señales de tamaño finito, buscando la mínima alteración en el contenido espectral de las mismas. La extensiva investigación en el campo del análisis de Fourier ha dado lugar a una amplia variedad de funciones de ventaneo con diversas propiedades espectrales (Harris, 1978; Nuttall, 1981). En la bibliografía se encuentran diversas publicaciones que tratan sobre la aplicación de funciones de ventaneo en aplicaciones de PIV. Gui (2000) empleó una función de ventaneo Gaussiana para reducir los errores sistemáticos producidos por el efecto de pérdida de correlación. Nogueira (1999) demostró como una función de ventaneo Gaussiana puede ser utilizada para modificar el efecto de filtro pasa bajos espacial producido en la estimación PIV. Eckstein y Vlachos (2009) realizaron un estudio sobre la aplicación de distintas funciones de ventaneo, y reportaron que todas ellas producen un error aleatorio similar, pero inferior que el producido mediante el relleno con ceros sin ventaneo. En la sección V.4 se analizan los efectos del relleno

con ceros y el ventaneo sobre el error en la estimación, reportando resultados obtenidos por simulaciones de Montecarlo. A raíz de los resultados obtenidos, se consideran apropiadas ambas estrategias, ya que permiten reducir el error aleatorio en los resultados. Cabe destacar que aplicar el ventaneo y luego realizar la correlación cruzada produce un error sistemático que, sin embargo, puede ser reducido significativamente mediante una normalización, dividiendo la función de correlación por la correlación cruzada de las funciones de ventaneo, tal como se discute en la sección V.4.

II.5. ESTABLECIMIENTO DE LA GRILLA

La grilla determina los puntos en el plano de la imagen sobre los que se estimarán los vectores de desplazamiento. Es conveniente el empleo de grillas regulares ya que resulta más fácil su manipulación y el cálculo de cantidades derivadas es menos complejo. De esta forma, la grilla queda completamente definida por el espaciamiento de los puntos en las direcciones vertical y horizontal, y por la distancia máxima entre los bordes de la imagen y los puntos de la grilla.

El espaciamiento entre los puntos de grilla se establece en función del tamaño de ventanas y del factor de solapamiento, el cual determina el grado de superposición entre ventanas correspondientes a puntos de grilla vecinos (por ejemplo, utilizar ventanas de 32x32 píxeles con un solapamiento del 50% implica utilizar un espaciamiento de grilla de 16x16 píxeles). El solapamiento de ventanas adyacentes genera información compartida entre ellas, por lo que los errores en los vectores de desplazamiento estimados adyacentes guardarán cierto grado de correlación. En última instancia, la correlación entre los vectores de desplazamiento estimados aumenta la incertidumbre de cantidades derivadas como la vorticidad (Sciacchitano & Wieneke, 2016). Los factores de solapamiento utilizados comúnmente son del 50 y 75%. En el caso de emplear esquemas de pasos múltiples, se debe seleccionar un espaciamiento de grilla para cada paso.

II.6. CORRELACIÓN CRUZADA

La técnica PIV se basa en la estimación de la velocidad del fluido en el plano de la imagen, sobre los distintos puntos que conforman la grilla previamente definida (sección II.5), a partir del desplazamiento medio de grupos de partículas en áreas más pequeñas de la imagen llamadas ventanas de interrogación (sección II.3). Sobre cada punto de la grilla, se toma un par de ventanas de interrogación de la primera y segunda imagen a ser procesadas, centradas en dicho punto (o bien, en los métodos de pasos múltiples con desplazamiento discreto, desplazadas desde el punto de grilla según el campo de desplazamientos estimado en un paso previo, sección II.11). Luego, mediante una técnica de procesamiento, es posible utilizar el desplazamiento relativo del grupo de partículas entre ambas ventanas para estimar el desplazamiento local. La técnica de procesamiento más utilizada para este propósito se basa en la aplicación de la función de correlación cruzada. La evaluación de esta función consiste en superponer ambas ventanas, y luego, aplicando distintos desplazamientos de una ventana con respecto a

la otra, calcular la suma de los productos de las intensidades píxel a píxel. Para aquellos desplazamientos para los cuales existe un alto grado de coherencia entre el contenido de las ventanas, se produce un máximo o pico en el plano de correlación. El pico máximo del plano de correlación entonces puede ser utilizado como una estimación del desplazamiento.

Debido a que la función de correlación cruzada está evaluada para desplazamientos discretos, el desplazamiento estimado de esta forma tiene una precisión en el rango de $[-\frac{1}{2}, \frac{1}{2}]$ píxeles. No obstante, una precisión mayor puede alcanzarse haciendo uso de una técnica de estimación subpíxel, que consiste en emplear la información del plano de correlación en torno al pico máximo (sección II.7).

En la sección II.6.1 se da una definición formal de la función de correlación cruzada, mientras que en la sección II.6.2 se presenta una forma de calcular esta mediante el empleo de la transformada de Fourier.

Ciertos autores proponen algoritmos alternativos para estimar el desplazamiento más probable para las partículas en un par de ventanas. Por ejemplo, Gui y Merzkirch (1996a; 1996b) proponen un método basado en mínimos cuadrados. En la sección II.6.3 se describe este método y se demuestra como la función de error cuadrático se puede calcular por medio de la transformada de Fourier. Por otro lado, Ronneberger y otros (1998) proponen calcular la función de correlación de forma normalizada con la media y el desvío estándar de las ventanas. Este método produce un plano de correlación normalizado, con valores en el rango $[-1, 1]$. En la sección II.6.4 se describe este método y se muestra como puede ser calculada la correlación normalizada utilizando la transformada de Fourier. En la sección V.5 se compara el desempeño de los distintos métodos de correlación mencionados, reportando resultados obtenidos por medio del procesamiento de imágenes sintéticas.

II.6.1. Correlación cruzada directa

Sean las ventanas de interrogación g_1 y g_2 , representadas por medio de dos matrices de tamaños $Ni_1 \times Nj_1$ y $Ni_2 \times Nj_2$, respectivamente. Los valores de intensidad para cada píxel de la ventana g_1 vienen dados por $g_1(i, j)$, con $i = 1, 2, \dots, Ni_1$ y $j = 1, 2, \dots, Nj_1$. Análogamente, los valores de intensidad para cada píxel de la ventana g_2 vienen dados por $g_2(i, j)$, con $i = 1, 2, \dots, Ni_2$ y $j = 1, 2, \dots, Nj_2$. Se definen las ventanas en el dominio extendido g_1^* y g_2^* como dos funciones $g_1^*: \mathbb{Z}^2 \rightarrow \mathbb{R}$, $g_2^*: \mathbb{Z}^2 \rightarrow \mathbb{R}$, tales que

$$g_1^*(i, j) = \begin{cases} g_1(i, j) & \text{si } i \in \{1, 2, \dots, Ni_1\} \text{ y } j \in \{1, 2, \dots, Nj_1\} \\ 0 & \text{en caso contrario} \end{cases}, \text{ con } i, j \in \mathbb{Z},$$

$$g_2^*(i, j) = \begin{cases} g_2(i, j) & \text{si } i \in \{1, 2, \dots, Ni_2\} \text{ y } j \in \{1, 2, \dots, Nj_2\} \\ 0 & \text{en caso contrario} \end{cases}, \text{ con } i, j \in \mathbb{Z}.$$

Finalmente, se define la función de correlación cruzada, $C: \mathbb{Z}^2 \rightarrow \mathbb{R}$, como

$$C(m, n) = \sum_{i=1}^{Ni_1} \sum_{j=1}^{Nj_1} g_1^*(i, j) g_2^*(i + m, j + n)$$

Ec. II.6-1

$$= \sum_{i=1}^{Ni_2} \sum_{j=1}^{Nj_2} g_1^*(i - m, j - n) g_2^*(i, j),$$

donde m y n representan los desplazamientos discretos relativos entre el par de ventanas para los que se evalúa C .

Otra alternativa es definir las ventanas en el dominio extendido g_1^* y g_2^* de forma cíclica como

$$g_1^*(i + k Ni_1, j + l Nj_1) = g_1(i, j), \text{ con } i \in \{1, 2, \dots, Ni_1\}, j \in \{1, 2, \dots, Nj_1\} \text{ y } k, l \in \mathbb{Z},$$

$$g_2^*(i + k Ni_2, j + l Nj_2) = g_2(i, j), \text{ con } i \in \{1, 2, \dots, Ni_2\}, j \in \{1, 2, \dots, Nj_2\} \text{ y } k, l \in \mathbb{Z}.$$

La función de correlación obtenida con g_1^* y g_2^* definidas de esta forma se conoce como *correlación cíclica*.

A la función de correlación se le pueden agregar unas funciones de peso W_1 y W_2 , que sirvan para ponderar la contribución relativa de los términos en la suma. De esta forma, la función de correlación queda definida por

$$C(m, n) = \frac{\sum_{i=1}^{Ni_1} \sum_{j=1}^{Nj_1} W_1(i, j) W_2(i + m, j + n) g_1^*(i, j) g_2^*(i + m, j + n)}{\sum_{i=1}^{Ni_1} \sum_{j=1}^{Nj_1} W_1(i, j) W_2(i + n, j + m)}$$

Ec. II.6-2

$$= \frac{\sum_{i=1}^{Ni_1} \sum_{j=1}^{Nj_1} \tilde{g}_1^*(i, j) \tilde{g}_2^*(i + m, j + n)}{S(m, n)} = \frac{\tilde{C}(m, n)}{S(m, n)},$$

donde \tilde{C} es la correlación cruzada entre \tilde{g}_1^* y \tilde{g}_2^* , siendo $\tilde{g}_1^* = g_1^* W_1$ y $\tilde{g}_2^* = g_2^* W_2$, respectivamente, y S es la correlación cruzada entre W_1 y W_2 . Las funciones de peso a su vez se pueden representar como

$$W_1(i, j) = \begin{cases} \varphi_1(i, j) \lambda_1(i, j) & \text{si } i \in \{1, 2, \dots, Ni_1\} \text{ y } j \in \{1, 2, \dots, Nj_1\}, \text{ con } i, j \in \mathbb{Z}, \\ 0 & \text{en caso contrario} \end{cases}$$

$$W_2(i, j) = \begin{cases} \varphi_2(i, j) \lambda_2(i, j) & \text{si } i \in \{1, 2, \dots, Ni_2\} \text{ y } j \in \{1, 2, \dots, Nj_2\}, \text{ con } i, j \in \mathbb{Z}, \\ 0 & \text{en caso contrario} \end{cases}$$

donde φ_1 y φ_2 representan funciones de ventaneo (sección II.4), mientras que λ_1 y λ_2 son máscaras binarias con valores posibles 0 o 1, que sirven para enmascarar píxeles que se desea que no tengan influencia en la correlación. La aplicación de máscaras binarias en el cálculo de la correlación cruzada es una técnica conocida en la bibliografía

como técnica de máscara digital (Gui, Merzkirch, & Fei, 2000; Gui, Wereley, & Kim, 2003).

En principio S en la ecuación 2 es función de n y m , y su determinación implica llevar a cabo una correlación cruzada adicional por cada par de ventanas constituyendo un doble costo computacional. No obstante, si las funciones de peso a emplear son las mismas para todos los casos en una aplicación determinada, el cálculo de S se debe llevar a cabo solo una vez, economizando el procedimiento. Alternativamente, se puede calcular el desplazamiento discreto a partir del máximo de la función de correlación \tilde{C} , y luego calcular los valores de S únicamente en las posiciones requeridas para aplicar el algoritmo de estimación subpíxel, significando un ahorro en costo computacional. En los casos de que alguna o ambas ventanas W_1 y W_2 sean de valor constante, la función S será constante, por lo que su aplicación resultará irrelevante a fines de la estimación del desplazamiento, permitiendo omitirla en la ecuación previa, y calcular el desplazamiento directamente a partir de \tilde{C} .

La cantidad de operaciones requeridas para calcular la función de correlación $C(n, m)$ para cada desplazamiento es del orden de $\min([Ni_1, Ni_2]) \min([Nj_1, Nj_2])$. Si se desea evaluar la función de correlación para cada desplazamiento posible se requieren del orden de $Ni_1 Ni_2 Nj_1 Nj_2$ operaciones.

II.6.2. Correlación cruzada estándar

Este esquema de correlación consiste en calcular la correlación cruzada entre dos ventanas utilizando la transformada rápida de Fourier y el teorema de convolución (anexo IX.4). Sean como antes dos ventanas g_1 y g_2 representadas por matrices de tamaños $Ni_1 \times Nj_1$ y $Ni_2 \times Nj_2$, respectivamente. Sea la función $O: \mathbb{N} \rightarrow \mathbb{N}$

$$O(N) = k, \text{ tal que } 2^{k-1} < N \leq 2^k, \text{ y } k \in \mathbb{N}.$$

El mínimo tamaño de ventana requerido para calcular la correlación cruzada de las ventanas g_1 y g_2 aplicando el método de FFT puede determinarse como

$$S_i = 2^{O(\text{Max}[Ni_1, Ni_2])},$$

$$S_j = 2^{O(\text{Max}[Nj_1, Nj_2])}.$$

Luego, se definen las ventanas g'_1 y g'_2 , ambas matrices de tamaño $S_i \times S_j$, obtenidas agregando perimetralmente ceros a las ventanas g_1 y g_2 para llevarlas a las dimensiones requeridas. En adelante, se denominan ventanas efectivas a las ventanas g_1 y g_2 , y ventanas de correlación a g'_1 y g'_2 . Finalmente, la función de correlación puede calcularse por el método de transformada rápida como

$$C = \mathcal{F}^{-1} \{ \overline{\mathcal{F}\{g'_1\}} \mathcal{F}\{g'_2\} \},$$

Ec. II.6-3

donde \mathcal{F} denota la transformada de Fourier, $\overline{\mathcal{F}}$ denota su complejo conjugado, y \mathcal{F}^{-1} denota la transformada de Fourier inversa. La función de correlación calculada de esta forma es equivalente a calcular la función de correlación cíclica directa entre las ventanas g'_1 y g'_2 . Si en su lugar, las ventanas se rellenan con ceros tomando como tamaño de las ventanas de correlación el doble del usado previamente, es decir, con

$$S_i = 2^{O(\text{Max}[Ni_1, Ni_2]) + 1},$$

$$S_j = 2^{O(\text{Max}[Nj_1, Nj_2]) + 1},$$

entonces la función de correlación obtenida mediante la Ec. II.6-3 es equivalente a la obtenida por correlación directa entre las ventanas g_1 y g_2 (Ec. II.6-1).

Para efectuar la correlación cruzada por este método se requieren en total 3 aplicaciones de la transformada de Fourier, cada una requiriendo del orden de $S_i S_j (S_i + S_j)$ operaciones utilizando la definición de transformada de Fourier, o bien del orden de $S_i S_j \log S_i S_j$ operaciones con el algoritmo de la transformada rápida de Fourier.

Si se adicionan funciones de peso W_1 y W_2 , la función de correlación puede calcularse utilizando la transformada de Fourier como

$$C = \frac{\mathcal{F}^{-1}\{\overline{\mathcal{F}\{W_1 g'_1\}} \mathcal{F}\{W_2 g'_2\}\}}{\mathcal{F}^{-1}\{\overline{\mathcal{F}\{W_1\}} \mathcal{F}\{W_2\}\}} \quad \text{Ec. II.6-4}$$

II.6.3. Mínimos cuadrados

Otra forma de determinar el desplazamiento más probable entre las partículas contenidas en un par de ventanas es mediante el concepto de mínimos cuadrados. La utilización de mínimos cuadrados para la estimación del desplazamiento en PIV fue propuesta por Gui y Merzkirch (1996a; 1996b). Sean como antes dos ventanas g_1 y g_2 representadas por matrices de tamaños $Ni_1 \times Nj_1$ y $Ni_2 \times Nj_2$, respectivamente, y las respectivas ventanas en el dominio extendido g_1^* y g_2^* . Sean las funciones de peso W_1 y W_2 . Se busca minimizar la diferencia entre las dos ventanas para un desplazamiento dado (m, n) a través de la función de error cuadrático $E: \mathbb{Z}^2 \rightarrow \mathbb{R}$ tal que:

$$E(m, n) = \frac{\sum_{i=1}^{Ni_1} \sum_{j=1}^{Nj_1} W_1(i, j) W_2(i + m, j + n) [g_1^*(i, j) - g_2^*(i + m, j + n)]^2}{\sum_{i=1}^{Ni_1} \sum_{j=1}^{Nj_1} W_1(i, j) W_2(i + m, j + n)}$$

Expandiendo esta función, se hace posible expresar el error cuadrático en función de tres términos:

$$E(m, n) = \frac{MS_1(m, n)}{S(m, n)} + \frac{MS_2(m, n)}{S(m, n)} - 2 \frac{C(m, n)}{S(m, n)},$$

donde

$$C(m, n) = \frac{\sum_{i=1}^{N_{i_1}} \sum_{j=1}^{N_{j_1}} W_1(i, j) W_2(i + m, j + n) g_1^*(i, j) g_2^*(i + m, j + n)}{S(m, n)},$$

$$MS_1(m, n) = \frac{\sum_{i=1}^{N_{i_1}} \sum_{j=1}^{N_{j_1}} W_1(i, j) W_2(i + m, j + n) [g_1^*(i, j)]^2}{S(m, n)},$$

$$MS_2(m, n) = \frac{\sum_{i=1}^{N_{i_1}} \sum_{j=1}^{N_{j_1}} W_1(i, j) W_2(i + m, j + n) [g_2^*(i + m, j + n)]^2}{S(m, n)},$$

$$S(m, n) = \sum_{i=1}^{N_{i_1}} \sum_{j=1}^{N_{j_1}} W_1(i, j) W_2(i + m, j + n).$$

Si, como se hizo en la sección II.6.2, a las ventanas g_1 y g_2 se les aplica rellenado de ceros para convertirlas en las ventanas de igual tamaño g_1' y g_2' , es posible calcular más rápidamente estas cantidades mediante la transformada rápida de Fourier como

$$C = \frac{\mathcal{F}^{-1}\{\overline{\mathcal{F}\{W_1 g_1'\}} \mathcal{F}\{W_2 g_2'\}\}}{S},$$

$$MS_1 = \frac{\mathcal{F}^{-1}\{\overline{\mathcal{F}\{W_1 g_1'^2\}} \mathcal{F}\{W_2\}\}}{S},$$

$$MS_2 = \frac{\mathcal{F}^{-1}\{\overline{\mathcal{F}\{W_1\}} \mathcal{F}\{W_2 g_2'^2\}\}}{S},$$

$$S = \mathcal{F}^{-1}\{\overline{\mathcal{F}\{W_1\}} \mathcal{F}\{W_2\}\}.$$

Se observa que en el caso más general se requieren 10 operaciones de transformada rápida de Fourier. Con un costo computacional más elevado, este método no suele implementarse en la práctica para la estimación de desplazamientos.

II.6.4. Correlación cruzada normalizada

La aplicación del método de correlación cruzada normalizada a PIV fue propuesto por Ronneberger y otros (1998). Sean como antes dos ventanas g_1 y g_2 representadas por matrices de tamaños $N_{i_1} \times N_{j_1}$ y $N_{i_2} \times N_{j_2}$, respectivamente, y sus respectivas ventanas en el dominio ampliado g_1^* y g_2^* . Sean las funciones de peso W_1 y W_2 . Se define la correlación cruzada normalizada como la función $\rho: \mathbb{Z}^2 \rightarrow \mathbb{R}$ tal que

$$\rho(m, n) = \frac{\sigma_{1,2}(m, n)}{\sigma_1(m, n) \sigma_2(m, n)},$$

donde,

$$\sigma_{1,2}(m, n) = C(m, n) - \mu_1(m, n) \mu_2(m, n),$$

$$\sigma_1(m, n) = \sqrt{MS_1(m, n) - \mu_1(m, n)^2},$$

$$\sigma_2(m, n) = \sqrt{MS_2(m, n) - \mu_2(m, n)^2},$$

$$C(m, n) = \frac{\sum_{i=1}^{N_{i_1}} \sum_{j=1}^{N_{j_1}} W_1(i, j) W_2(i + m, j + n) g_1^*(i, j) g_2^*(i + m, j + n)}{S(m, n)},$$

$$\mu_1(m, n) = \frac{\sum_{i=1}^{N_{i_1}} \sum_{j=1}^{N_{j_1}} W_1(i, j) W_2(i + m, j + n) g_1^*(i, j)}{S(m, n)},$$

$$\mu_2(m, n) = \frac{\sum_{i=1}^{N_{i_1}} \sum_{j=1}^{N_{j_1}} W_1(i, j) W_2(i + m, j + n) g_2^*(i + m, j + n)}{S(m, n)},$$

$$MS_1(m, n) = \frac{\sum_{i=1}^{N_{i_1}} \sum_{j=1}^{N_{j_1}} W_1(i, j) W_2(i + m, j + n) [g_1^*(i, j)]^2}{S(m, n)},$$

$$MS_2(m, n) = \frac{\sum_{i=1}^{N_{i_1}} \sum_{j=1}^{N_{j_1}} W_1(i, j) W_2(i + m, j + n) [g_2^*(i + m, j + n)]^2}{S(m, n)},$$

$$S(m, n) = \sum_{i=1}^{N_{i_1}} \sum_{j=1}^{N_{j_1}} W_1(i, j) W_2(i + m, j + n).$$

La correlación cruzada normalizada es más costosa computacionalmente, pero tiene la ventaja de que sus posibles valores se encuentran en el rango $[-1, 1]$, donde valores cercanos a 1 indican un elevado grado de correlación para un desplazamiento entre ventanas dado. Esto es una medida directa de que tan bueno es el ajuste de los patrones para distintos desplazamientos. Si a las ventanas g_1 y g_2 se las rellena con ceros para convertirlas en las ventanas de igual tamaño g'_1 y g'_2 , es posible calcular más rápidamente estas cantidades mediante la transformada rápida de Fourier como

$$C = \frac{\mathcal{F}^{-1}\{\overline{\mathcal{F}\{W_1 g'_1\}} \mathcal{F}\{W_2 g'_2\}\}}{S},$$

$$\mu_1 = \frac{\mathcal{F}^{-1}\{\overline{\mathcal{F}\{W_1 g'_1\}} \mathcal{F}\{W_2\}\}}{S},$$

$$\mu_2 = \frac{\mathcal{F}^{-1}\{\overline{\mathcal{F}\{W_1\}} \mathcal{F}\{W_2 g'_2\}\}}{S},$$

$$MS_1 = \frac{\mathcal{F}^{-1}\{\overline{\mathcal{F}\{W_1 g'^2_1\}} \mathcal{F}\{W_2\}\}}{S},$$

$$MS_2 = \frac{\mathcal{F}^{-1}\{\overline{\mathcal{F}\{W_1\}} \mathcal{F}\{W_2 g'^2_2\}\}}{S},$$

$$S = \mathcal{F}^{-1}\{\overline{\mathcal{F}\{W_1\}} \mathcal{F}\{W_2\}\}.$$

Se observa que, en el caso más general, se requieren efectuar un total de 12 operaciones de transformada rápida de Fourier.

II.7. ESTIMACIÓN SUBPÍXEL

La estimación del desplazamiento a partir del máximo del plano de correlación, que está calculado para desplazamientos discretos, tiene una precisión de $\pm 1/2$ píxel. No obstante, es posible emplear la información de los valores de la función de correlación en un entorno del pico máximo para obtener una estimación con precisión subpíxel. En las secciones siguientes se describen distintos algoritmos empleados para este fin. En la sección V.8 se compara el desempeño de los distintos métodos de estimación subpíxel, a través de resultados obtenidos del procesamiento de imágenes sintéticas.

II.7.1. Determinación del centroide

El cálculo del centroide o centro de masas de la función de correlación en torno al pico de máxima correlación puede emplearse para estimar la ubicación del pico con precisión subpíxel (Fisher & Naidu, 1996; Raffel, y otros, 2018). El centroide se define como el cociente entre el momento de primer orden y el momento de segundo orden de una señal. Sea C la función de correlación, calculada para desplazamientos discretos, y sea (i, j) el desplazamiento en el que ocurre el mayor valor en C . Sean δ_x y δ_y las componentes reales de desplazamiento en las direcciones x e y , respectivamente. Estas pueden estimarse por medio de los estimadores $\check{\delta}_x$ y $\check{\delta}_y$ dados por

$$\check{\delta}_x = i + \frac{\sum_{n=-k}^k n C(i+n, j)}{\sum_{n=-k}^k C(i+n, j)},$$

$$\check{\delta}_y = j + \frac{\sum_{n=-k}^k n C(i, j+n)}{\sum_{n=-k}^k C(i, j+n)},$$

para algún $k \in \mathbb{N}$. Por ejemplo, con $k = 1$ se tiene el estimador de tres puntos

$$\check{\delta}_x = i + \frac{C(i+1, j) - C(i-1, j)}{C(i+1, j) + C(i, j) + C(i-1, j)},$$

$$\check{\delta}_y = j + \frac{C(i, j+1) - C(i, j-1)}{C(i, j+1) + C(i, j) + C(i, j-1)}.$$

II.7.2. Interpolación lineal

Este método consiste en aproximar los valores que adopta la función de correlación entre los desplazamientos discretos por medio de una interpolación lineal (Fisher & Naidu, 1996). De esta forma, el desplazamiento en x con precisión subpíxel puede estimarse por medio del estimador

$$\check{\delta}_x = \begin{cases} i + \frac{1}{2} \frac{C(i+1, j) - C(i-1, j)}{C(i, j) - C(i-1, j)} & \text{si } C(i+1, j) \geq C(i-1, j) \\ i + \frac{1}{2} \frac{C(i+1, j) - C(i-1, j)}{C(i, j) - C(i+1, j)} & \text{en caso contrario} \end{cases}.$$

El estimador para δ_y se define de forma análoga.

II.7.3. Ajuste parabólico de tres puntos

El método de ajuste parabólico de tres puntos consiste en estimar δ_x y δ_y por separado, mediante el ajuste de los puntos $C(i-1, j)$, $C(i, j)$ y $C(i+1, j)$, y los puntos $C(i, j-1)$, $C(i, j)$ y $C(i, j+1)$ a dos curvas parabólicas. Se demuestra para el desplazamiento en x . se plantea la ecuación general de una curva cuadrática,

$$P(x) = ax^2 + bx + c.$$

Se busca que esta curva coincida con el plano de correlación en los puntos $(i-1, j)$, (i, j) y $(i+1, j)$, por lo tanto, se llega al sistema de ecuaciones

$$\begin{cases} a(i-1)^2 + b(i-1) + c = C(i-1, j) \\ a i^2 + b i + c = C(i, j) \\ a(i+1)^2 + b(i+1) + c = C(i+1, j) \end{cases}.$$

Al resolver este sistema para a , b y c , se obtiene

$$\begin{cases} a = \frac{C(i-1, j) - 2C(i, j) + C(i+1, j)}{2} \\ b = -2i a + \frac{C(i+1, j) - C(i-1, j)}{2} \\ c = C(i, j) - i^2 a - i b \end{cases}.$$

Por último, se define el estimador $\check{\delta}_x$ como el máximo del polinomio de ajuste $P(x)$, el cual por condición de máximo cumple

$$P'(\check{\delta}_x) = 2a\check{\delta}_x + b = 0 \rightarrow \check{\delta}_x = \frac{-b}{2a},$$

y aplicando las expresiones para a y b obtenidas por el ajuste, resulta

$$\check{\delta}_x = i + \frac{1}{2} \left(\frac{C(i-1, j) - C(i+1, j)}{C(i-1, j) - 2C(i, j) + C(i+1, j)} \right).$$

Análogamente para y ,

$$\check{\delta}_y = j + \frac{1}{2} \left(\frac{C(i, j-1) - C(i, j+1)}{C(i, j-1) - 2C(i, j) + C(i, j+1)} \right).$$

II.7.4. Ajuste gaussiano de tres puntos

El método de ajuste gaussiano de tres puntos (Fisher & Naidu, 1996; Brady M. R., 2006; Brady M. R., 2009; Raffel, y otros, 2018) consiste en estimar δ_x y δ_y por separado, mediante el ajuste de los puntos $C(i-1, j)$, $C(i, j)$ y $C(i+1, j)$, y los puntos $C(i, j-1)$, $C(i, j)$ y $C(i, j+1)$ a dos curvas gaussianas. Se demuestra para el desplazamiento en x . Se plantea la ecuación general de una función gaussiana,

$$f(x) = \check{I}_x e^{-\frac{1}{2\check{\sigma}_x^2}(x-\check{\delta}_x)^2}.$$

Si a esta función se le aplica el logaritmo natural, se obtiene la expresión

$$\text{Ln } f(x) = ax^2 + bx + c,$$

donde

$$\left\{ \begin{array}{l} a = -\frac{1}{2\check{\sigma}_x^2} \\ b = \frac{\check{\delta}_x}{\check{\sigma}_x^2} \\ c = \text{Ln } \check{I} - \frac{\check{\delta}_x^2}{2\check{\sigma}_x^2} \end{array} \right. .$$

Esto equivale a hacer el ajuste a una función cuadrática (tal como se mostró en la sección II.7.3) de los puntos $\text{Ln } C(i-1, j)$, $\text{Ln } C(i, j)$ y $\text{Ln } C(i+1, j)$. Siguiendo los resultados de la sección II.7.3, el estimador de desplazamiento en x con precisión subpíxel que surge de ajustar los tres puntos a una función gaussiana, viene dado por

$$\check{\delta}_x = i + \frac{1}{2} \left(\frac{\text{Ln}[C(i-1, j)] - \text{Ln}[C(i+1, j)]}{\text{Ln}[C(i-1, j)] - 2\text{Ln}[C(i, j)] + \text{Ln}[C(i+1, j)]} \right).$$

A partir de la expresión para a , el factor de escala σ_x del pico de máxima correlación puede estimarse como

$$\check{\sigma}_x = \sqrt{\frac{1}{2\text{Ln}[C(i, j)] - \text{Ln}[C(i-1, j)] - \text{Ln}[C(i+1, j)]}}.$$

Mientras que el máximo de la función puede estimarse como

$$\check{I}_x = e^{\frac{(i-\check{\delta}_x)^2}{2\check{\sigma}_x^2}}.$$

Los estimadores en la dirección y se determinan de forma análoga.

II.7.5. Ajuste gaussiano de cuatro puntos

El método de ajuste gaussiano de cuatro puntos (Brady M. R., 2006; 2009), consiste en ajustar 4 puntos del entorno del pico máximo, a la superficie gaussiana de cuatro parámetros dada por

$$f(x, y) = \check{I} e^{-\frac{1}{2\check{\sigma}^2}[(x-\check{\delta}_x)^2+(y-\check{\delta}_y)^2]}.$$

Sean (x_1, y_1) , (x_2, y_2) , (x_3, y_3) y (x_4, y_4) cuatro puntos del plano de correlación, y sean $a_1 = C(x_1, y_1)$, $a_2 = C(x_2, y_2)$, $a_3 = C(x_3, y_3)$ y $a_4 = C(x_4, y_4)$ los correspondientes valores de correlación. Los parámetros que ajustan la función f a estos cuatro puntos quedan dados por

$$\check{\delta}_x = \frac{\alpha_1 \ln a_1 + \alpha_2 \ln a_2 + \alpha_3 \ln a_3 + \alpha_4 \ln a_4}{2(\beta_1 \ln a_1 + \beta_2 \ln a_2 + \beta_3 \ln a_3 + \beta_4 \ln a_4)},$$

$$\check{\delta}_y = \frac{\gamma_1 \ln a_1 + \gamma_2 \ln a_2 + \gamma_3 \ln a_3 + \gamma_4 \ln a_4}{2(\beta_1 \ln a_1 + \beta_2 \ln a_2 + \beta_3 \ln a_3 + \beta_4 \ln a_4)},$$

$$\check{\sigma}^2 = \frac{(x_4 - \check{\delta}_x)^2 + (y_4 - \check{\delta}_y)^2 - (x_3 - \check{\delta}_x)^2 - (y_3 - \check{\delta}_y)^2}{2(\ln a_3 - \ln a_4)},$$

$$\check{I} = a_1 e^{\frac{1}{2\check{\sigma}^2}[(x_1 - \check{\delta}_x)^2 + (y_1 - \check{\delta}_y)^2]}$$

donde

$$\alpha_1 = x_4^2(y_2 - y_3) + x_3^2(y_4 - y_2) + [x_2^2 + (y_2 - y_3)(y_2 - y_4)](y_3 - y_4),$$

$$\alpha_2 = x_4^2(y_3 - y_1) + x_3^2(y_1 - y_4) - [x_1^2 + (y_1 - y_3)(y_1 - y_4)](y_3 - y_4),$$

$$\alpha_3 = x_4^2(y_1 - y_2) + x_2^2(y_4 - y_1) + [x_1^2 + (y_1 - y_2)(y_1 - y_4)](y_2 - y_4),$$

$$\alpha_4 = x_3^2(y_2 - y_1) + x_2^2(y_1 - y_3) - [x_1^2 + (y_1 - y_2)(y_1 - y_3)](y_2 - y_3),$$

$$\gamma_1 = -x_3^2 x_4 + x_2^2(x_4 - x_3) + x_4(y_2^2 - y_3^2) + x_2(x_3^2 - x_4^2 + y_3^2 - y_4^2) + x_3(x_4^2 - y_2^2 + y_4^2),$$

$$\gamma_2 = x_3^2 x_4 + x_1^2(x_3 - x_4) + x_4(y_3^2 - y_1^2) - x_3(x_4^2 - y_1^2 + y_4^2) + x_1(-x_3^2 + x_4^2 - y_3^2 + y_4^2),$$

$$\gamma_3 = -x_2^2 x_4 + x_1^2(x_4 - x_2) + x_4(y_1^2 - y_2^2) + x_2(x_4^2 - y_1^2 + y_4^2) + x_1(x_2^2 - x_4^2 + y_2^2 - y_4^2),$$

$$\gamma_4 = x_2^2 x_3 + x_1^2 (x_2 - x_3) + x_3 (y_2^2 - y_1^2) - x_2 (x_3^2 - y_1^2 + y_3^2) + x_1 (-x_2^2 + x_3^2 - y_2^2 + y_3^2).$$

$$\beta_1 = x_4 (y_2 - y_3) + x_2 (y_3 - y_4) + x_3 (y_4 - y_2),$$

$$\beta_2 = x_4 (y_3 - y_1) + x_3 (y_1 - y_4) + x_1 (y_4 - y_3),$$

$$\beta_3 = x_4 (y_1 - y_2) + x_1 (y_2 - y_4) + x_2 (y_4 - y_1),$$

$$\beta_4 = x_3 (y_2 - y_1) + x_2 (y_1 - y_3) + x_1 (y_3 - y_2).$$

II.7.6. Estimadores basados en el desarrollo en series de Taylor

Un estimador para la posición del pico con precisión subpíxel puede construirse a partir del desarrollo en series de Taylor de la señal cerca del pico (Fisher & Naidu, 1996). Si la derivada de una función f se desarrolla en series en torno a un punto x , y f presenta un máximo en $\delta_x = x + \Delta x$, entonces debe tenerse

$$f'(x + \Delta x) = 0 = f'(x) + \Delta x f''(x) + O(\Delta x^2).$$

Donde $O(\Delta x^2)$ contiene términos de orden 2 y superiores en Δx . Despreciando estos términos de orden mayor, se llega al estimador

$$\check{\delta}_x = x - \frac{f'(x)}{f''(x)}.$$

Las derivadas en esta expresión pueden aproximarse por medio de algún esquema de derivación numérica (Moin, 2010). Por ejemplo, usando esquemas de diferencias finitas de segundo orden, se llega a la expresión

$$\check{\delta}_x = x + \frac{1}{2} \left(\frac{f(x-1) - f(x+1)}{f(x-1) - 2f(x) + f(x+1)} \right),$$

que es equivalente al ajuste de tres puntos por medio de una parábola (sección II.7.3). Si se emplea un desarrollo de Taylor con más términos, se puede plantear la expresión

$$f'(x + \Delta x) = 0 = f'(x) + \Delta x f''(x) + \frac{\Delta x^2}{2} f'''(x) + \frac{\Delta x^3}{6} f^{(4)}(x) + O(\Delta x^4),$$

que, despreciando los términos de orden mayor, equivale a

$$f'(x) + \Delta x f''(x) + \frac{\Delta x^2}{2} f'''(x) + \frac{\Delta x^3}{6} f^{(4)}(x) = 0.$$

Aplicando esquemas de diferencias finitas de orden 4 para aproximar las derivadas, y resolviendo esta ecuación para Δx , se obtiene una estimación para δ_x .

11.7.7. Ajuste a superficie parabólica

Una forma robusta para estimar δ_x y δ_y es ajustar el pico de máxima correlación y sus puntos adyacentes a alguna superficie en dos dimensiones. En esta sección se plantea el ajuste a una superficie parabólica, que puede representarse de forma general por medio de la función

$$f(x, y; \boldsymbol{\theta}) = a_x x^2 + b_x x + a_y y^2 + b_y y + c,$$

para cierto arreglo de parámetros de ajuste $\boldsymbol{\theta} = [a_x, b_x, a_y, b_y, c]^T$. Sea C el plano de correlación determinado para desplazamientos discretos. Sean $\mathbf{x} = [x(1), x(1), \dots, x(N)]^T$ e $\mathbf{y} = [y(1), y(1), \dots, y(N)]^T$ las componentes horizontal y vertical de los puntos del plano de correlación que se desean ajustar, y sean

$$\mathbf{C} = [C(x(1), y(1)), C(x(2), y(2)), \dots, C(x(N), y(N))]^T,$$

$$\mathbf{f}(\boldsymbol{\theta}) = [f(x(1), y(1); \boldsymbol{\theta}), f(x(2), y(2); \boldsymbol{\theta}), \dots, f(x(N), y(N); \boldsymbol{\theta})]^T.$$

El ajuste por mínimos cuadrados consiste en determinar los parámetros de ajuste de la función f , de forma tal de minimizar la función de error cuadrático E dada por

$$E(\boldsymbol{\theta}) = \sum_{n=1}^N W(n) [C(x(n), y(n)) - f(x(n), y(n); \boldsymbol{\theta})]^2,$$

donde W es una función de peso con valores positivos que sirve para ponderar la contribución de los distintos términos en esta suma. Si se hace $\mathbf{H}_n = [x(n)^2, x(n), y(n)^2, y(n), 1]$, entonces el valor de la función f evaluada en el punto $(x(n), y(n))$ bajo los parámetros dados por $\boldsymbol{\theta}$ viene dada por

$$f(x(n), y(n); \boldsymbol{\theta}) = \mathbf{H}_n \boldsymbol{\theta}.$$

Si ahora se define la matriz de datos \mathbf{H} como

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_1 \\ \mathbf{H}_2 \\ \vdots \\ \mathbf{H}_N \end{bmatrix} = \begin{bmatrix} x(1)^2 & x(1) & y(1)^2 & y(1) & 1 \\ x(2)^2 & x(2) & y(2)^2 & y(2) & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x(N)^2 & x(N) & y(N)^2 & y(N) & 1 \end{bmatrix},$$

entonces puede escribirse

$$\mathbf{f}(\boldsymbol{\theta}) = \mathbf{H}\boldsymbol{\theta}.$$

Finalmente, la función de error cuadrático puede escribirse de forma matricial como

$$E(\boldsymbol{\theta}) = (\mathbf{C} - \mathbf{H}\boldsymbol{\theta})^T \mathbf{W} (\mathbf{C} - \mathbf{H}\boldsymbol{\theta}),$$

donde \mathbf{W} es la matriz diagonal de dimensión $N \times N$, cuyos elementos de la diagonal son $W(1), W(2), \dots, W(N)$. Este es un problema de mínimos cuadrados lineal que admite solución cerrada. El arreglo de parámetros $\boldsymbol{\theta}$ que minimiza E puede determinarse como (Kay, 1993)

$$\boldsymbol{\theta} = (\mathbf{H}^T \mathbf{W} \mathbf{H})^{-1} \mathbf{H}^T \mathbf{W} \mathbf{C}.$$

Una vez obtenido $\boldsymbol{\theta}$, el desplazamiento con precisión subpíxel puede estimarse como

$$\delta_x \approx -\frac{b_x}{2a_x},$$

$$\delta_y \approx -\frac{b_y}{2a_y}.$$

II.7.8. Ajuste a superficie gaussiana por mínimos cuadrados no lineales

Este método implica ajustar los puntos en el entorno del pico de máxima correlación por medio de una superficie gaussiana, de la forma

$$f(x, y) = I e^{-\frac{1}{2\sigma_x^2}(x-\delta_x)^2 - \frac{1}{2\sigma_y^2}(y-\delta_y)^2},$$

mediante un enfoque de mínimos cuadrados. Debido a que la función f es no lineal en los parámetros a ajustar, este es un problema de mínimos cuadrados no lineales, por lo que requiere de un procedimiento iterativo para aproximar el valor de los parámetros que minimizan el error cuadrático entre la superficie y los puntos del plano de correlación. Un algoritmo apropiado para este propósito es el algoritmo de Levenberg-Marquardt (Levenberg, 1944; Marquardt, 1963). La adaptación de este algoritmo para este caso se desarrolla en detalle en el anexo IX.7.

II.7.9. Ajuste a superficie gaussiana linealizada por mínimos cuadrados

Este enfoque consiste en ajustar los puntos en el entorno del pico máximo de correlación a una superficie gaussiana, aplicando previamente una estrategia de linealización utilizando el logaritmo natural (Nobach, 2005; Guo, 2011; Al-Nahhal, Dobre, Basar, Moloney, & Ikki, 2019). Esto se conoce como algoritmo de Caruana (Caruana, Searle, Heller, & Shupack, 1986). Sea la superficie gaussiana definida por medio de la función

$$f(x, y) = I e^{-\frac{1}{2\sigma_x^2}(x-\delta_x)^2 - \frac{1}{2\sigma_y^2}(y-\delta_y)^2},$$

para un conjunto de parámetros $I, \sigma_x, \sigma_y, \delta_x$ y δ_y . Si a esta expresión se le aplica el logaritmo natural, se obtiene

$$\text{Ln}[f(x, y)] = -\frac{1}{2\sigma_x^2}x^2 + \frac{\delta_x}{\sigma_x^2}x - \frac{1}{2\sigma_y^2}y^2 + \frac{\delta_y}{\sigma_y^2}y + \text{Ln}[I] - \frac{\delta_x^2}{2\sigma_x^2} - \frac{\delta_y^2}{2\sigma_y^2},$$

que se puede escribir como

$$\text{Ln}[f(x, y)] = a_x x^2 + b_x x + a_y y^2 + b_y y + c,$$

con

$$\begin{cases} a_x = -\frac{1}{2\sigma_x^2} & a_y = -\frac{1}{2\sigma_y^2} \\ b_x = \frac{\delta_x}{\sigma_x^2} & b_y = \frac{\delta_y}{\sigma_y^2} \\ c = \text{Ln}[I] - \frac{\delta_x^2}{2\sigma_x^2} - \frac{\delta_y^2}{2\sigma_y^2} \end{cases} .$$

A continuación, se plantea el problema de mínimos cuadrados sobre la función de error definida como

$$E = \sum_{n=1}^N W(n) \left[\text{Ln}[C(x(n), y(n))] - \text{Ln}[f(x(n), y(n))] \right]^2.$$

Puesto a que $\text{Ln}[f(x, y)]$ tiene la forma de una superficie parabólica, la solución de este problema de minimización es equivalente a la obtenida en la sección II.7.7 (salvo en que en este caso se ajustan los valores de $\text{Ln}[C]$ en lugar de C). El valor de los parámetros $I, \sigma_x, \sigma_y, \delta_x$ y δ_y puede derivarse a partir de los parámetros a_x, b_x, a_y, b_y , y c mediante las relaciones

$$\begin{cases} I = e^{c - \frac{b_x^2}{4a_x} - \frac{b_y^2}{4a_y}} & \delta_x = -\frac{b_x}{2a_x} \\ \sigma_x = \sqrt{-\frac{1}{2a_x}} & \delta_y = -\frac{b_y}{2a_y} \\ \sigma_y = \sqrt{-\frac{1}{2a_y}} \end{cases} .$$

Este método tiene la ventaja de ofrecer una solución cerrada sin necesidad de aplicar iteraciones, en contraste con el enfoque de mínimos cuadrados no lineales para el ajuste a una superficie gaussiana (sección II.7.8), pero es más sensible al error. Esto se analiza en Guo (2011).

II.7.10. Estimación por medio del flujo óptico

El flujo óptico se define como el movimiento aparente de patrones de luminosidad en una imagen. La estimación del flujo óptico tiene aplicación en el campo de visión artificial (Horn & Schunck, 1981; Hildreth, 1984), y también se ha implementado en PIV para la estimación subpíxel (Sugii, Nishio, Okuno, & Okamoto, 2000).

Sean g_1 y g_2 dos ventanas de interrogación de tamaño $N_i \times N_j$. Sean g_1 y g_2 los vectores columna de longitud $N = N_i N_j$ que surgen de reagrupar las ventanas g_1 y g_2 . Sean D_x y D_y matrices de $N \times N$ que representan operadores de diferencias finitas para las derivadas en x y en y , respectivamente, y sea la matriz H de dimensiones $N \times 2$ dada por

$$H = \left[\frac{D_x g_1 + D_x g_2}{2}, \frac{D_y g_1 + D_y g_2}{2} \right].$$

Un estimador para los desplazamientos en x y en y viene dado por

$$[\check{\delta}_x, \check{\delta}_y]^T = -(H^T W H)^{-1} H^T W (g_2 - g_1).$$

La derivación de esta expresión puede encontrarse en el anexo, sección IX.8.

Este método solo es válido para desplazamientos pequeños. Para aplicarlo, es necesario primero estimar el desplazamiento con precisión de $\pm \frac{1}{2}$ píxel, y luego aplicar este esquema de estimación sobre las ventanas de interrogación desplazadas según el desplazamiento previamente medido.

II.7.11. Ajuste de mínimos cuadrados sobre las ventanas de interrogación

El ajuste de imágenes por el método de mínimos cuadrados es una técnica que encuentra aplicación en el campo del procesamiento de imágenes (Thevenaz, Ruttimann, & Unser, 1998), pero que no ha sido ampliamente utilizada en PIV, debido posiblemente al mayor costo computacional que presenta en contraste con los métodos de correlación cruzada basados en el algoritmo de transformada rápida de Fourier. No obstante, el empleo de ajuste de imágenes permite tener en cuenta movimientos más generales del patrón de imágenes de partículas (translación, rotación, deformaciones de corte), mientras que el método de correlación cruzada solo tiene en cuenta la translación.

Sean g_1 y g_2 un par de ventanas de interrogación, extraídas de un par de capturas separadas por un tiempo Δt . Para un tiempo Δt lo suficientemente pequeño, el contenido de la ventana g_2 puede relacionarse con el contenido de la ventana g_1 por medio de la expresión

$$g_2 \approx T(g_1; \theta),$$

para algún conjunto de parámetros θ , donde T es una transformación afín, que aplica operaciones de traslación, rotación y deformaciones de corte sobre el contenido de la ventana g_1 , en función de los valores de θ . Alternativamente puede escribirse

$$g_1 \approx T^{-1}(g_2; \theta),$$

donde T^{-1} denota la función inversa de T . El ajuste de las ventanas por el enfoque de mínimos cuadrados implica encontrar el valor de los parámetros θ tal que minimicen la función de error dada por

$$E(\theta) = \sum (g_1 - T^{-1}(g_2; \theta))^2,$$

donde la suma se lleva a cabo sobre todos los píxeles de la ventana de interrogación. Este es un problema de mínimos cuadrados no lineales que requiere un enfoque iterativo para ser resuelto. Thevenaz, Ruttimann, & Unser (1998) proponen resolver el problema mediante una modificación del algoritmo de Levenberg-Marquardt (Levenberg, 1944; Marquardt, 1963).

II.8. VALIDACIÓN

Una vez calculado el vector de desplazamiento en cada punto de la grilla, se obtiene el campo de desplazamientos estimado. Los vectores de desplazamiento estimados poseen un error asociado al error inherente de los métodos de correlación y de estimación subpíxel empleados. Además, debido a ciertos factores locales, entre los que se encuentran la densidad deficiente de partículas, movimientos fuera de plano, cambios en la iluminación entre cuadros, entre otros, es posible que aparezcan picos espurios en el plano de correlación. Si la intensidad de alguno de estos es lo suficientemente grande para alcanzar un valor máximo dentro del plano de correlación, el algoritmo de estimación lo seleccionará para estimar el desplazamiento. La magnitud y dirección de un vector de desplazamiento estimado a partir de un pico espurio tendrán un valor completamente aleatorio, por lo que este vector no guardará ningún tipo de correlación espacial o continuidad topológica con respecto a los vectores de desplazamiento vecinos. Estos vectores que resultan erróneos en el sentido de que no guardan dicho grado de coherencia con sus vectores vecinos, son denominados vectores espurios. Los vectores espurios introducen errores en etapas posteriores del procesamiento, por lo que deben ser identificados y reemplazados.

Un modelo estadístico sencillo para los vectores estimados fue planteado por Westerweel (1994). Sea $\check{V}(i, j)$ el vector aleatorio que representa la estimación para el vector de desplazamiento en el punto de grilla (i, j) . Sea $\bar{V}(i, j)$ el vector que representa el desplazamiento medio de las imágenes de partículas dentro de la ventana de interrogación en dicho punto de grilla, y sea $\varepsilon(i, j)$ un vector aleatorio que representa el ruido en la estimación. La ocurrencia de vectores espurios puede modelarse a través de un vector aleatorio $U(i, j)$, cuyas componentes siguen una distribución uniforme con rango de valores determinado por la extensión del plano de correlación, y $\gamma(i, j)$ una

variable aleatoria, con valores posibles 0 y 1 y tal que $P[\gamma(i, j) = 0] = p(i, j)$ representa la probabilidad de ocurrencia de un vector espurio. De esta forma, el modelo propuesto para $\check{V}(i, j)$ toma la forma

$$\check{V}(i, j) = \gamma(i, j)[\bar{V}(i, j) + \varepsilon(i, j)] + [1 - \gamma(i, j)]U(i, j).$$

En la práctica, la probabilidad de ocurrencia de vectores espurios puede reducirse diseñando cuidadosamente el experimento, pero inevitablemente siempre existirá una probabilidad finita de ocurrencia de estos vectores. Por lo tanto, es necesario emplear algún método automático que identifique los vectores espurios. Las técnicas empleadas para tal fin se reconocen como técnicas de validación. Se han propuesto distintas técnicas de validación, que en general se basan en analizar el grado de coherencia de cada vector estimado con respecto a sus vectores vecinos. En las siguientes secciones se detallan los distintos métodos de validación tenidos en cuenta.

11.8.1. Validación por medio de valor umbral

Este método de validación consiste en definir un rango de valores aceptable para las componentes de desplazamiento en x y en y que se aplica sobre toda la grilla. Los vectores estimados que posean el valor de una o ambas componentes fuera de este rango se clasifican como vectores espurios. Este no es un método de validación robusto, pero sirve para aislar los vectores espurios con valores más extremos en un primer paso de validación, antes de aplicar un método de validación más fino.

11.8.2. Detectores basados en la media y la mediana

Los detectores basados en la media y la mediana son una familia de detectores presentados y analizados por Westerweel (Westerweel, 1994; Westerweel & Scarano, 2005). En el trabajo de 1994, Westerweel propone realizar la validación en base al cálculo de unos residuos $r(i, j)$ dados por

$$r(i, j) = |\check{V}(i, j) - \langle \check{V}(i, j) \rangle|,$$

donde $\check{V}(i, j)$ es el vector de desplazamiento estimado en el punto de grilla (i, j) , $\langle \check{V}(i, j) \rangle$ es una cantidad representativa del valor del vector de desplazamiento medio, $\bar{V}(i, j)$, y $|\mathbf{u}|$ denota el vector cuyas componentes son el valor absoluto de las componentes del vector \mathbf{u} . El valor residual puede entonces compararse con un valor umbral como criterio de clasificación del vector en válido o espurio. Si uno o ambos componentes de $r(i, j)$ superan el valor umbral, el vector es clasificado como espurio. Se proponen 3 formas para calcular $\langle \check{V}(i, j) \rangle$, basadas en la media global, la media y la mediana locales. En el trabajo de Westerweel (1994) se comparan estos tres métodos por medio de un análisis teórico basado en la estadística de ocurrencia de vectores espurios, y también por medio de ensayos sobre imágenes sintéticas, donde se concluye que el método más robusto es el basado en la mediana local.

Posteriormente, Westerweel & Scarano (2005) propusieron un nuevo esquema de validación basado en el método de la mediana local, al cual denominan detector universal de vectores espurios o UOD (del inglés, Universal Outlier Detector). Sea $\tilde{\mathbf{V}}_0$ un vector de desplazamientos estimado, el cual se desea validar. Se toman los vectores de desplazamiento estimados en una vecindad de $\tilde{\mathbf{V}}_0$ de 3×3 puntos de grilla. Este conjunto de vectores se denota por $\{\tilde{\mathbf{V}}_1, \tilde{\mathbf{V}}_2, \dots, \tilde{\mathbf{V}}_8\}$. Se calcula $\tilde{\mathbf{V}}_m$ como la mediana de dicho conjunto. A continuación, se calculan los residuos r_i como

$$r_i = |\tilde{\mathbf{V}}_i - \tilde{\mathbf{V}}_m|, \text{ con } i = 1, 2, \dots, 8.$$

Se calcula la mediana residual r_m como la mediana del conjunto $\{r_1, r_2, \dots, r_8\}$. Finalmente, se calcula el residuo normalizado como

$$r_0 = \frac{|\tilde{\mathbf{V}}_0 - \tilde{\mathbf{V}}_m|}{r_m + \varepsilon},$$

donde la división es componente a componente, y ε es el nivel de normalización mínimo, que se incluye para compensar el efecto que produciría la división por r_m en zonas del flujo con muy baja turbulencia en donde $r_m \rightarrow \mathbf{0}$. Los autores encontraron conveniente el empleo de $\varepsilon = 0,1$ píxeles, que se corresponde con el nivel de error cuadrático medio típico en estimaciones PIV (Westerweel, 2000). En el artículo de Westerweel se muestra que el residuo calculado de la forma propuesta tiene una distribución de probabilidades aproximadamente “universal”, por lo que se puede elegir un único valor de umbral como criterio de validación, válido para cualquier experimento. Esto lo demuestran aplicando el método de residuo normalizado a una gran variedad de experimentos PIV normalizados, con números de Reynolds desde 0,1 a 10^7 . Los autores encuentran apropiado el empleo del valor 2 como umbral de comparación para el residuo normalizado. Si alguno de los componentes de r_0 excede este valor, el vector es clasificado como espurio.

Algunas variantes de este método fueron propuestas. Por ejemplo, Shinneeb, Bugg, & Balachandar (2004), propusieron un método basado en el residuo normalizado, pero empleando un valor de umbral adaptativo para cada punto de la grilla, determinado en función del gradiente local del campo. El método consiste en calcular un campo de desplazamientos provisional empleando una técnica de detección de vectores espurios agresiva. Los vectores clasificados como espurios en este campo provisional son reemplazados por una media ponderada gaussiana a partir de los vectores vecinos. A partir del campo resultante en este paso se calcula el valor de umbral adaptativo para cada punto de la grilla, haciendo uso de la diferencia entre vectores vecinos como una medida de la variabilidad local del flujo. Antes de su aplicación, la grilla de valores de umbral es suavizada por medio de una media móvil de ventana gaussiana.

Otra modificación introducida sobre el método de residuo normalizado fue propuesta por Duncan, Dabiri, Hove, & Gharib (2010). Este método es una generalización del método UOD que busca ser aplicable en grillas no uniformes, como

las que resultan en aplicaciones de seguimiento individual de partículas o PTV (del inglés, Particle Tracking Velocimetry) y esquemas híbridos como el PIV super resolución (sección III). Esto es posible incluyendo la distancia entre los puntos de la grilla como factor de ponderación. Sea $\tilde{\mathbf{V}}_0$ un vector de desplazamientos estimado, el cual se desea validar. Sea $\{\tilde{\mathbf{V}}_1, \tilde{\mathbf{V}}_2, \dots, \tilde{\mathbf{V}}_N\}$ un conjunto de N vectores de desplazamiento estimados vecinos a $\tilde{\mathbf{V}}_0$, y sea $\{d_1, d_2, \dots, d_N\}$ el conjunto de distancias de $\tilde{\mathbf{V}}_0$ a cada uno de los vectores vecinos del conjunto anterior, medidas en píxeles. Se define la cantidad d_m como la mediana del conjunto de distancias. Se calcula el conjunto de desplazamientos ponderados, $\{\tilde{\mathbf{V}}_1^*, \tilde{\mathbf{V}}_2^*, \dots, \tilde{\mathbf{V}}_N^*\}$, donde

$$\tilde{\mathbf{V}}_i^* = \frac{\tilde{\mathbf{V}}_i}{d_i + \varepsilon_a}, \text{ con } i = 1, 2, \dots, N,$$

siendo ε_a un factor de tolerancia adaptativo. Luego, se calcula la cantidad $\tilde{\mathbf{V}}_m^*$ como la mediana del conjunto de desplazamientos ponderados. Por otro lado, el desplazamiento ponderado $\tilde{\mathbf{V}}_0^*$ se calcula como

$$\tilde{\mathbf{V}}_0^* = \frac{\tilde{\mathbf{V}}_0}{d_m + \varepsilon_a}.$$

Los residuos son calculados como

$$r_i = |\tilde{\mathbf{V}}_i^* - \tilde{\mathbf{V}}_m^*|, \text{ con } i = 1, 2, \dots, N,$$

y se determina la cantidad r_m como la mediana del conjunto de residuos. Finalmente, se calcula el residuo normalizado de $\tilde{\mathbf{V}}_0$ como

$$r_0 = \frac{|\tilde{\mathbf{V}}_0^* - \tilde{\mathbf{V}}_m^*|}{r_m + \varepsilon_a}.$$

El valor de ε_a se establece de forma adaptativa para cada vector a validar, de forma de que se cumpla

$$\varepsilon_a(\varepsilon_a + d_m) = \varepsilon,$$

siendo ε el valor de normalización mínimo definido para el método de UOD. El factor ε_a definido de esta forma garantiza que este método resulte equivalente al método UOD si se toma el mismo valor de distancia para todos los vecinos.

11.8.3. Detección basada en redes neuronales

Liang, Jiang, & Li (2003) propusieron un método de detección de vectores espurios basado en una red neuronal celular. La arquitectura de esta red está caracterizada por una única capa de $M \times N$ neuronas operando en paralelo, dispuestas en un arreglo de grilla regular bidimensional de M filas por N columnas, con conexiones de retroalimentación locales. Se define el valor r como el orden de interconexión local de

las unidades neuronales. $r = 1$ implica que cada neurona esta interconectada únicamente con sus 8 vecinos, $r = 2$ implica que cada neurona esta interconectada con sus 8 vecinos y los vecinos de estas, y así sucesivamente. La salida de cada neurona puede ser 0 o 1. La red se construye de forma de asimilar cada unidad neuronal a un vector de desplazamiento del campo estimado, con las dimensiones M y N coincidentes a las dimensiones de la grilla del campo de desplazamientos. Se busca entonces que la salida de cada neurona sea 0 si el vector es un vector espurio, o 1 en caso contrario.

El algoritmo comienza inicializando todas las salidas neuronales en 1. Luego, la salida de cada neurona de la red se actualiza de forma iterativa hasta que la red converja a un estado estable. Sea $U_i = [u_i, v_i]$ el vector de desplazamiento estimado asociado a la neurona i , y sea $y_i(n)$ la salida de la unidad neuronal i durante el paso n , con $i = 1, 2, \dots, MN$. La salida de la neurona y_i durante el paso $n + 1$ se actualiza como

$$y_i(n + 1) = H\left(\sum_{j=1}^{MN} Wu_{ij}y_j(n)\right)H\left(\sum_{j=1}^{MN} Wv_{ij}y_j(n)\right),$$

donde $H(x)$ es la función escalón definida por

$$H(x) = \begin{cases} 1 & \text{si } x \geq 0 \\ 0 & \text{si } x < 0 \end{cases}$$

y Wu_{ij} y Wv_{ij} son los pesos de interconexión de las neuronas i y j , calculados como

$$Wu_{ij} = \max(-T, T - |u_i - u_j|),$$

$$Wv_{ij} = \max(-T, T - |v_i - v_j|),$$

donde T es un valor de umbral. Por otro lado Wu_{ij} y Wv_{ij} valen cero para neuronas no interconectadas (puesto a que no se influyen entre sí), y para $i = j$ (puesto a que no hay autoexcitación). El algoritmo se puede resumir en los siguientes pasos:

1. Inicializar la salida de todas las neuronas en 1. Calcular el valor de los pesos Wu_{ij} y Wv_{ij} para todas las neuronas i y j interconectadas.
2. Tomar una neurona al azar y actualizar su salida según la expresión de actualización.
3. Repetir el paso 2 con una neurona diferente hasta que todas las neuronas de la red hayan sido actualizadas. Esto constituye un paso de iteración.
4. Si al completar el paso de iteración, el estado de la salida de las neuronas no ha cambiado con respecto al estado de salidas al inicio de la iteración, entonces la red ha convergido, y el algoritmo finaliza. Caso contrario, retomar el paso 2 y realizar un nuevo paso de iteración.

En ocasiones puede ocurrir que la red no converja a un único estado, sino que permanezca oscilando entre 2 o más estados. Por eso es conveniente aplicar un criterio de finalización basado en un número máximo de pasos de iteración para evitar que el algoritmo quede encerrado en un ciclo infinito.

Para un entendimiento más claro de cómo se comporta la red, se analizan distintos escenarios para la interacción entre dos unidades neuronales. Primero, suponer que dos neuronas i y j tienen la salida correcta. Si ambos vectores U_i y U_j son válidos, entonces las cantidades $|u_i - u_j|$ y $|v_i - v_j|$ serán pequeñas, luego, Wu_{ij} y Wv_{ij} tenderán a adoptar un valor positivo, por lo que las neuronas contribuirán mutuamente a mantener su estado de salida en 1. Si U_i es un vector válido, pero U_j es espurio, entonces las cantidades $|u_i - u_j|$ y $|v_i - v_j|$ serán grandes, luego, Wu_{ij} y Wv_{ij} tenderán a adoptar un valor negativo. Esto contribuirá a mantener el estado de salida de la neurona j en 0, mientras que no contribuirá con la salida de la neurona i (puesto a que $y_j = 0$). Finalmente, si ambos son vectores espurios, entonces las neuronas no tendrán ningún tipo de interacción (puesto a que $y_i = y_j = 0$).

En segundo lugar, se plantea el escenario en el que ambas neuronas tienen la salida opuesta a la que deberían. Si ambos vectores U_i y U_j son válidos, pero las salidas de las neuronas i y j están en 0, entonces estas no contribuirán entre sí, por lo que los estados deberán cambiar en función de otras conexiones de la red. Si U_i es un vector válido, pero U_j es espurio, entonces Wu_{ij} y Wv_{ij} tenderán a adoptar valores negativos. Puesto que $y_i = 0$ y $y_j = 1$, el estado de la neurona j no se verá afectado por el estado de la neurona i , y la neurona j contribuirá a mantener el estado de la neurona i . Este constituye una situación indeseable. Finalmente, si ambos vectores son inválidos, Wu_{ij} y Wv_{ij} tenderán a adoptar valores negativos, y las neuronas contribuirán mutuamente a cambiar sus estados a 0.

En tercer y último lugar, se consideran los escenarios en donde una de las salidas es correcta y la otra es opuesta a la esperada. Si ambos vectores U_i y U_j son válidos, $y_i = 1$ y $y_j = 0$, entonces Wu_{ij} y Wv_{ij} tenderán a adoptar un valor positivo. La neurona j no contribuirá a modificar el estado de la neurona i , pero la neurona i contribuirá a modificar el estado de la neurona j a 1. Si U_i es un vector válido, pero U_j es espurio, entonces Wu_{ij} y Wv_{ij} tenderán a ser negativos. Si en este escenario, $y_i = 1$ y $y_j = 1$ entonces ambas neuronas contribuirán mutuamente a cambiar su estado a 0. Por otro lado, si $y_i = 0$ y $y_j = 0$, las unidades no interactuarán entre sí. Finalmente, si ambos vectores U_i y U_j son espurios, $y_i = 1$ y $y_j = 0$, Wu_{ij} y Wv_{ij} tenderá a ser negativo, la unidad i contribuirá a mantener la salida de j en 0, mientras que la unidad j no tendrá efecto sobre la unidad i .

Un método de validación alternativo, basado en la técnica de la red neuronal celular fue propuesto por Shinneeb, Bugg, & Balachandar (2004). Este se basa en aplicar la técnica tal como se detalló en esta sección, pero con un valor umbral adaptativo para

cada punto de la grilla, determinado a partir de los gradientes de desplazamiento locales del campo.

11.8.4. Bootstrapping

El método de Bootstrapping para la validación de campos PIV, propuesto por Pun, Susanto, & Dabiri (2007), es un método basado en la generación de estadísticas a partir del campo estimado, para la detección de vectores espurios. En forma general, el bootstrapping es un método aplicado para realizar inferencia estadística. La técnica consiste en realizar un muestreo aleatorio de una población de datos, extrayendo aleatoriamente un subconjunto de esta. El muestreo se efectúa con reemplazo, en el sentido de que luego de extraer una muestra, los elementos extraídos se reincorporan a la población de datos para que puedan ser reelegidos en una muestra posterior. Para cada muestra extraída, se calculan los estadísticos deseados. El proceso de muestreo y cálculo de estadísticos se repite una determinada cantidad de veces. Al finalizar, se analiza la distribución de estadísticos obtenida para realizar inferencias.

El método implementado por los autores se basa en muestrear un porcentaje p de vectores pertenecientes al campo de desplazamientos estimado. Luego, los componentes de los vectores muestreados se ajustan a una superficie en dos dimensiones por mínimos cuadrados, para calcular los vectores restantes y generar un nuevo campo de velocidades. Este campo de velocidades es almacenado, ya que sus vectores son los que se usarán posteriormente para la validación. El proceso se repite un total de N veces. Al finalizar, se procede a analizar cada vector por separado para su validación. Sea \check{V}_i un vector a ser validado, y sea $\{V_{i,1}^{bt}, V_{i,2}^{bt}, \dots, V_{i,N}^{bt}\}$ el conjunto de vectores ajustados por el procedimiento de bootstrapping, asociados al punto de grilla de \check{V}_i . Si se realiza un histograma de cada uno de los componentes de desplazamiento de este conjunto de vectores, se observará en general un histograma unimodal si \check{V}_i es un vector válido, con el pico en torno a las componentes del desplazamiento medio real, \bar{V}_i . Por otro lado, se observará un histograma bimodal en el caso de que \check{V}_i sea espurio, con un pico estrecho en torno a las componentes de \bar{V}_i , y un pico más ancho y de menor intensidad en torno a las componentes de \check{V}_i (originado a partir de los muestreos en los que el \check{V}_i fue incluido en la muestra). Esto permite, en primer lugar, utilizar la moda de los histogramas para construir una estimación de \bar{V}_i , y, en segundo lugar, establecer un criterio de validación basado en la diferencia entre \check{V}_i y dicha estimación.

El criterio de validación propuesto viene dado por

$$\left| \frac{x_i - x_m}{x_m} \right| < t,$$

donde x_i es alguna de las componentes de desplazamiento del vector \check{V}_i , x_m es la moda del histograma del conjunto de vectores de desplazamiento ajustados, asociado a dicho componente, y t es un valor de umbral. La cantidad del miembro izquierdo de la inequación se denomina razón de moda. Si todas las componentes del vector \check{V}_i cumplen

con este criterio, entonces el vector es clasificado como válido, de lo contrario es considerado espurio.

Debido a que, en zonas con desplazamientos bajos, $x_m \rightarrow 0$, la razón de moda tiende a infinito, incluso para valores de $|x_i - x_m|$ bajos. Por lo tanto, se aplica un criterio adicional, dado por

$$|x_i - x_m| < tol,$$

donde tol es un valor de tolerancia. Si la componente en cuestión cumple con este criterio, entonces la componente es tomada como válida, sin aplicar el criterio de la razón de moda.

otro parámetro del método es el ancho de los intervalos utilizados para la construcción del histograma. Los autores proponen calcular el ancho de los intervalos de forma adaptativa para cada localización i , por medio de la expresión

$$b_i^x = \frac{\max_j(x_{i,j}^{bt}) - \min_j(x_{i,j}^{bt})}{N/2},$$

donde $x_{i,j}^{bt}$ representa la componente x del vector $V_{i,j}^{bt}$.

Finalmente, se propone un esquema de validación multietapa, que implica emplear M sucesivas etapas de validación por bootstrapping, partiendo de un valor de umbral inicial t_0 relativamente alto para eliminar solo los vectores espurios con mayores desvíos, y este valor de umbral se reduce progresivamente en las posteriores etapas para refinar la validación. La expresión propuesta para calcular el valor de umbral de la etapa de validación m viene dado por

$$t(m) = t_0 - t_0 \frac{(m-1)}{(M-1)}.$$

Los autores realizaron ensayos utilizando distintas combinaciones de parámetros sobre campos de desplazamientos generados de forma sintética, a los que se les agregaron vectores espurios. Una combinación de parámetros adecuada es $p = 15\%$, $t_0 = 0,2$, $tol = 0,0025$, $N = 700$ y $M = 12$.

11.8.5. Método todo en uno

El método todo en uno o AIO (del inglés, All In One) propuesto por Garcia (2011), es un método de validación, que además permite simultáneamente efectuar el paso de reemplazo de vectores espurios y suavizar el campo de desplazamientos para reducir el nivel de ruido aleatorio. La técnica se basa en un ajuste no paramétrico del campo de desplazamientos estimado, $\tilde{V} = [\tilde{u}, \tilde{v}]$, a un campo suave $\tilde{V}_s = [\tilde{u}_s, \tilde{v}_s]$, por medio de un ajuste de mínimos cuadrados por penalización, con las funciones de error dadas por

$$E_u(\tilde{\mathbf{u}}_s) = \sum_{i,j} (\tilde{u}(i,j) - \tilde{u}_s(i,j))^2 + s P(\tilde{\mathbf{u}}_s),$$

$$E_v(\tilde{\mathbf{v}}_s) = \sum_{i,j} w(i,j) [\tilde{v}(i,j) - \tilde{v}_s(i,j)]^2 + s P(\tilde{\mathbf{v}}_s),$$

donde w es una función de pesos, s es un parámetro de suavizado, y P es una función cuyo valor es proporcional a la rugosidad de la señal de entrada, basando la medida de rugosidad en las derivadas de segundo orden de la señal. El primer término de los miembros de la derecha de estas ecuaciones son una medida de la fidelidad del ajuste, que mide que tanto se aproximan los valores de $\tilde{\mathbf{V}}_s$ a los valores del campo estimado original $\tilde{\mathbf{V}}$, mientras que el segundo término es una medida de la suavidad de $\tilde{\mathbf{V}}_s$. Está claro que si s es igual a 0, entonces la solución es trivial, puesto a que si se hace $\tilde{\mathbf{V}}_s = \tilde{\mathbf{V}}$, entonces el valor de las funciones de error se vuelve 0. Por otro lado, para valores mayores de s , el término de rugosidad gana peso y entonces la solución que garantiza la minimización del error será un compromiso entre suavidad del campo resultante y fidelidad de los datos.

Los detalles matemáticos de la aplicación de este método, y la forma óptima de selección del parámetro s se describen en el anexo IX.9.

II.9. REEMPLAZO DE VECTORES ESPURIOS

Una vez validado el campo de desplazamientos, y clasificados los vectores en válidos y espurios, es posible remover los vectores inválidos. No obstante, para el reporte de resultados, cálculo de cantidades derivadas, y para posteriores etapas de procesamiento en métodos de pasos múltiples (sección II.11), es conveniente mantener el formato de los resultados en forma de una grilla uniforme, por lo que los “huecos” generados por la remoción de los vectores espurios deben ser rellenados con valores adecuados. En la literatura se proponen algunos métodos para este fin.

Una técnica posible es emplear algún algoritmo de interpolación de puntos dispersos, para obtener el valor de los vectores faltantes por interpolación a partir de los vectores válidos. Entre los métodos de interpolación de puntos dispersos se encuentra la interpolación lineal (Raffel, y otros, 2018), y la interpolación de vecinos naturales.

Otra opción es emplear un filtro de media móvil, y calcular los vectores faltantes como medias ponderadas de los vectores válidos de su entorno. Por ejemplo, Agüí & Jimenez (1987) propusieron emplear este método utilizando una función gaussiana para los pesos.

Finalmente, se pueden mencionar los métodos basados en la graduación por medio del ajuste de los vectores válidos a una función suavizada utilizando mínimos cuadrados con penalización. Esto no solo permite obtener el valor de los datos faltantes, sino que funcionan como una operación de suavizado (sección II.10), ya que el campo obtenido

ajusta a los vectores válidos en el sentido de mínimos cuadrados, pero no es interpolante de los mismos. Entre estos métodos se encuentra el método de todo en uno propuesto por Garcia (2011), el cual por medio de un ajuste del campo de desplazamientos estimados crudo utilizando mínimos cuadrados con penalización, efectúa simultáneamente las operaciones de validación, reemplazo de vectores espurios y suavización (sección II.8.5).

II.10. SUAVIZADO

Debido al error inherente de los métodos de correlación y estimación subpíxel, el campo de desplazamientos, aún luego de la validación y reemplazo de espurios, está contaminado con ruido. Esta contaminación puede ser perjudicial en posteriores etapas de procesamiento como por ejemplo en la estimación de cantidades derivadas como el gradiente, en donde el ruido puede verse amplificado. Por tal motivo, puede ser conveniente aplicar una operación de suavizado con el objetivo de reducir el nivel de ruido.

Entre las técnicas de suavizado se encuentran la aplicación de una media móvil, que equivale a convolucionar el campo vectorial con una ventana de suavizado. Para este fin, es conveniente utilizar una ventana de suavizado con un tamaño inferior al tamaño efectivo de la ventana de interrogación utilizada por el algoritmo de correlación cruzada, para minimizar un efecto de filtro pasa bajos adicional al ya introducido por la técnica PIV (Raffel, y otros, 2018).

Otro método implica ajustar el campo vectorial por mínimos cuadrados con penalización. Entre estas técnicas se encuentra la técnica de todo en uno propuesta por Garcia (2011) (sección II.8.5).

II.11. PROCESAMIENTO DE PASOS MÚLTIPLES

Como se mencionó en secciones previas, la operación de estimación del desplazamiento a partir de la correlación cruzada de un par de ventanas de interrogación genera una estimación del desplazamiento vinculada al desplazamiento promedio de las imágenes de partículas dentro de la ventana, en lugar de estar asociada al desplazamiento real del campo evaluado en cada punto de la grilla. De esta forma, el campo de desplazamiento estimado se puede interpretar como una versión filtrada del campo original por medio de un filtro pasa bajos espacial, al cual además se le ha agregado ruido. El efecto de filtrado espacial está relacionado con el tamaño efectivo de las ventanas de interrogación, y este puede disminuirse reduciendo el tamaño de estas. No obstante, existen restricciones que impiden reducir las ventanas de interrogación por debajo de cierto límite. En primer lugar, debido a la naturaleza cíclica de la función de correlación cruzada obtenida mediante la técnica de Transformada de Fourier, solo es posible estimar los desplazamientos en el rango $[-N/2, N/2]$, siendo N el tamaño de la ventana de correlación. Esta limitación puede reducirse si se aplica rellenado con ceros sobre las ventanas. Sin embargo, el efecto de pérdida de pares producido por los

movimientos de las partículas en el plano (algunas partículas de la primera ventana salen de los límites de la segunda ventana debido a su desplazamiento en el plano) genera un deterioro en la amplitud del pico correspondiente al desplazamiento medio, dificultando su detección satisfactoria. Por lo tanto, el tamaño de la ventana debe ser lo suficientemente grande para minimizar este efecto. En la práctica se aplica la denominada regla de un cuarto, que establece que el tamaño efectivo de las ventanas de correlación debe ser mayor o igual a un cuarto del desplazamiento de las imágenes de partículas dentro de la ventana.

Reducir el desplazamiento de las imágenes de partículas mediante la modificación de las condiciones de flujo no suele ser una opción, ya que es la condición de flujo particular la que se desea estudiar a partir del experimento. Otra forma de reducir el desplazamiento de las imágenes es mediante la reducción del intervalo de tiempo entre pulsos láseres. No obstante, esto trae aparejado una reducción en el rango dinámico del desplazamiento (Adrian, 1997). Las técnicas de procesamiento que permiten aumentar la resolución espacial sin necesidad de modificar las condiciones experimentales son las técnicas de procesamiento de pasos múltiples, que se discuten en las secciones siguientes.

II.11.1. Desplazamiento de ventanas discreto

Una técnica efectiva para aumentar la resolución espacial, denominada desplazamiento discreto de ventanas o DWO (del inglés, Discrete Window Offset) fue propuesta inicialmente por Keane & Adrian (1993) y posteriormente aplicada por Soria (1996), Westerweel, Dabiri, & Gharib (1997) y Scarano & Riethmuller (1999).

La técnica consiste en realizar un proceso iterativo en sucesivos pasos de procesamiento PIV, empleando la información del campo estimado en un paso previo como entrada en el paso siguiente. Inicialmente, no se tiene información sobre el flujo, por lo que se ejecuta una etapa de procesamiento PIV consistente en los pasos discutidos en las secciones II.6 a II.10 (correlación cruzada, estimación subpíxel, validación, reemplazo de vectores espurios y suavizado). El tamaño inicial de las ventanas se determina de forma de cumplir con la regla de un cuarto, mientras que el espaciamiento de la grilla se establece en función del nivel de solapamiento establecido por el usuario. El procesamiento PIV a partir del segundo paso de iteración se ejecuta como sigue. Sea $\check{V}(i, j) = [u(i, j), v(i, j)]$ el vector de desplazamiento estimado en el paso previo para el punto de grilla (i, j) , con componentes horizontal y vertical $u(i, j)$ y $v(i, j)$, respectivamente. Sean Δx y Δy los valores enteros más cercanos a $u(i, j)$ y $v(i, j)$, de forma respectiva. La ventana de interrogación correspondiente al primer cuadro se toma centrada en el punto (i, j) , mientras que la ventana de interrogación del segundo cuadro se toma centrada en el punto $(i + \Delta x, j + \Delta y)$. Al tomar las ventanas de esta forma, se logra compensar el efecto de pérdida de pares de partículas, eliminando la restricción impuesta por la regla de un cuarto. Esto permite seleccionar tamaños de ventana más pequeñas en pasos PIV posteriores, alcanzando mayores resoluciones espaciales que las posibles con métodos de un solo paso.

Una variante más precisa de la técnica de desplazamiento de ventana discreto fue propuesta más adelante por Wereley & Meinhart (2001). La ubicación espacial más apropiada para el vector de desplazamiento calculado por PIV se encuentra desplazada a partir del centroide de la primera ventana de interrogación, por una cantidad igual a la mitad del vector de desplazamiento estimado (Spedding & Rignot, 1993). Aplicar esto implicaría que el campo de desplazamientos estimado quede definido sobre una grilla irregular, lo cual no es conveniente. Wereley & Meinhart propusieron implementar la técnica de desplazamiento de ventanas discreto, pero desplazando ambas ventanas de forma simétrica, tomando la primera ventana de interrogación centrada en el punto $(i - \Delta x/2, j - \Delta y/2)$, y la segunda ventana centrada en el punto $(i + \Delta x/2, j + \Delta y/2)$. De esta forma, la distancia entre los centroides de ambas ventanas queda dada por $[\Delta x, \Delta y]$, correspondiente con el desplazamiento medio local de las imágenes de partículas, y se cumple que el punto asignado para el vector de desplazamiento, (i, j) , se encuentra desplazado del centroide de la primera ventana de interrogación según el vector $[\Delta x/2, \Delta y/2]$. Los autores reportaron que el método de desplazamiento de ventanas discreto implementado de esta forma tiene una precisión del orden de Δt^2 , siendo Δt el intervalo de tiempo entre pulsos láser, mientras que el método de DWO implementado según el párrafo anterior presenta una precisión del orden de Δt . Esto fue probado mediante simulaciones y resultados experimentales.

II.11.2. Deformación de imágenes

El desplazamiento de las imágenes de partículas es medido mediante el método de correlación cruzada bajo la suposición de que el movimiento de las partículas es aproximadamente uniforme dentro de las ventanas de interrogación. En la práctica, sin embargo, esta suposición no se cumple debido a que la mayoría de los flujos de interés presentan distribuciones de velocidad con elevados gradientes. Esto ocasiona que los desplazamientos de las partículas dentro de las ventanas de interrogación difieran entre sí, generando una disminución en el grado de emparejamiento entre las imágenes de partículas de los pares de ventanas, lo cual en última instancia genera un ensanchamiento de los picos de correlación, una disminución de la relación señal ruido, o, para gradientes más elevados, la división de un pico de correlación en varios picos. Estos efectos traen como consecuencia tanto un aumento en la incertidumbre de las estimaciones, como así también una mayor ocurrencia de vectores espurios.

Un método adecuado para compensar los efectos de los gradientes de desplazamiento en el flujo es el conocido como método de deformación de imágenes (Jambunathan, Ju, Dobbins, & Ashforth-Frost, 1995; Scarano, 2001). Este es un método de procesamiento de pasos múltiples que utiliza la información del campo de desplazamientos estimado en pasos previos para aplicar una deformación sobre las imágenes, que tienda a reducir los gradientes de desplazamiento dentro de las ventanas.

El método de deformación de imágenes se aplica como sigue. Al inicio del procesamiento no se tiene información sobre el flujo, por lo que se aplica un procesamiento PIV según las etapas discutidas en las secciones II.6 a II.10 (correlación

cruzada, estimación subpíxel, validación, reemplazo de vectores espurios y suavizado). El tamaño de las ventanas se selecciona de forma de cumplir con la regla de un cuarto y el espaciamiento de la grilla se selecciona según el factor de solapamiento establecido. A partir del segundo paso de iteración PIV en adelante, se utiliza el campo de desplazamientos estimado en el paso previo para aplicar una deformación sobre las imágenes. El procesamiento posterior se ejecuta sobre estas imágenes deformadas. Al deformar las imágenes según el campo de desplazamientos estimado, se compensa el desplazamiento relativo de las partículas entre las ventanas de interrogación. Esto elimina el efecto de pérdida de pares de partículas por movimientos en el plano, y por lo tanto permite utilizar ventanas de correlación más pequeñas al eliminar la restricción impuesta por la regla de un cuarto. Por otro lado, a diferencia del método de desplazamiento de ventanas discreto, también compensa los efectos de gradientes de desplazamiento dentro de la ventana, dando lugar a picos de correlación más estrechos y con mayor relación señal ruido, y reduciendo la ocurrencia de vectores espurios.

Para efectuar la deformación de las imágenes primero es necesario interpolar el campo de desplazamientos estimado, que está evaluado solo para los puntos de la grilla, a cada píxel de la imagen. Luego, sean (i, j) las coordenadas de un píxel de la imagen, y sean $u(i, j)$ y $v(i, j)$ las componentes horizontal y vertical, respectivamente, del vector de desplazamiento interpolado en el punto (i, j) . Sean I_1 e I_2 el par de imágenes siendo procesadas, y $I_1(i, j)$ e $I_2(i, j)$ las intensidades del píxel (i, j) de cada imagen. Las imágenes deformadas I_1^* y I_2^* se obtienen como

$$I_1^*(i, j) = I_1(i, j),$$

$$I_2^*(i, j) = I_2\left(i + u(i, j), j + v(i, j)\right),$$

si se emplea un esquema de primer orden, o

$$I_1^*(i, j) = I_1\left(i - \frac{1}{2}u(i, j), j - \frac{1}{2}v(i, j)\right),$$

$$I_2^*(i, j) = I_2\left(i + \frac{1}{2}u(i, j), j + \frac{1}{2}v(i, j)\right),$$

si se emplea un esquema de segundo orden. Esto requiere conocer el valor de las imágenes en el espacio entre píxeles, por lo que se requiere de algún método de interpolación de imágenes.

III. SUPER RESOLUCIÓN

En la técnica PIV, la resolución espacial está limitada por el tamaño de las ventanas de interrogación utilizadas para efectuar la correlación cruzada. Esto se debe al efecto de filtro pasa bajos espacial que produce el método (Willert & Gharib, 1991). Por otro lado, en las técnicas basadas en el seguimiento puntual de partículas o PTV (del inglés, Particle Tracking Velocimetry), la resolución espacial está limitada por el desplazamiento máximo de las imágenes de partículas entre cuadros (debido a que no se tiene información de la posición intermedia de las partículas en las capturas) y por el espaciamiento promedio de las partículas.

Debido a que los métodos PIV requieren que haya varias imágenes de partículas por ventana de interrogación, está claro que los métodos PTV idealmente permitirían alcanzar una mayor resolución espacial. No obstante, la implementación de métodos PTV presenta mayor complejidad. Por un lado, se requiere de un algoritmo para la identificación de partículas. Esta es una etapa importante, ya que, si no se detecta un número adecuado de partículas, la resolución espacial obtenida se verá deteriorada. Por otro lado, se requiere de un algoritmo para emparejar las partículas entre pares de imágenes. Una técnica híbrida entre los métodos de PIV y PTV, que permite salvar parcialmente esta última dificultad es el método de PIV super resolución (Keane, Adrian, & Zhang, 1995).

La técnica de PIV super resolución o PIVSR se basa en efectuar, en primer lugar, un procesamiento PIV para estimar el campo de desplazamientos en una grilla regular sobre el plano de la imagen y, a continuación, utilizar este campo estimado para asistir al algoritmo PTV. El algoritmo PTV consta de una etapa de identificación de partículas, seguida de un algoritmo de seguimiento, para emparejar las partículas de la primera imagen con la partícula de la segunda imagen, y estimar el desplazamiento de cada partícula en función del cambio de posición entre cuadros. El emparejamiento de las partículas se realiza buscando para cada partícula en el primer cuadro, candidatos en el segundo cuadro, que se encuentren dentro de cierta área de búsqueda. Finalmente, se realiza un paso de validación para eliminar vectores espurios. Debido a que el campo de desplazamientos se obtiene en una grilla irregular, puede ser conveniente realizar una interpolación de este campo sobre una grilla regular. Al utilizar el campo de desplazamientos estimado por PIV como entrada en el método PTV, es posible inferir con mayor precisión la ubicación de cada partícula en el segundo cuadro. Esto permite reducir las áreas de búsqueda, y como consecuencia, reducir la posibilidad de emparejar incorrectamente las partículas.

En el presente trabajo se propuso implementar un método PIVSR como sigue. El campo de desplazamientos es estimado por medio de la técnica PIV con pasos múltiples, empleando las distintas etapas discutidas en las secciones II.3 a II.11. Para la identificación de partículas, se propone emplear el método de correlación por máscara de partícula, presentado por Takehara & Etoh (1998). El método consiste en calcular la

correlación cruzada normalizada (sección II.6.4) entre la imagen y una máscara generada por medio de una función gaussiana de la forma

$$f(x, y) = e^{-\frac{1}{2\sigma^2}[x^2+y^2]}.$$

Los valores de correlación obtenidos por medio de la correlación cruzada normalizada están comprendidos en el rango $[-1, 1]$. Este tipo de correlación es sensible a la distribución de patrones de intensidad en las imágenes, y no a la magnitud de esas intensidades. Esto significa que un elemento de la imagen que posea una distribución de intensidades similar a la de la máscara (patrón de intensidades concéntrico convexo) tendrá un valor de correlación cercano a 1, independientemente de su valor de máxima intensidad. En otras palabras, imágenes de partículas con distribución de intensidades similar producirán valores de correlación cercanos a 1 y semejantes, independientemente de sus picos de intensidad. Por otro lado, los elementos lineales de la imagen (como las superficies sólidas) y zonas de ruido de fondo presentarán valores de correlación bajos. Al establecer un valor de umbral, es posible establecer un criterio de detección de partículas, al seleccionar como los centros de las partículas a las ubicaciones en el plano de correlación que posean picos de correlación por encima de dicho valor umbral. Takehara & Etoh proponen utilizar un valor de $\sigma = 3$, un tamaño de máscara de $1,5\sigma \times 1,5\sigma$, y un valor de umbral de 0,7.

Una vez obtenidos los centros de las partículas en la primera imagen por medio de este método, se procede a determinar el vector de desplazamiento en estos puntos, por interpolación sobre el campo de desplazamientos estimado por PIV. Luego, para cada partícula, se toma una ventana de interrogación en la primera imagen, de tamaño $Ni_1 \times Nj_1$, centrada en la partícula, y una ventana de interrogación en la segunda imagen, de tamaño $Ni_2 \times Nj_2$, centrada en el punto desplazado desde el centro de la partícula según el vector de desplazamiento estimado. La primera ventana solo debe contener a la partícula. En general para aplicaciones PIV, el diámetro de partículas se encuentra comprendido entre 2 a 3 píxeles, por lo que tomar esta ventana de tamaño 3×3 o 5×5 resulta razonable. Por otro lado, la segunda ventana conforma el área de búsqueda, y la elección de sus dimensiones depende de que tan precisa sea la estimación previa del vector de desplazamiento.

A partir de las dos ventanas extraídas de esta forma se procede a calcular la correlación cruzada normalizada. Puesto que el área de búsqueda puede contener más de una partícula, el plano de correlación obtenido puede presentar más de un pico de correlación. De este plano se seleccionan los N_p picos de mayor valor, sobre los cuales se efectúa la estimación subpíxel. La estimación obtenida para el pico con mayor valor de correlación es seleccionada como el desplazamiento estimado de la partícula, mientras que los valores de desplazamiento estimados para los demás picos son almacenados para su uso posterior en la etapa de validación. El proceso de extracción de ventanas, correlación cruzada y estimación del desplazamiento se repite para cada partícula detectada en la primera imagen.

Una vez obtenida la nueva estimación de desplazamiento para cada partícula detectada, se procede a efectuar la etapa de validación. Puesto a que los vectores estimados se encuentran distribuidos aleatoriamente en el espacio, ya no conforman una grilla regular. Se deben utilizar métodos de validación adecuados como el método UOD generalizado para grillas no regulares (sección II.8.2). Si un vector es detectado como espurio, entonces se prueba el valor de los demás $N_p - 1$ desplazamientos estimados almacenados, obtenidos según se detalló en el párrafo anterior. Si alguno de estos vectores de desplazamiento pasa la validación, entonces el desplazamiento para la partícula es reemplazado por este nuevo valor. Por otro lado, si ninguno de estos vectores pasa la validación, entonces la estimación se descarta (se asume que no se encontró ninguna partícula aceptable dentro del área de búsqueda para emparejar con la partícula de la primera imagen).

Finalmente, el campo de desplazamientos validado, que conforma una grilla irregular, puede ser llevado a la forma de una grilla regular por alguna técnica de interpolación de puntos dispersos, o mediante una graduación con una superficie aproximada por mínimos cuadrados. Esta última técnica puede resultar más adecuada ya que ofrece además suavizado, atenuando el ruido presente en los vectores estimados (que en general presentan mayor nivel de ruido que las estimaciones obtenidas por PIV).

IV. SOFTWARE IMPLEMENTADO

Al comienzo de este trabajo, se optó por utilizar el software de acceso libre Prana, el cual se encuentra desarrollado en Matlab. El código de este software fue estudiado en detalle para conocer sus métodos internos de procesamiento y algoritmos utilizados, y para determinar sus ventajas y limitaciones, en base a todas las técnicas de procesamiento y algoritmos detallados en secciones anteriores. En el anexo IX.11 se muestra el análisis del código de Prana, en donde se describen cada una de las funciones del código relacionadas al procesamiento PIV, incluyendo sus argumentos de entrada y salida.

A partir del análisis realizado, se comprobó que Prana tiene una serie de ventajas, como ser una interfaz del usuario sencilla y compacta, posibilidad de empleo de máscaras estáticas y dinámicas, procesamiento PIV de pasos múltiples (desplazamiento de ventanas discreto y deformación de imágenes) que permite el ajuste del tamaño de ventanas y resolución de la grilla para cada paso, disponibilidad de métodos validación (por umbral, UOD y bootstrapping), y suavizado (media móvil con ventana gaussiana). Por otro lado, también se comprobaron sus desventajas y limitaciones, entre las que se encuentran algunas fallas en su interfaz gráfica, limitación en el uso de funciones de ventaneo (se aplica función de ventaneo gaussiana por defecto), método de validación basado en redes neuronales celulares no disponible (el cual se reporta que tiene mejor desempeño que el método UOD), umbral fijo para los métodos de validación (se reporta que los métodos de validación UOD y redes neuronales poseen mejor desempeño cuando se utilizan umbrales variables basados en el gradiente de desplazamiento), opciones de suavizado limitadas (solo se permite la opción de suavizado con una media móvil con ventana gaussiana), y método de super resolución no disponible. El programa tampoco cuenta con herramientas de preprocesamiento de imágenes, ni con una herramienta de visualización de resultados. Además, el programa tiene la opción de efectuar otros procesamientos como ser identificación y dimensionamiento de partículas, y seguimiento de partículas, pero estas herramientas no funcionan correctamente.

Debido a la necesidad de contar con una herramienta de procesamiento más completa, y la dificultad para modificar el código y la interfaz de Prana, en una instancia más avanzada del trabajo se decidió desarrollar un software propio, que cuente con todas las técnicas de procesamiento descritas en las secciones anteriores. El software fue desarrollado en Matlab, bajo el nombre de Bariloche PIV Processing. El programa incluye todas las técnicas de procesamiento descritas en este trabajo, incluyendo las de preprocesamiento de imagen, y también cuenta con una herramienta de previsualización de resultados, que permite obtener una vista del campo resultante del procesamiento de un par de imágenes antes de ordenar el procesamiento de toda la serie, lo cual es útil para el refinamiento de parámetros para cada caso en particular.

En la Figura 2 se muestra la pantalla principal del programa. Esta cuenta con los menús para acceder a las distintas herramientas disponibles. A través del menú, puede

accederse a la función de preprocesamiento de imagen (Figura 3), la cual permite seleccionar imágenes almacenadas en el ordenador y llevar a cabo tareas de sustracción de fondo, filtrado y mejora de contraste, y también permite generar una máscara estática a partir de la serie de imágenes seleccionada, a partir del método de prueba de normalidad descrito en la sección II.2.



Figura 2 – Bariloche PIV Processing. Pantalla principal del programa.

La otra función accesible desde el menú principal es la de procesamiento PIV (Figura 4), la cual cuenta con un panel a su izquierda en la que pueden gestionarse los distintos trabajos del usuario, permitiendo guardar, cargar, copiar o borrar los mismos. El otro panel situado a la derecha dispone de la configuración de opciones para el procesamiento. La primera pestaña de este panel contiene la descripción de las propiedades experimentales. En dicha pestaña puede ingresarse información acerca del experimento, que incluye datos de la cámara, los láseres, del fluido, las partículas, entre otras. En general esta información es opcional, pero la indicación del intervalo de tiempo entre pulsos láser y el factor de magnificación es necesaria para que el programa pueda calcular el campo de velocidades (en unidades de longitud sobre tiempo) a partir del campo de desplazamientos (en unidades de píxeles). La segunda pestaña del panel derecho se corresponde con las opciones de imagen (Figura 5), en donde se seleccionan las imágenes a ser procesadas, y las máscaras estáticas o dinámicas (si es que se utilizan). La tercera pestaña incluye las opciones para el procesamiento PIV (Figura 6), que comprenden la especificación del método PIV (desplazamiento de ventana discreto o deformación de imagen), opciones de grilla y tamaño de ventana, función de ventaneo, métodos de correlación y estimación subpíxel, método de validación y método de suavizado. El programa cuenta con una opción de previsualización (Figura 7), que permite visualizar el campo de desplazamientos resultante y los planos de correlación de cada punto de grilla para un par de imágenes a partir de una configuración dada. Esto permite ajustar los parámetros de procesamiento en cada caso antes de ordenar el

procesamiento de toda la serie de imágenes. La cuarta pestaña (Figura 8) incluye las opciones de guardado, en la cual es posible especificar qué información es deseable almacenar como salida. Finalmente, la quinta pestaña (Figura 9) muestra un seguimiento del estado de procesamiento, que incluye los tiempos promedio de cada paso de procesamiento, y el tiempo estimado de finalización del trabajo en curso. El software también cuenta con una guía de usuario (Figura 10) que contiene información destinada a orientar al usuario para un correcto uso del programa.

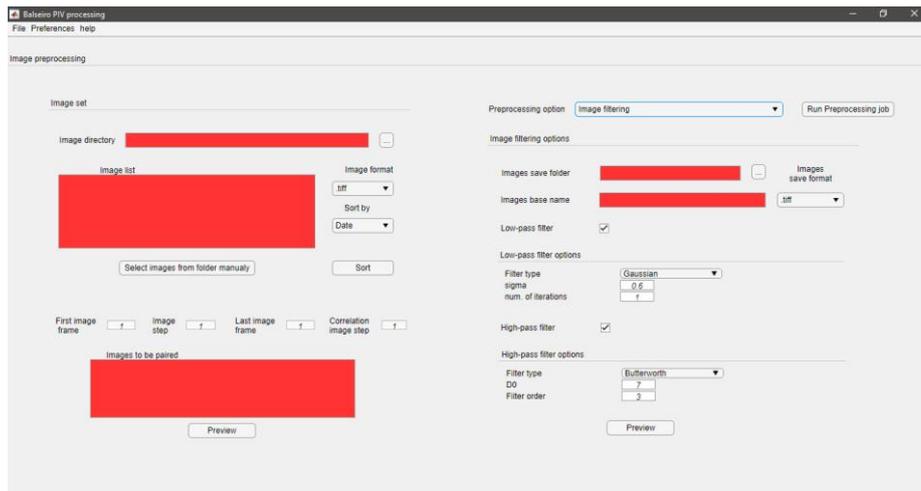


Figura 3 – Bariloche PIV Processing. Preprocesamiento de imágenes.

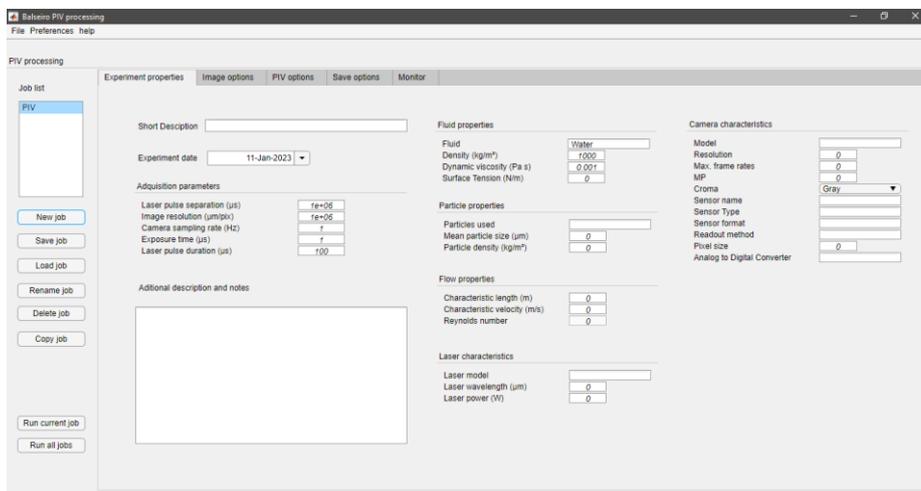


Figura 4 – Bariloche PIV Processing. Propiedades experimentales.

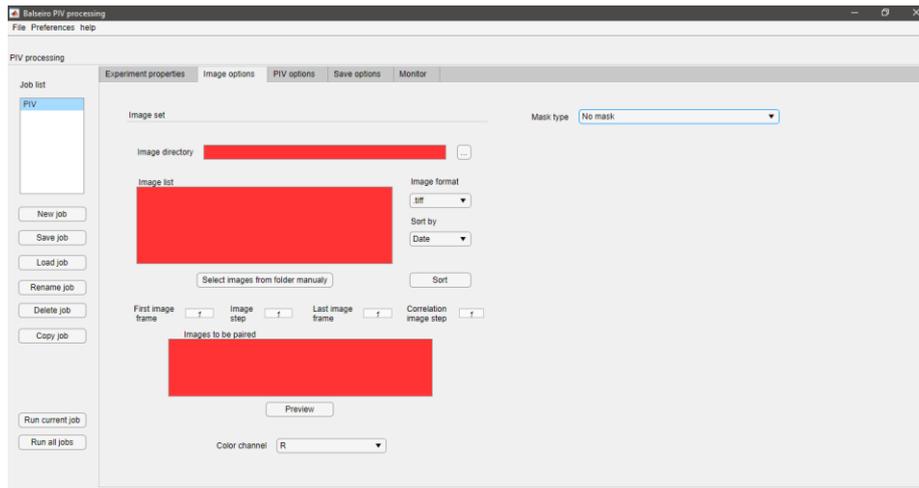


Figura 5 – Bariloche PIV Processing. Opciones de imagen.

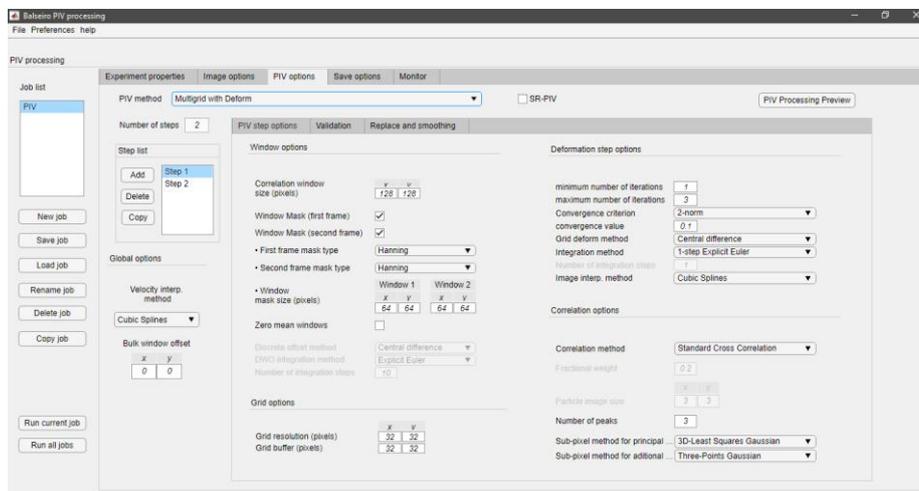


Figura 6 – Bariloche PIV Processing. Opciones de procesamiento PIV.

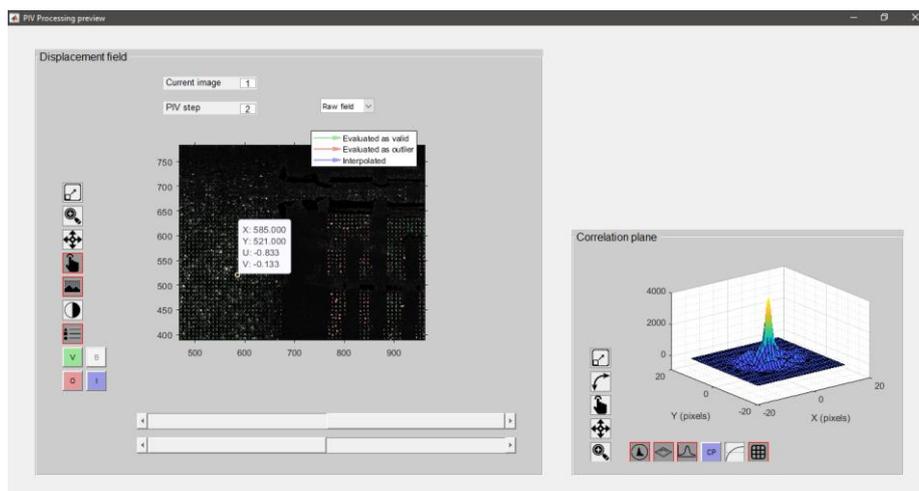


Figura 7 – Bariloche PIV Processing. Previsualización de resultados.

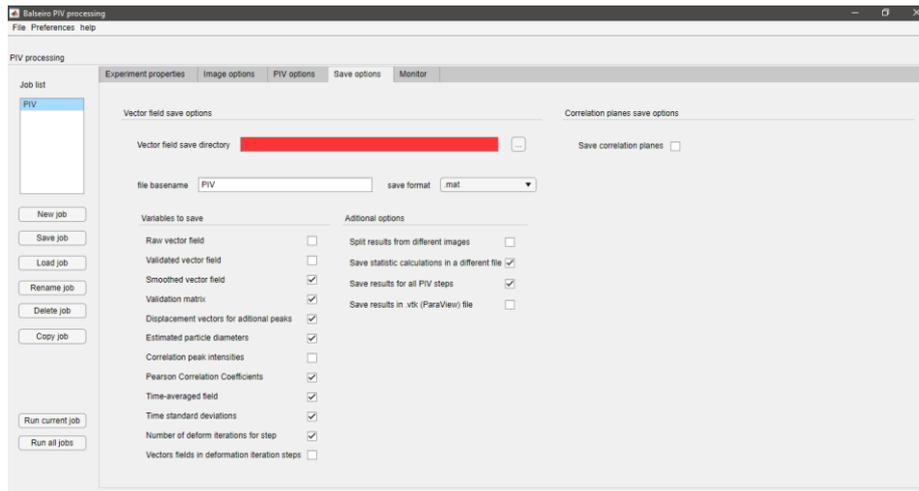


Figura 8 – Bariloche PIV Processing. Opciones de guardado.

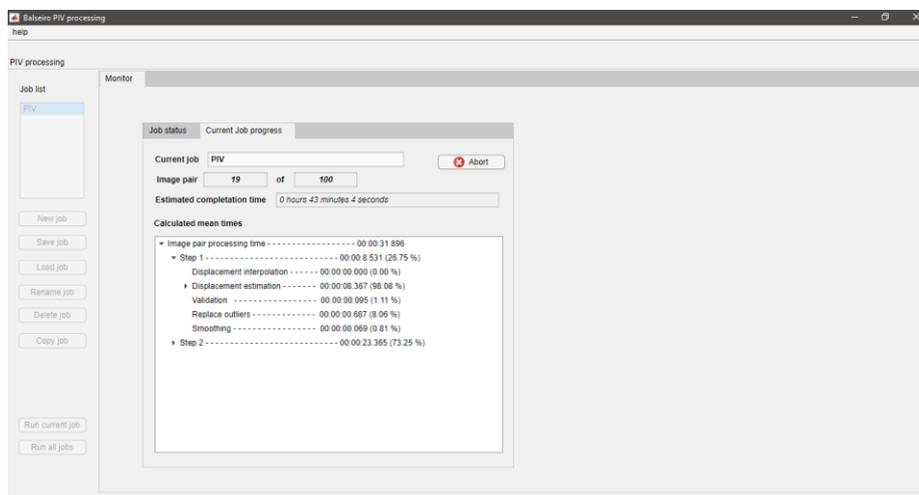


Figura 9 – Bariloche PIV Processing. Monitoreo de procesamiento.

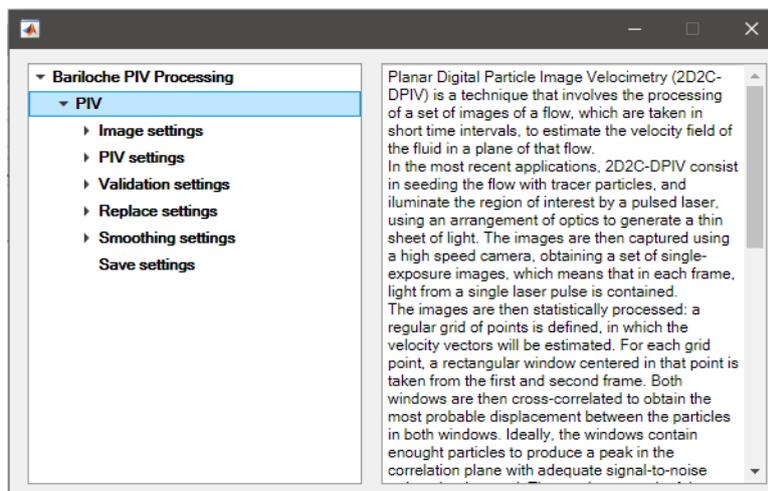


Figura 10 – Bariloche PIV Processing. guía de usuario.

El programa desarrollado tiene la ventaja frente a Prana de poseer herramientas de procesamiento más modernas, una interfaz más elegante y organizada, guía de usuario, y además es portable, permitiendo ser instalado y ejecutado en un ordenador de forma directa, aunque este no cuente con Matlab. Como desventaja se comprobó que es de ejecución mucho más lenta. Se planea en un futuro desarrollar el programa en un lenguaje de programación que sea de código abierto, y que permita su ejecución de una forma más eficiente.

V. ANÁLISIS DE ERRORES Y DESEMPEÑO

En esta sección se analizan, por medio de simulaciones numéricas, el desempeño de los distintos algoritmos y métodos de procesamiento descritos en las secciones anteriores, que conforman las distintas etapas del procesamiento en PIV. Los métodos de correlación se ponen a prueba mediante la evaluación de pares de ventanas generados de forma sintética, aplicando el método de Montecarlo. Las ventanas son generadas variando una serie de parámetros de forma aleatoria, como ser el diámetro, intensidad y densidad de las partículas, nivel de ruido y desplazamiento de las partículas. El rango de variación de estos parámetros se selecciona de forma de generar una amplia variedad de condiciones, buscando producir imágenes realistas, representativas de los escenarios que podrían encontrarse en situaciones reales, desde imágenes con buen contraste y alta densidad de partículas hasta imágenes con poco contraste y poca cantidad de partículas. Para cada par de ventanas se estima el desplazamiento de las imágenes de partículas, y se determina el error de la estimación teniendo en cuenta el desplazamiento verdadero. Realizando esto para un gran número de ventanas con distinta combinación de parámetros, es posible generar estadísticas para la distribución del error de las estimaciones en función de los distintos parámetros que se varían en la simulación. Una descripción detallada de la forma en la que se generaron las ventanas sintéticas se encuentra en el anexo IX.10.

En todas las simulaciones realizadas, la densidad media de partículas, ρ_p , el diámetro medio de las partículas dentro de la ventana, μ_{Dp} , el desvío estándar del diámetro de partícula dentro de la ventana, σ_{Dp} , la media del ruido, μ_R , el desvío estándar del ruido, σ_R , la relación señal-ruido, SR (definida como el cociente entre la intensidad media de las partículas dentro de la ventana, μ_I , y el nivel de ruido, σ_R) y el parámetro de variación de intensidad de partícula dentro de la ventana, $\bar{\sigma}_I$, son tomados aleatoriamente para cada par de ventanas, considerando los límites mostrados en la Tabla 1. Los restantes parámetros importantes como el desplazamiento verdadero y gradientes de desplazamiento, tamaños de ventana, métodos de correlación y estimación subpíxel, y cantidad de pares de ventanas evaluadas son aclarados en cada simulación particular. En la Figura 11 se muestran, a modo de ejemplo, algunas ventanas generadas sintéticamente bajo los parámetros usados para las simulaciones. El error de cada estimación fue calculado como la diferencia entre el desplazamiento medio verdadero aplicado sobre las partículas en la ventana y el desplazamiento estimado. El error sistemático fue calculado a partir de la media de los errores de estimación, mientras que el error aleatorio fue calculado como el desvío estándar de los errores de estimación.

Parámetro	Unidad de medida	Límite inferior	Límite superior
ρ_p	partículas/píxel	6/(32x32)	14/(32x32)
μ_{Dp}	píxel	2	4
σ_{Dp}	píxel	0	0,5
μ_R	profundidad de píxel	0	0
σ_R	Profundidad de píxel	1	5
SR	Adimensional	6	20
$\bar{\sigma}_I$	Profundidad de píxel	0	1/6

Tabla 1 – Límite superior e inferior para los parámetros utilizados en la generación de ventanas sintéticas en las simulaciones llevadas a cabo

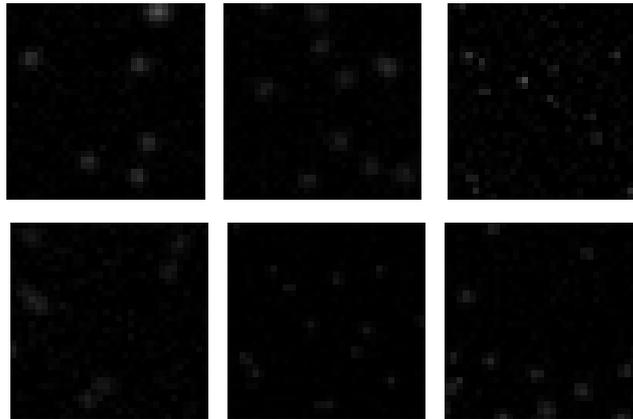


Figura 11 – Conjunto de ventanas aleatorias de 32x32 píxeles generadas sintéticamente según se detalla en esta sección, mostradas a modo de ejemplo.

V.1. COMPARACIÓN ENTRE LA CORRELACIÓN DIRECTA Y LA CORRELACIÓN ESTÁNDAR

En esta sección, se comparan los métodos de correlación directa y el método de correlación estándar basado en FFT. En la Figura 12 se compara el costo computacional de ambos métodos en función del tamaño de ventana efectivo, utilizando ventanas cuadradas aleatorias de igual tamaño ($Ni_1 = Ni_2 = Nj_1 = Nj_2$). Se procesaron un total de 100.000 pares de ventanas. El costo computacional fue calculado de forma normalizada como el cociente entre el tiempo medio de procesamiento para cada tamaño de ventana y un tiempo de referencia. Como tiempo de referencia se tomó el

tiempo medio de procesamiento de ventanas cuadradas de tamaño efectivo de 16x16 píxeles por el método de correlación estándar. En el caso del método de correlación estándar, las ventanas fueron rellenas con ceros de forma tal de que el tamaño de ventana resultante sea potencia de dos y que además garantice que el plano de correlación obtenido coincida con el obtenido por correlación directa. puede observarse que el costo computacional del método de correlación por transformada rápida de Fourier se vuelve significativamente menor para tamaños de ventana efectiva de más de 16 píxeles. Cabe destacar que los cálculos fueron llevados a cabo en Matlab, empleando una CPU de uso general de un núcleo. En otro tipo de herramientas de cálculo estos tiempos podrían verse afectados. Por otro lado, en la Figura 13 se comparan los planos de correlación obtenidos por transformada rápida sin relleno con ceros (equivalente a la correlación directa cíclica) y a la obtenida por transformada rápida con relleno (equivalente a la correlación directa). Se puede observar que ambos planos son ligeramente diferentes. Los efectos circulares de la correlación cíclica se pueden interpretar como ruido aditivo sobre el plano de correlación.

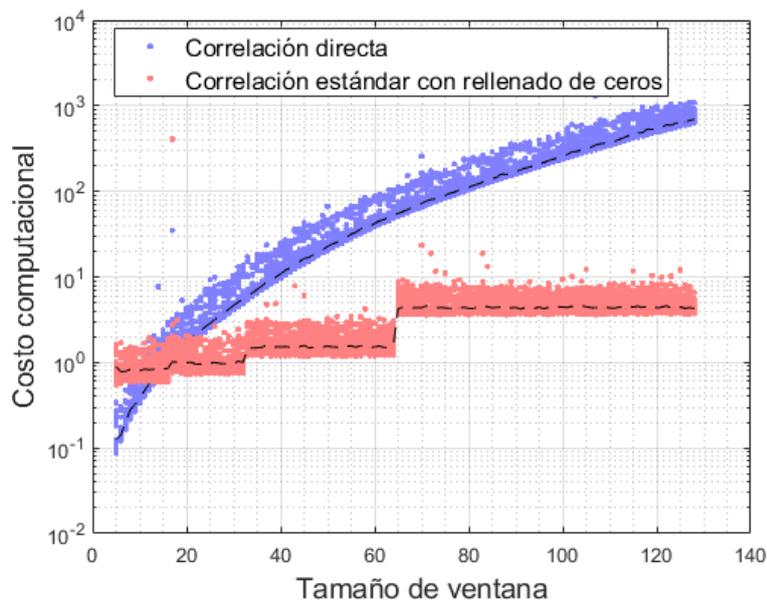


Figura 12 - Comparación del costo computacional en función del tamaño de ventana (en píxeles), para la correlación directa vs la correlación estándar con relleno de ceros. El costo

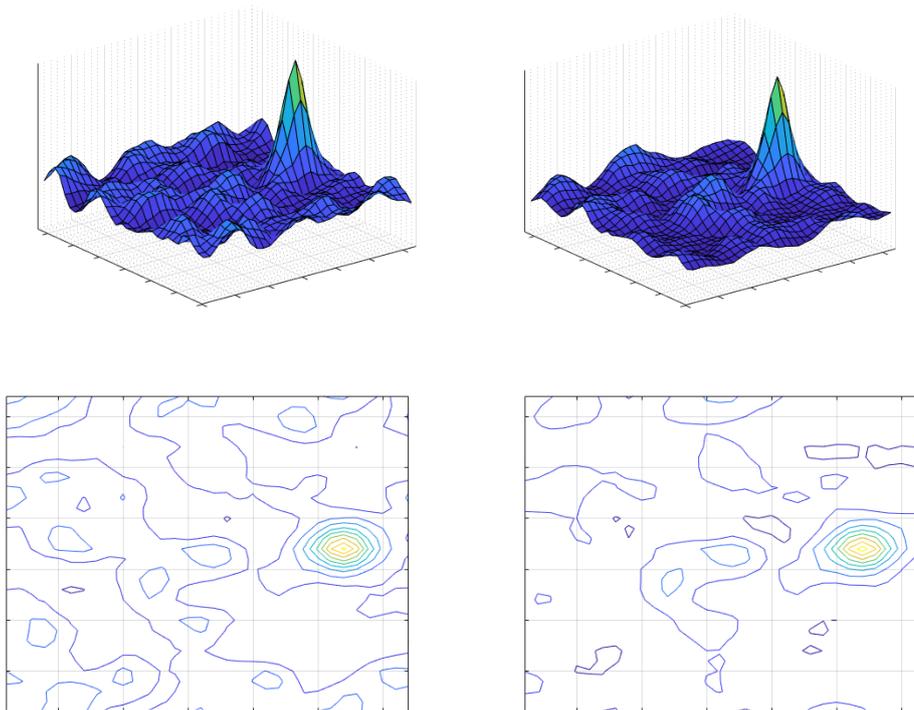


Figura 13 – Comparación entre el plano de correlación obtenido por correlación estándar sin aplicar rellenado de ceros (izquierda) vs el obtenido aplicando rellenado (derecha). Arriba se muestran las funciones de correlación como superficies, mientras que abajo se muestran las curvas de nivel.

V.2. EFECTO DEL TAMAÑO DE VENTANA EN EL ERROR ALEATORIO

La determinación del desplazamiento por medio de la técnica basada en correlación cruzada consiste en utilizar el pico de máxima correlación de la función de correlación cruzada, y un método de estimación subpíxel, para establecer el desplazamiento medio más probable entre las imágenes de partículas contenidas en las ventanas de interrogación. La precisión de este método depende de la amplitud de la señal de correlación y el nivel de ruido presente en dicha señal. Diversos factores deterioran la señal de correlación, entre los que se hallan la pérdida de pares de partículas por desplazamientos fuera de la ventana, la variación de intensidad de las partículas entre ventanas debido a los movimientos fuera de plano y la iluminación no homogénea del haz de luz, y los gradientes de desplazamiento dentro de la ventana. La intensidad del pico de máxima correlación se incrementa con la cantidad de pares de partículas emparejadas entre las ventanas. Si el desplazamiento de las imágenes de partículas dentro de la ventana es uniforme, entonces el incremento del tamaño de ventana implica un incremento en la cantidad de partículas emparejadas, y como consecuencia, un aumento en la intensidad del pico de máxima correlación y en la relación señal-ruido de la función de correlación cruzada. Esto último produce una disminución en el error aleatorio de la estimación.

La Figura 14 muestra los resultados obtenidos a partir de un ensayo de Montecarlo, para el error aleatorio resultante en función del tamaño de ventana (longitud del lado, en píxeles), al aplicar el método de estimación de desplazamiento basado en la correlación cruzada, ajuste gaussiano de tres puntos, y la aplicación de distintas funciones de ventaneo, sobre ventanas generadas aleatoriamente tal como se describe en el anexo IX.10. Se tuvieron en cuenta los límites de la Tabla 1 para la elección aleatoria de estos parámetros. En estas simulaciones no se han incluido efectos de gradiente de desplazamiento, por lo que todos los centros de las partículas se desplazan por igual. El desplazamiento horizontal, u_m , y vertical, v_m , para las partículas de cada par de ventanas, fueron seleccionados aleatoriamente en el rango $[-\frac{1}{2}, \frac{1}{2}]$. Se utilizaron pares de ventanas cuadradas de igual tamaño, con tamaños comprendidos aleatoriamente entre 16 y 128 píxeles. La simulación para cada función de ventaneo comprendió la evaluación de 600.000 pares de ventanas.

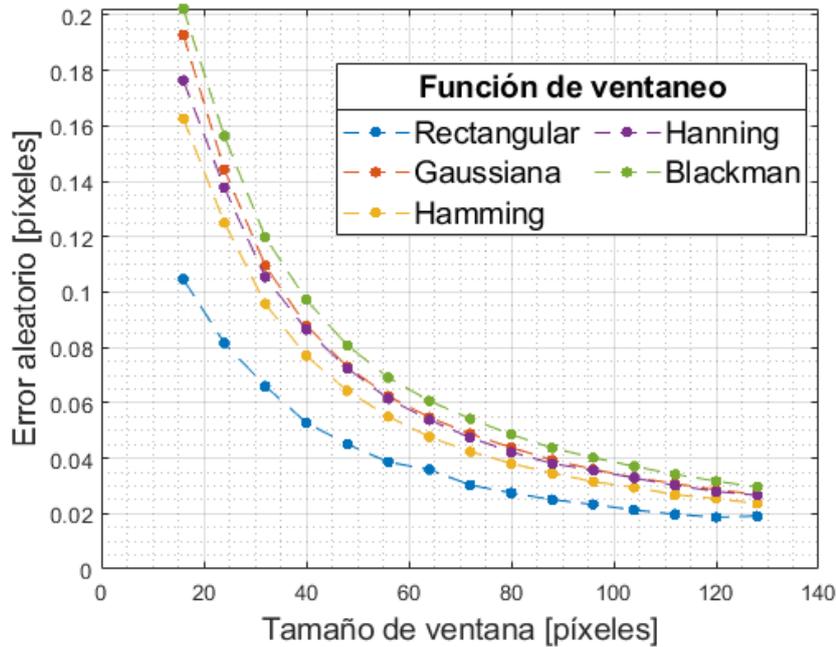


Figura 14 – Comparación del error aleatorio en la estimación del desplazamiento de partículas con desplazamiento uniforme, en función del tamaño de ventana, aplicando distintas funciones de ventaneo.

Puede observarse la tendencia en la disminución del error aleatorio con el incremento del tamaño de ventana. En el caso del empleo de funciones de ventaneo, la cantidad aparente de partículas dentro de la ventana disminuye, debido a la disminución de la intensidad de la señal hacia los bordes. Esto genera un tamaño de ventana efectivo inferior al tamaño real de la ventana, que se manifiesta en un mayor error aleatorio comparado con el obtenido con ventanas rectangulares para un mismo tamaño de ventana. Sin embargo, el empleo de funciones de ventaneo permite reducir tanto la

magnitud del error sistemático como su dependencia con el desplazamiento (sección V.4), lo que a su vez permite reducir los efectos de enclavamiento de picos o “peak locking” en inglés (sección V.6).

V.3. EFECTOS DEL GRADIENTE DE DESPLAZAMIENTO DENTRO DE LAS VENTANAS

Como se demostró en la sección anterior, el uso de ventanas de mayor tamaño cuando las partículas tienen un desplazamiento uniforme genera una disminución del error aleatorio debido al incremento de partículas emparejadas y el consecuente aumento de la relación señal-ruido de la función de correlación cruzada. No obstante, en situaciones donde se presentan gradientes de desplazamiento dentro de las ventanas de interrogación, el nivel de emparejamiento entre centros de partículas disminuye en función de este gradiente. En la Figura 15 se muestran los resultados obtenidos por simulaciones de Montecarlo para el error aleatorio en función del tamaño de ventana, para distintos valores de gradiente de desplazamiento horizontal con respecto a la componente horizontal, du/dx , en donde se utilizaron ventanas cuadradas con tamaños comprendidos aleatoriamente entre 16 y 128 píxeles, correlación estándar y ajuste gaussiano de tres puntos. En estas simulaciones, los restantes parámetros de gradientes se fijaron en 0. El desplazamiento horizontal, u_m , y vertical, v_m , para las partículas de cada par de ventanas, fueron seleccionados aleatoriamente en el rango $[-\frac{1}{2}, \frac{1}{2}]$. Los restantes parámetros para la generación de cada par de ventanas fueron seleccionados aleatoriamente según el rango de valores establecido por los límites de la Tabla 1. La simulación para cada valor de du/dx comprendió la evaluación de 500.000 pares de ventanas.

Se manifiesta una disminución del error aleatorio en función del tamaño de ventana para tamaños de ventana bajos, debido al incremento de la cantidad de partículas emparejadas. Sin embargo, por encima de cierto tamaño de ventana, el error aleatorio comienza a aumentar, debido a que las partículas adicionales no son coincidentes en sus centros por efecto del gradiente, y contribuyen a aumentar el nivel de ruido del plano de correlación, en lugar de incrementar la amplitud del pico de máxima correlación. Existe entonces un tamaño de ventana óptimo que depende del valor del gradiente.

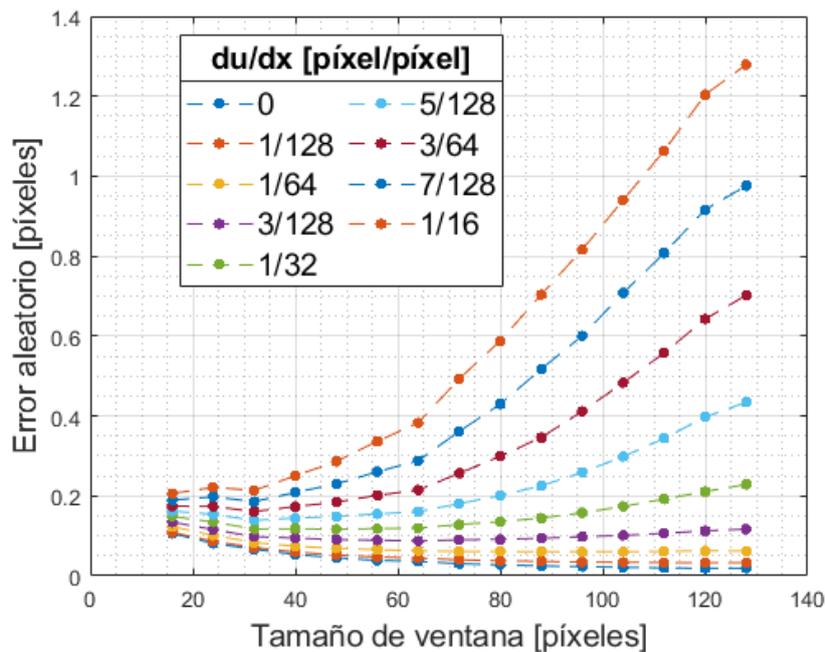


Figura 15 – Comparación del error aleatorio en la estimación del desplazamiento en función del tamaño de ventana, para distintos valores de gradiente de desplazamiento horizontal, aplicando correlación estándar y ajuste gaussiano de tres puntos.

V.4. EFECTOS DEL RELLENADO CON CEROS Y EL VENTANEO EN EL ERROR SISTEMÁTICO Y ALEATORIO

Se llevaron a cabo una serie de ensayos mediante el método de Montecarlo sobre ventanas de interrogación generadas de forma sintética, para determinar los efectos del relleno con ceros, el uso de funciones de ventaneo, y el uso de ventanas de distinto tamaño, en el error sistemático y el error aleatorio de los resultados, utilizando el método de correlación estándar y ajuste gaussiano de tres puntos. Los métodos estudiados incluyen el uso de pares de ventanas cuadradas de 32x32 píxeles, el uso de ventanas de 32x32 píxeles y relleno con ceros con y sin normalización, el uso de funciones de ventaneo tipo Hamming con pares de ventanas de 32x32 píxeles y 50x50 píxeles, y el uso de pares de ventanas de distinto tamaño, con 32x32 píxeles para la primera ventana y 64x64 píxeles para la segunda. No se aplicaron gradientes de desplazamiento dentro de las ventanas, y los restantes parámetros se seleccionaron aleatoriamente para cada par de ventanas según los límites de la Tabla 1. La Figura 16 muestra los resultados obtenidos para los errores sistemático y aleatorio en función del desplazamiento horizontal de las imágenes de partículas, a partir del procesamiento de 500.000 pares de ventanas con desplazamientos de partículas en el rango [0, 20] mediante distintos métodos mencionados, mientras que la Figura 17 muestra los resultados obtenidos del procesamiento de otros 500.000 pares de ventanas con desplazamientos de partículas en el rango [0, 1], utilizando los mismos esquemas.

A partir de la Figura 16 y la Figura 17, puede observarse que el uso de ventanas de igual tamaño sin aplicar rellenado de ceros (equivalente a efectuar la correlación cíclica) presenta un error sistemático (debido al efecto de pérdida de correlación) y un error aleatorio (debido a las distintas fuentes de ruido, entre las que se encuentra el efecto periódico de la correlación cíclica o alias). Estos errores se incrementan con el aumento del desplazamiento medio de las partículas (método 1 en las figuras). Debido a la ambivalencia que genera la condición de periodicidad, el desplazamiento máximo medible es igual a la mitad del tamaño de la ventana, que en este caso es de 16 píxeles. También puede observarse un comportamiento oscilatorio en la distribución de los errores: estos errores son menores para desplazamientos enteros, y se incrementan para desplazamientos fraccionales. Este comportamiento oscilatorio es la fuente del error conocido como enclavamiento de picos (sección V.6).

Si se emplea rellenado con ceros (método 2 en las figuras), se eliminan los efectos periódicos en la correlación, y la correlación obtenida por el método de Fourier se vuelve equivalente a la obtenida de forma directa. De esta forma, el error aleatorio disminuye para desplazamientos mayores a 1 píxel. También se incrementa el rango de desplazamientos medible. Sin embargo, el sesgo permanece inalterado, debido a los efectos de pérdida de correlación. Los efectos de pérdida de correlación pueden reducirse si se aplica una normalización (Ec. II.6-2, método 3 en las figuras), dividiendo la función de correlación cruzada por la correlación cruzada de las funciones de ventaneo (en este caso, rectangulares).

Otra forma de reducir los efectos de pérdida de correlación es emplear el método sugerido por Gui (2000), que implica utilizar una función de ventaneo sobre las ventanas de interrogación. Se observa que, empleando este método sobre ventanas de igual tamaño (método 4), se obtiene una disminución apreciable en el error sistemático, sobre todo para desplazamientos de menos de 1 píxel. Esto último es importante ya que en los métodos de pasos múltiples (sección II.11), el desplazamiento a estimar en el segundo y posteriores pasos de la iteración PIV se encuentra comprendido en el rango $[-\frac{1}{2}, \frac{1}{2}]$, lo que implica que la técnica sugerida por Gui permite obtener estimaciones con muy poco sesgo a partir del segundo paso de iteración. No obstante, este método genera un aumento en el error aleatorio, debido a que disminuye el tamaño de ventana efectivo, al reducir la contribución que tienen las imágenes de partículas más alejadas del centro de las ventanas sobre la función de correlación cruzada. Esto puede ser compensado utilizando ventanas más grandes (método 5), logrando reducir tanto el error sistemático como el aleatorio.

Finalmente, el error de sesgo puede eliminarse empleando ventanas de distinto tamaño (método 6). Esto es debido a que, para cierto rango de desplazamientos, una ventana se encuentra totalmente contenida sobre la otra, por lo que el efecto de pérdida de correlación no se manifiesta. A partir de cierto valor de desplazamiento, una ventana deja de estar contenida en la otra, y el sesgo vuelve a tener lugar. Esto se puede ver en la Figura 16, a partir del salto en el error sistemático producido en torno a los 16 píxeles.

Como conclusión, el empleo de rellenado con ceros y normalización, aplicación de funciones de ventaneo y uso de ventanas de distinto tamaño son técnicas efectivas para reducir los errores sistemáticos y aleatorios en la determinación de los desplazamientos.

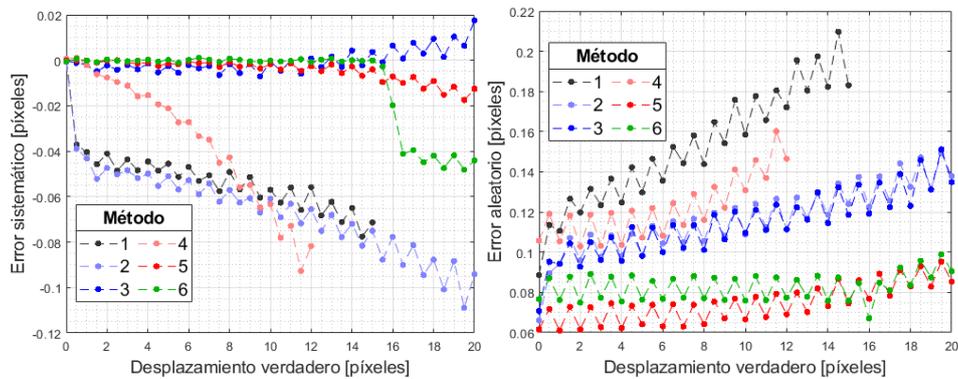


Figura 16 – Comparación del error sistemático y aleatorio de distintos enfoques de ventaneo. Método 1: ventanas de interrogación cuadradas de 32x32 píxeles; método 2: ventanas de 32x32 píx. con rellenado de ceros; método 3: ventanas de 32x32 píx. con rellenado de ceros y normalización; método 4: ventanas de interrogación cuadradas de 32x32 píx., con aplicación de función de ventaneo (Hamming); método 5: ventanas de interrogación cuadradas de 50x50 píx., con aplicación de función de ventaneo (Hamming); método 6: primera ventana de 32x32 píx. y segunda ventana de 64x64 píx.

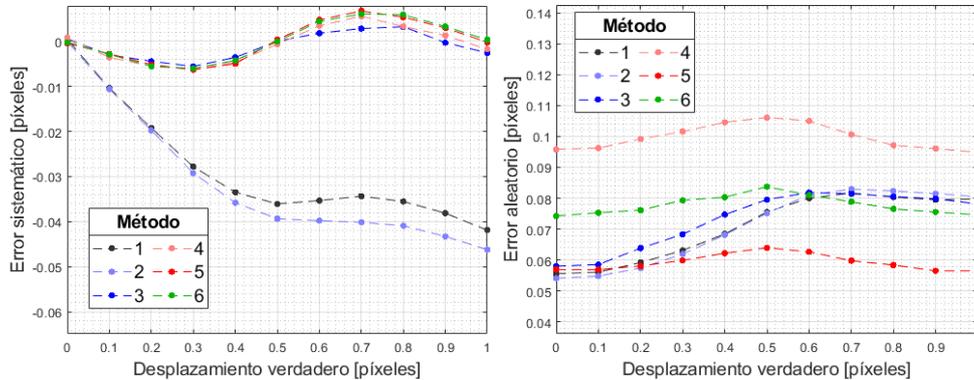


Figura 17 – Comparación del error sistemático y aleatorio de distintos enfoques de ventaneo, para desplazamientos en el rango de 0 a 1 píxel. Método 1: ventanas de interrogación cuadradas de 32x32 píxeles; método 2: ventanas de 32x32 píx. con rellenado de ceros; método 3: ventanas de 32x32 píx. con rellenado de ceros y normalización; método 4: ventanas de interrogación cuadradas de 32x32 píx., con aplicación de función de ventaneo (Hamming); método 5: ventanas de interrogación cuadradas de 50x50 píx., con aplicación de función de ventaneo (Hamming); método 6: primera ventana de 32x32 píx. y segunda ventana de 64x64 píx.

V.5. COMPARACIÓN ENTRE LA CORRELACIÓN CRUZADA ESTÁNDAR, LA CORRELACIÓN NORMALIZADA Y EL MÉTODO DE MÍNIMOS CUADRADOS

En esta sección se compara el desempeño de los métodos de correlación cruzada estándar, correlación cruzada normalizada, y mínimos cuadrados, abordados en la sección II.6, para la estimación del desplazamiento de las imágenes de partículas dentro de las ventanas de interrogación. En la Figura 18 se muestran las gráficas de las funciones obtenidas por cada uno de estos métodos. La función de correlación normalizada produce un plano de correlación similar a la correlación estándar, pero con valores de correlación acotados en el intervalo $[-1, 1]$. En el método de mínimos cuadrados, por otro lado, el desplazamiento es estimado a partir del mínimo de la función de error. En este caso, si se aplica un método de estimación subpíxel basado en el ajuste a una superficie gaussiana, el pico correspondiente al mínimo de la función de error puede invertirse para realizar el ajuste.

En la Figura 19 y la Figura 20 se muestran los resultados obtenidos por medio de ensayos de Montecarlo para los errores sistemáticos y aleatorios en función del desplazamiento horizontal, utilizando los tres métodos de correlación mencionados. Se utilizaron ventanas rectangulares de distinto tamaño (32x32 píxeles para la primera ventana, 64x64 píxeles para la segunda ventana), y ajuste gaussiano de tres puntos para la estimación subpíxel. El desplazamiento de las partículas dentro de cada par de ventanas se aplicó uniformemente, sin considerar gradientes de desplazamiento dentro de la ventana. Para cada par de ventanas, el desplazamiento de las imágenes de partículas se seleccionó aleatoriamente dentro de los intervalos mostrados en las figuras. Los restantes parámetros se seleccionaron aleatoriamente dentro de los límites mostrados en la Tabla 1. Cada caso comprendió la evaluación de 500.000 pares de ventanas sintéticas.

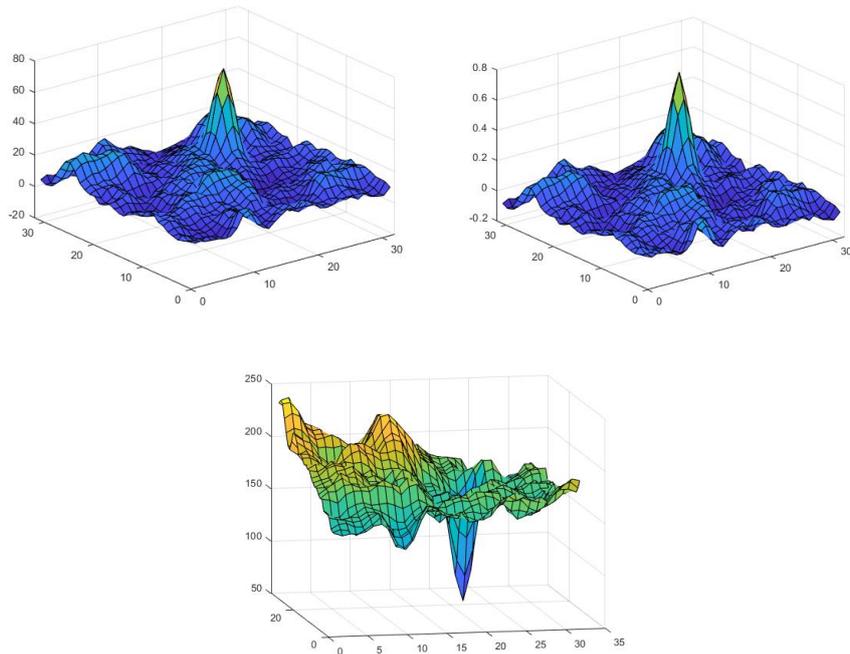


Figura 18 – Funciones de correlación obtenidas mediante distintos métodos, para el mismo par de ventanas. Arriba a la izquierda, plano de correlación para la correlación cruzada estándar; arriba a la derecha, plano de correlación para la correlación cruzada normalizada; abajo, función de error para el método de mínimos cuadrados.

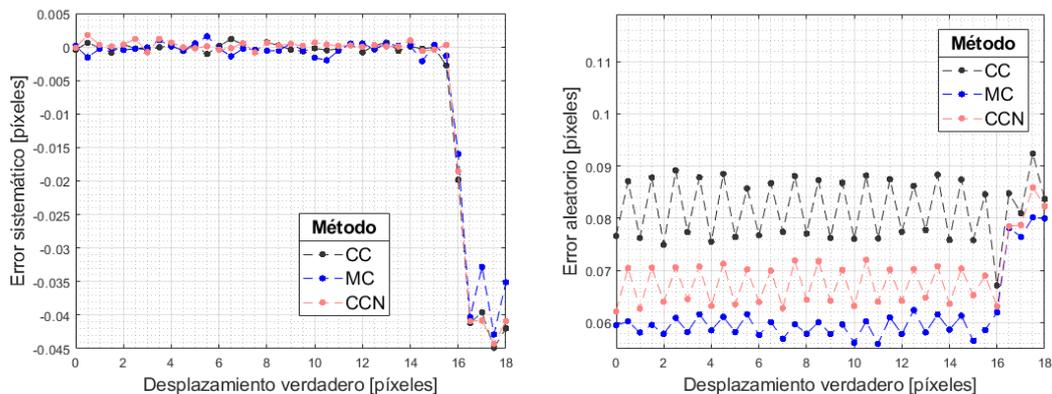


Figura 19 – Comparación del error sistemático y aleatorio en función del desplazamiento, en el rango de desplazamientos $[0,18]$, para los métodos de correlación cruzada (CC), mínimos cuadrados (MC) y correlación cruzada normalizada (CCN), utilizando la primera ventana de 32×32 píxeles, la segunda de 64×64 píxeles, y ajuste gaussiano de tres puntos para la estimación subpíxel.

Al analizar las figuras, se observa una disminución en los errores sistemático y aleatorio al emplear los métodos de mínimos cuadrados y correlación cruzada normalizada, con respecto al método de correlación cruzada. También se observa una menor variabilidad entre los errores para desplazamientos enteros y fraccionales, resultando en un menor efecto de enclavamiento de picos (sección V.6).

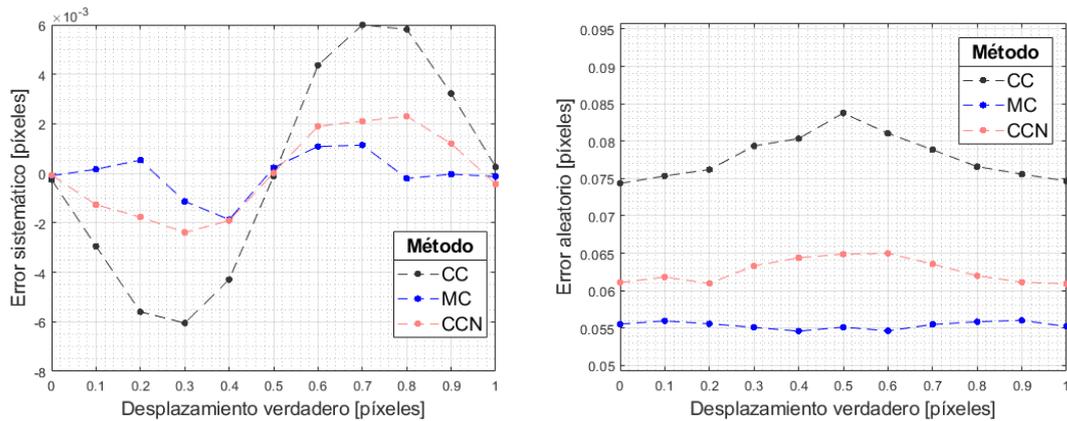


Figura 20 – Comparación del error sistemático y aleatorio en función del desplazamiento, en el rango de desplazamientos $[0,1]$, para los métodos de correlación cruzada (CC), mínimos cuadrados (MC) y correlación cruzada normalizada (CCN), utilizando la primera ventana de 32×32 píxeles, la segunda de 64×64 píxeles, y ajuste gaussiano de tres puntos para la estimación subpíxel.

V.6. ERROR DEBIDO AL EFECTO DE ENCLAVAMIENTO DE PICOS

El efecto de enclavamiento de picos es un fenómeno ampliamente conocido, citado e investigado en la bibliografía (Gui & Wereley, 2002; Angele & Muhammad-Klingmann, 2005; Nogueira, Legrand, Jiménez, & De Gregorio, 2021), que se manifiesta como un sesgo en la estimación del desplazamiento hacia valores de desplazamiento enteros. Además de afectar las mediciones instantáneas, este fenómeno también impacta sobre el cálculo de medidas estadísticas como la velocidad media local, y el desvío estándar local de la velocidad (asociado a la energía cinética turbulenta), por lo que es deseable reducirlo.

A partir de un análisis estadístico, es posible determinar que la principal fuente de enclavamiento de picos es la distribución no uniforme del error sistemático con respecto al desplazamiento verdadero. La variación de la amplitud del error aleatorio con respecto al desplazamiento verdadero también es una fuente de enclavamiento de picos. A continuación, se presenta un análisis similar al propuesto por Gui & Wereley (2002).

Sea U una variable aleatoria que representa valores de una de las componentes de desplazamiento, muestreados a partir de un campo dado, y \tilde{U} una variable aleatoria que representa las estimaciones del componente de desplazamiento realizadas sobre dicho campo, a partir del método de correlación cruzada. Sea f_U la función de densidad de probabilidad de U , $f_{\tilde{U}}$ la función de densidad de probabilidad de \tilde{U} , y $f_{U,\tilde{U}}$ la función de densidad de probabilidad conjunta de U y \tilde{U} . La función de densidad de probabilidad de \tilde{U} puede obtenerse a partir de la función de densidad de probabilidad conjunta como

$$f_{\tilde{U}}(\tilde{u}) = \int_{-\infty}^{\infty} f_{U,\tilde{U}}(u, \tilde{u}) du.$$

Por otro lado, la función de densidad de $f_{U,\tilde{U}}$ puede escribirse como

$$f_{U,\tilde{U}}(u, \tilde{u}) = f_{\tilde{U}|U}(u, \tilde{u}) f_U(u),$$

donde $f_{\tilde{U}|U}$ es la función de densidad de probabilidad condicional de \tilde{U} dado U . De esta forma, resulta

$$f_{\tilde{U}}(\tilde{u}) = \int_{-\infty}^{\infty} f_{\tilde{U}|U}(u, \tilde{u}) f_U(u) du.$$

En base a esta expresión, se observa que para que U y \tilde{U} tengan la misma función de densidad de probabilidad, debe tenerse necesariamente $f_{\tilde{U}|U}(u, \tilde{u}) = \delta(\tilde{u} - u)$, siendo δ la función delta de Dirac. Esto equivale a tener un estimador perfecto para \tilde{U} , de forma de que reproduzca exactamente el valor de U , lo cual no es posible en la práctica. Para cualquier otra forma de la función $f_{\tilde{U}|U}$, se genera una distorsión de la distribución de probabilidades del desplazamiento.

En forma general, la función $f_{\tilde{U}|U}$ depende de $\tilde{u} - u$ y de una serie de parámetros $\boldsymbol{\theta}(u) = [\mu(u), \sigma(u), \gamma_1(u), \gamma_2(u), \dots]^T$, donde μ es un parámetro de centro, σ es un parámetro de escala, y $\gamma_1, \gamma_2, \dots$ son parámetros de forma. Aplicando esto, se tiene

$$f_{\tilde{U}}(\tilde{u}) = \int_{-\infty}^{\infty} f_{\tilde{U}|U}(\tilde{u} - u, \boldsymbol{\theta}(u)) f_U(u) du.$$

A partir de esta expresión, se ve que si $\boldsymbol{\theta}$ no es función de u , entonces la función de densidad de probabilidad $f_{\tilde{U}}$ se obtiene como la convolución de f_U con $f_{\tilde{U}|U}$. Por otro lado, si $\boldsymbol{\theta}$ es función de u , entonces la distribución de $f_{\tilde{U}}$ se verá distorsionada de forma irregular. Por ejemplo, si $f_{\tilde{U}|U}$ posee menor dispersión para valores enteros de u , y la dispersión aumenta para valores fraccionales, la distribución de $f_{\tilde{U}}$ estará más concentrada en torno a los valores enteros, en contraste con f_U . A este efecto se lo denomina enclavamiento de picos.

Al evaluar las variaciones en el error aleatorio y sistemático de las estimaciones de desplazamiento en función del desplazamiento verdadero de la Figura 16 y de la Figura 17, puede inferirse qué esquemas de correlación producirán un mayor efecto de enclavamiento de picos. Se llevaron a cabo simulaciones de Montecarlo sobre ventanas generadas de forma sintética para evidenciar los efectos de enclavamiento de picos. Las ventanas fueron generadas según se detalla en el anexo IX.10. Los desplazamientos verdaderos fueron tomados con una distribución uniforme en el rango de 0 a 3, y de 10 a 13. No se consideraron efectos de gradiente de desplazamiento. Los restantes parámetros implicados en la generación de las ventanas sintéticas se seleccionaron aleatoriamente según los límites mostrados en la Tabla 1. Cada histograma mostrado en

esta sección se basa en el procesamiento de 500.000 pares de ventanas. La correlación cruzada estándar produce un efecto de enclavamiento de picos apreciable en el rango de desplazamientos bajos, y se magnifica para desplazamientos altos, tal como se aprecia en la Figura 21. Cuando se emplea rellenado con ceros y normalización, los efectos de enclavamiento de picos se atenúan, y se vuelven menos sensibles a la magnitud del desplazamiento (Figura 22). Utilizar funciones de ventaneo no produce mejoras significativas en cuanto a la reducción del enclavamiento de picos, tal como se muestra en la Figura 23. Sin embargo, en la Figura 24 se aprecia que si la aplicación de la función de ventaneo se combina con un incremento en el tamaño de las ventanas se logra una reducción del efecto. Finalmente se contempla el uso de un mayor tamaño para la segunda ventana y distintos métodos de correlación (Figura 25). El uso de ventanas de distinto tamaño en conjunto con correlación cruzada estándar produce un efecto de enclavamiento de picos apreciable. Sin embargo, al utilizar el esquema de mínimos cuadrados o el de correlación cruzada normalizada, se logra una reducción significativa del efecto.

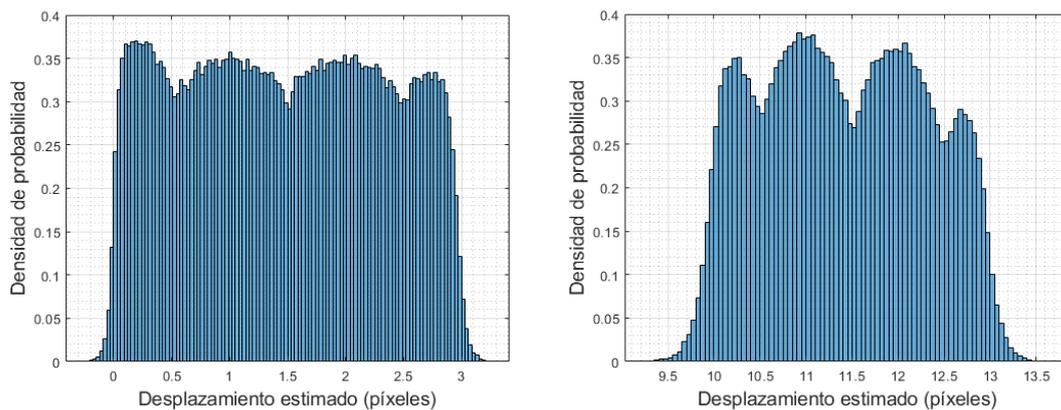


Figura 21 – Histograma para el desplazamiento estimado utilizando correlación cruzada estándar sobre ventanas de 32x32 píxeles, y ajuste gaussiano de tres puntos para la estimación subpíxel. A la izquierda, distribución de valores estimados cuando los desplazamientos verdaderos están uniformemente distribuidos entre 0 y 3 píxeles. A la derecha, distribución de valores estimados cuando los desplazamientos verdaderos están uniformemente distribuidos entre 10 y 13 píxeles.

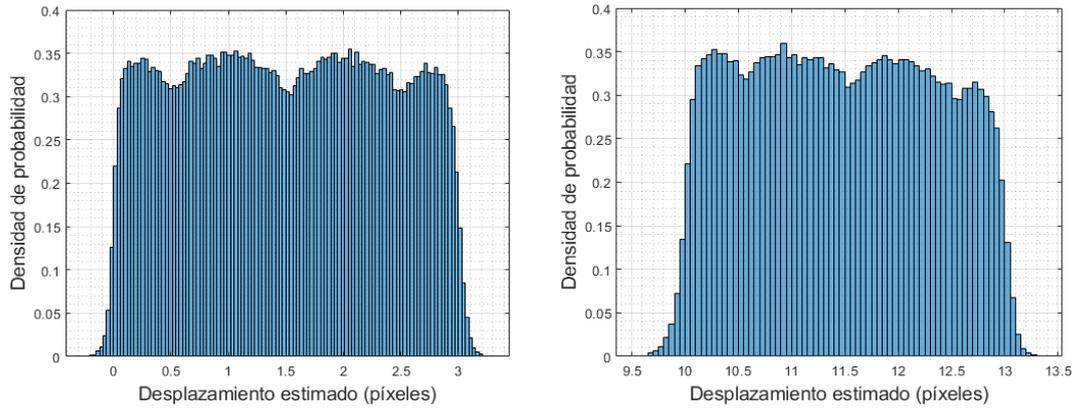


Figura 22 – Histograma para el desplazamiento estimado utilizando correlación cruzada estándar, rellenado con ceros y normalización, sobre ventanas de 32x32 píxeles, y ajuste gaussiano de tres puntos para la estimación subpíxel. A la izquierda, distribución de valores estimados cuando los desplazamientos verdaderos están uniformemente distribuidos entre 0 y 3 píxeles. A la derecha, distribución de valores estimados cuando los desplazamientos verdaderos están uniformemente distribuidos entre 10 y 13 píxeles.

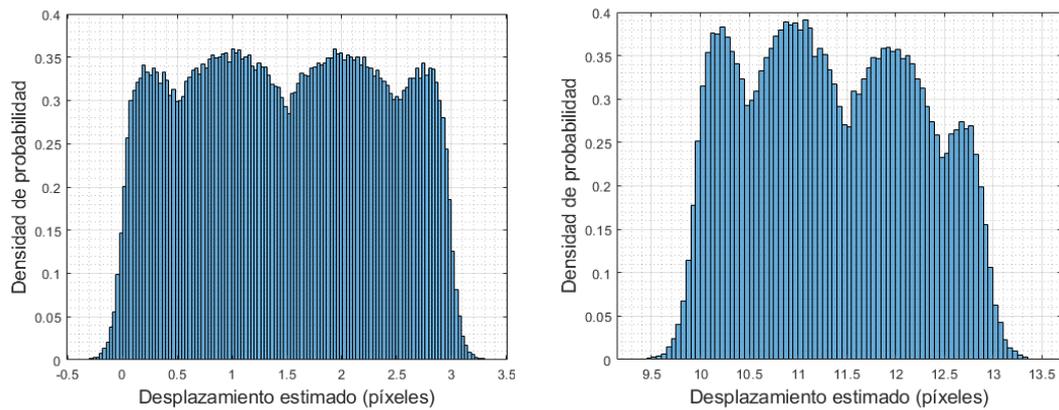


Figura 23 – Histograma para el desplazamiento estimado utilizando correlación cruzada estándar y aplicación de función de ventaneo (Hamming), sobre ventanas de 32x32 píxeles, y ajuste gaussiano de tres puntos para la estimación subpíxel. A la izquierda, distribución de valores estimados cuando los desplazamientos verdaderos están uniformemente distribuidos entre 0 y 3 píxeles. A la derecha, distribución de valores estimados cuando los desplazamientos verdaderos están uniformemente distribuidos entre 10 y 13 píxeles.

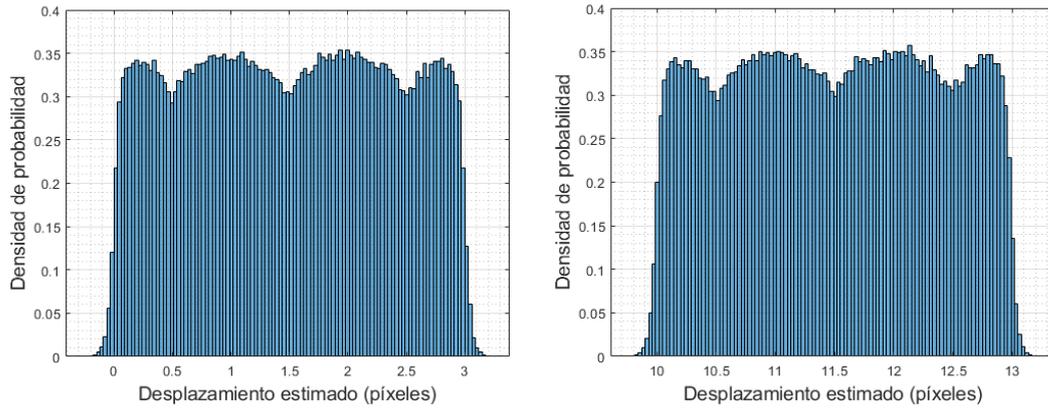


Figura 24 – Histograma para el desplazamiento estimado utilizando correlación cruzada estándar y aplicación de función de ventaneo (Hamming), sobre ventanas de 50x50 píxeles, y ajuste gaussiano de tres puntos para la estimación subpíxel. A la izquierda, distribución de valores estimados cuando los desplazamientos verdaderos están uniformemente distribuidos entre 0 y 3 píxeles. A la derecha, distribución de valores estimados cuando los desplazamientos verdaderos están uniformemente distribuidos entre 10 y 13 píxeles.

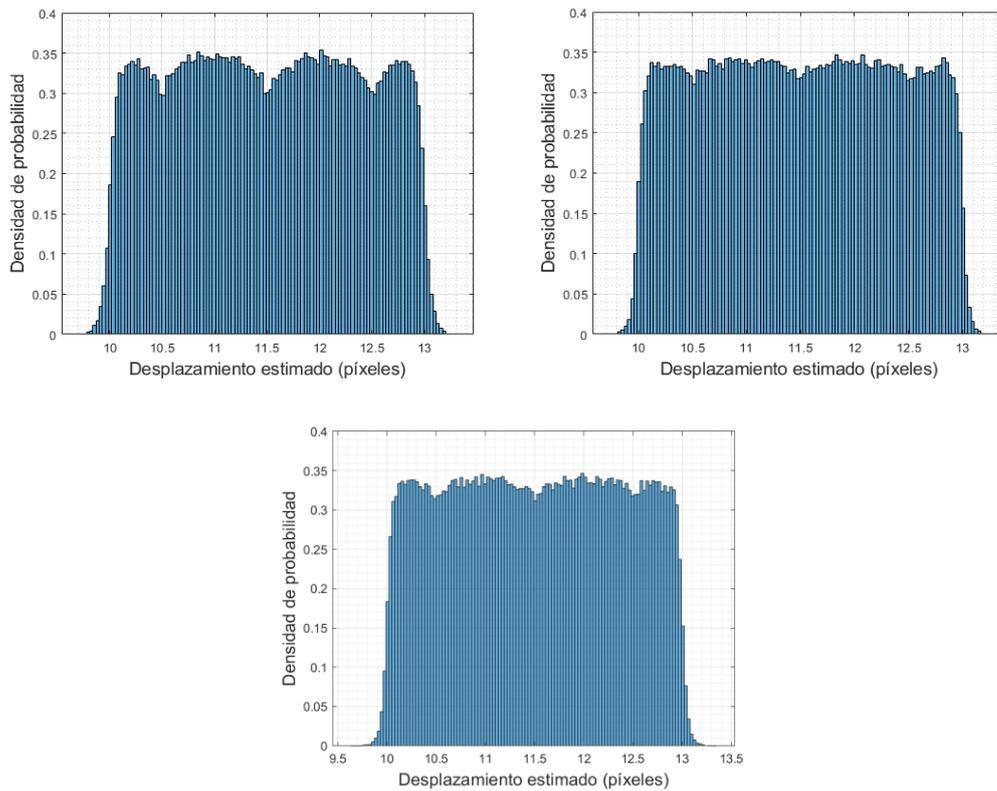


Figura 25 – Histograma para el desplazamiento estimado utilizando distintos esquemas de correlación, sobre ventanas de 32x32 píxeles (primera ventana) y 64x64 píxeles (segunda ventana), y ajuste gaussiano de tres puntos para la estimación subpíxel. Los desplazamientos verdaderos tienen distribución uniforme entre 10 y 13 píxeles. Arriba a la izquierda, correlación cruzada estándar. Arriba a la derecha, correlación cruzada normalizada. Abajo, mínimos cuadrados.

**V.7. DIÁMETRO ÓPTIMO DE PARTÍCULAS CON DISTRIBUCIÓN GAUSSIANA PARA LA ESTIMACIÓN
BASADA EN EL AJUSTE GAUSSIANO DE TRES PUNTOS**

En esta sección se presenta un desarrollo analítico con el objetivo de determinar un factor de escala óptimo para las partículas, el cual se relaciona directamente con su diámetro, para minimizar la incertidumbre en la estimación del desplazamiento por el método de ajuste gaussiano de tres puntos. Los resultados analíticos son entonces contrastados con simulaciones numéricas. Suponiendo que las imágenes de las partículas tienen una distribución de intensidades gaussiana con factor de escala σ_p , entonces el pico de correlación también tendrá una distribución gaussiana con factor de escala $\sigma = \sqrt{2}\sigma_p$ (anexo IX.6).

Sea $y(x)$ el valor del pico máximo de correlación, que ocurre en la coordenada horizontal x , y sean $y(x - 1)$ y $y(x + 1)$ los valores del plano de correlación en los puntos vecinos por izquierda y por derecha, respectivamente. El estimador para el desplazamiento horizontal según el método del ajuste gaussiano de tres puntos viene dado por (sección II.7.4)

$$\check{\delta}_x = x + \frac{1}{2} \left(\frac{\text{Ln}[y(x - 1)] - \text{Ln}[y(x + 1)]}{\text{Ln}[y(x - 1)] - 2 \text{Ln}[y(x)] + \text{Ln}[y(x + 1)]} \right).$$

El primer término de esta expresión representa la parte entera del desplazamiento, mientras que el segundo término se corresponde con la estimación de la parte fraccional del desplazamiento, por lo que no se pierde generalidad si se hace $x = 0$. En dicho caso, se estaría estimando la parte fraccional del desplazamiento, δ'_x , comprendida entre $[-\frac{1}{2}, \frac{1}{2}]$. De esta forma, el estimador para la parte fraccional queda dado por

$$\check{\delta}'_x = \frac{\text{Ln}[y(-1)] - \text{Ln}[y(1)]}{2 \text{Ln}[y(-1)] - 4 \text{Ln}[y(0)] + 2 \text{Ln}[y(1)]}.$$

Incorporando la hipótesis de que la función y tiene una distribución gaussiana, puede escribirse

$$y(-1) = A e^{-\frac{1}{2\sigma^2}(1+\delta'_x)^2} + W_1,$$

$$y(0) = A e^{-\frac{\delta'^2_x}{2\sigma^2}} + W_2,$$

$$y(1) = A e^{-\frac{1}{2\sigma^2}(1-\delta'_x)^2} + W_3,$$

Donde A representa la amplitud de la señal, y W_1 , W_2 y W_3 son variables aleatorias que representan ruido aditivo presente en la señal, y se asume que estas son independientes y están idénticamente distribuidas con media cero y desvío estándar σ_R .

A continuación, se introduce la función $Y = \check{\delta}'_x - \delta'_x$, la cual es una variable aleatoria que representa el desvío entre el desplazamiento estimado y el

desplazamiento verdadero para distintas realizaciones. Se busca determinar la función de valor cuadrático medio, $VCM = \sqrt{E[Y^2]}$, para cuantificar la incertidumbre de la estimación, donde $E[Y^2]$ representa el valor esperado de la variable aleatoria Y^2 . Para ello, se plantea el desarrollo de Taylor de primer orden de la variable Y en función de las fuentes de ruido,

$$Y \approx Y_0 + \frac{\partial Y}{\partial W_1} W_1 + \frac{\partial Y}{\partial W_2} W_2 + \frac{\partial Y}{\partial W_3} W_3,$$

Donde Y_0 y las derivadas parciales están evaluadas para el valor medio de las variables W_1 , W_2 y W_3 . A partir de esta expresión, y teniendo en cuenta que $E[W_1^2] = E[W_2^2] = E[W_3^2] = \sigma_R^2$, la forma aproximada para el valor cuadrático medio viene dada por

$$VCM \approx \sigma_R \sqrt{\left(\frac{\partial Y}{\partial W_1}\right)^2 + \left(\frac{\partial Y}{\partial W_2}\right)^2 + \left(\frac{\partial Y}{\partial W_3}\right)^2}.$$

Al plantear las derivadas parciales de Y con respecto a cada una de las fuentes de error, evaluadas para el valor medio del ruido, se obtiene

$$\frac{\partial Y}{\partial W_1} = \frac{-\sigma^2(1 - 2\delta'_x)}{2A} e^{\frac{1}{2\sigma^2}(1+\delta'_x)^2},$$

$$\frac{\partial Y}{\partial W_2} = 2\sigma^2 \frac{\delta'_x}{A} e^{\frac{\delta'^2_x}{2\sigma^2}},$$

$$\frac{\partial Y}{\partial W_3} = \frac{-\sigma^2(1 - 2\delta'_x)}{2A} e^{\frac{1}{2\sigma^2}(1+\delta'_x)^2}.$$

Introduciendo el valor de estas derivadas en la expresión para la función VCM , finalmente resulta

$$VCM \approx \frac{\sigma^2}{2 SNR} \sqrt{(1 - 2\delta'_x) e^{\frac{1}{\sigma^2}(1+\delta'_x)^2} + (1 + 2\delta'_x) e^{\frac{1}{\sigma^2}(1-\delta'_x)^2} + \delta'^2_x e^{\frac{\delta'^2_x}{\sigma^2}}},$$

donde $SNR = A^2/\sigma_R^2$ representa la relación señal-ruido. El valor de σ que minimiza esta función puede obtenerse analíticamente para $\delta'_x = 0$ y vale $\sigma = 1/\sqrt{2}$, mientras que para $\delta'_x = 0,5$ vale $\sigma = 1/(2\sqrt{2})$. En la Figura 26 se muestran las curvas para el valor cuadrático medio teórico, y se las compara con resultados obtenidos mediante simulación numérica y el método de Montecarlo. Para las simulaciones, se generaron curvas gaussianas aleatorias con factor de escala σ y media δ'_x , con pico máximo de 3, a las que se les agregó ruido con distribución normal, media 0 y desvío estándar de 0,1. El factor de escala para cada curva se tomó aleatoriamente entre 0,1 y 2 píxeles, y se evaluaron los casos para desplazamientos verdaderos de 0, 0,3 y 0,5 píxeles. Cada caso involucra el procesamiento de 1.000.000 curvas gaussianas. Para cada curva se estimó el desplazamiento por medio de ajuste gaussiano de tres puntos, y se calculó el desvío

de la estimación, con el que se estimó el valor cuadrático medio del error en función de σ . Se observa que la curva teórica para $\delta'_x = 0$ ajusta bien a los resultados de la simulación, mientras que para $\delta'_x = 0,5$, el valor óptimo predicho para σ es bastante inferior al obtenido mediante la simulación. Para concluir, el valor óptimo para σ se encuentra en torno a $1/\sqrt{2}$. Teniendo en cuenta que el factor de escala de la partícula σ_p se relaciona con el del pico de correlación por medio de $\sigma_p = \sigma/\sqrt{2}$, y considerando el diámetro de la partícula como $D_p = 4\sigma_p$, entonces el diámetro óptimo de la imagen de partículas para la estimación del desplazamiento por ajuste gaussiano se encuentra en torno a los 2 píxeles.

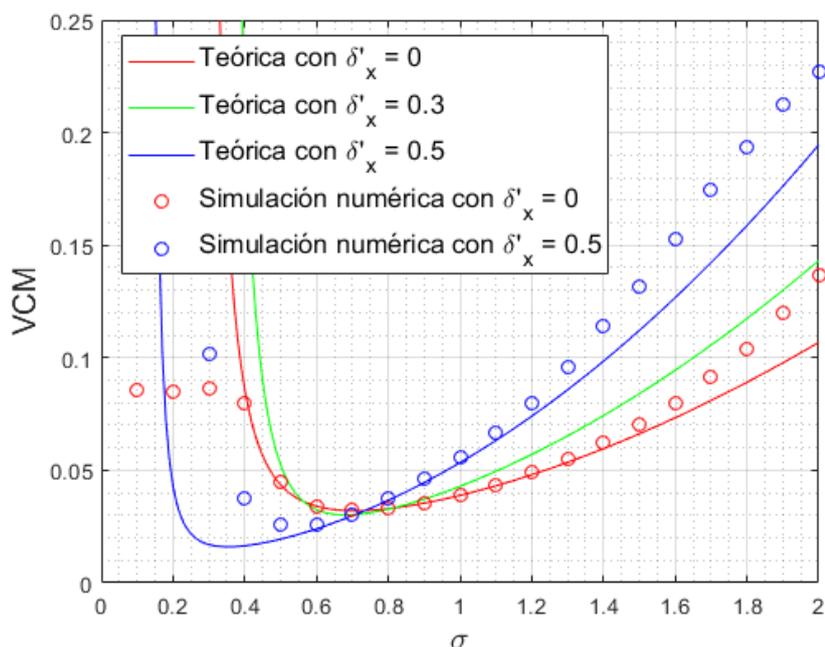


Figura 26 – Valor cuadrático medio del error en la estimación del desplazamiento por medio del método de ajuste gaussiano de tres puntos.

V.8. COMPARACIÓN ENTRE LOS DISTINTOS MÉTODOS DE ESTIMACIÓN SUBPÍXEL

En esta sección se evalúan el error aleatorio y sistemático producido por distintos métodos de estimación subpíxel en la estimación del desplazamiento de imágenes de partículas, a través de ensayos de Montecarlo. Los ensayos consisten en la evaluación del desplazamiento de imágenes de partículas, por medio de la correlación cruzada estándar y los distintos métodos de estimación subpíxel, en 500.000 pares de ventanas generadas de forma sintética. Las ventanas consisten en arreglos cuadradas de distinto tamaño (la primera ventana de 32x32 píxeles, la segunda de 64x64 píxeles), generados de forma sintética como se detalla en el anexo IX.10, con los parámetros seleccionados de forma aleatoria dentro de los rangos establecidos por los límites de la Tabla 1. Los desplazamientos verdaderos para las partículas de cada par de ventanas se

seleccionaron aleatoriamente en el rango $[-\frac{1}{2}, \frac{1}{2}]$. Se consideraron gradientes de desplazamiento horizontal du/dx y du/dy , tomados de forma aleatoria en el rango de 0 a $1,5/32$.

Para mejor claridad de los gráficos, los métodos fueron segmentados en 3 grupos, tomando el método de estimación gaussiana de tres puntos como referencia. En la Figura 27 se muestran el error aleatorio y sistemático en función del desplazamiento verdadero, producidos por los métodos basados en ajuste gaussiano, mientras que en la Figura 28 se muestran los métodos basados en ajuste cuadrático, y en la Figura 29 los basados en flujo óptico. Comparado con el ajuste gaussiano de tres puntos, el ajuste gaussiano por mínimos cuadrados muestra una ligera reducción del error aleatorio, aunque manifiesta un mayor error sistemático y mayor variabilidad con el desplazamiento, significando un mayor efecto de enclavamiento de picos. Los métodos basados en flujo óptico presentan un comportamiento similar, logrando una reducción significativa del error aleatorio, a costa de un incremento en el sesgo de las estimaciones. Por otro lado, los métodos de ajuste cuadrático muestran una reducción en el error aleatorio para desplazamientos bajos, pero muestran peor desempeño que el método de ajuste gaussiano de tres puntos para desplazamientos en torno a los 0,5 píxeles, y a su vez, presentan un peor desempeño en cuanto a error sistemático. Se concluye que, de los métodos analizados, el más robusto es el método de ajuste gaussiano de tres puntos, por presentar un mejor desempeño general teniendo en cuenta ambos tipos de error, y por mostrar una menor variabilidad del error con el desplazamiento, significando en un menor efecto de enclavamiento de picos.

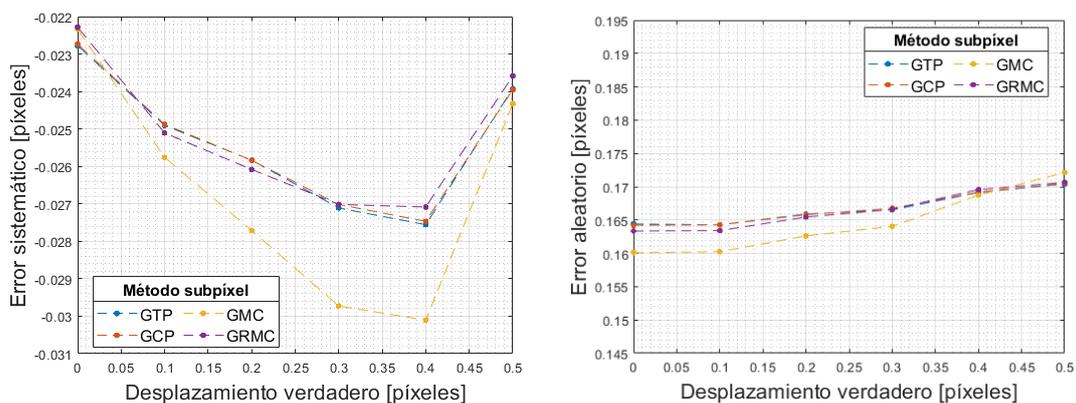


Figura 27 – Evaluación de los errores sistemático y aleatorio en función del desplazamiento verdadero para los métodos GTP (ajuste gaussiano de tres puntos), GCP (ajuste gaussiano de cuatro puntos), GMC (ajuste gaussiano por mínimos cuadrados), y GRMC (ajuste gaussiano rotado por mínimos cuadrados).

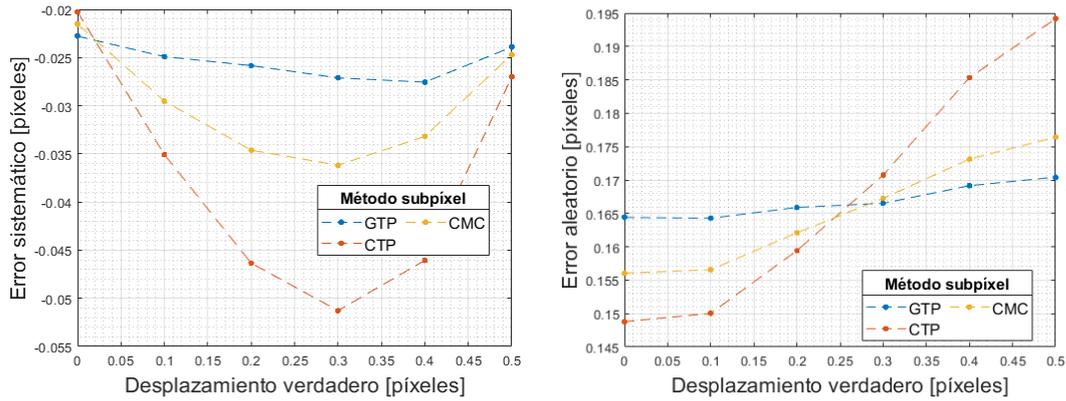


Figura 28 – Evaluación de los errores sistemático y aleatorio en función del desplazamiento verdadero para los métodos CTP (ajuste cuadrático de tres puntos) y CMC (ajuste cuadrático por mínimos cuadrados). Se incluye el ajuste gaussiano de tres puntos como referencia.

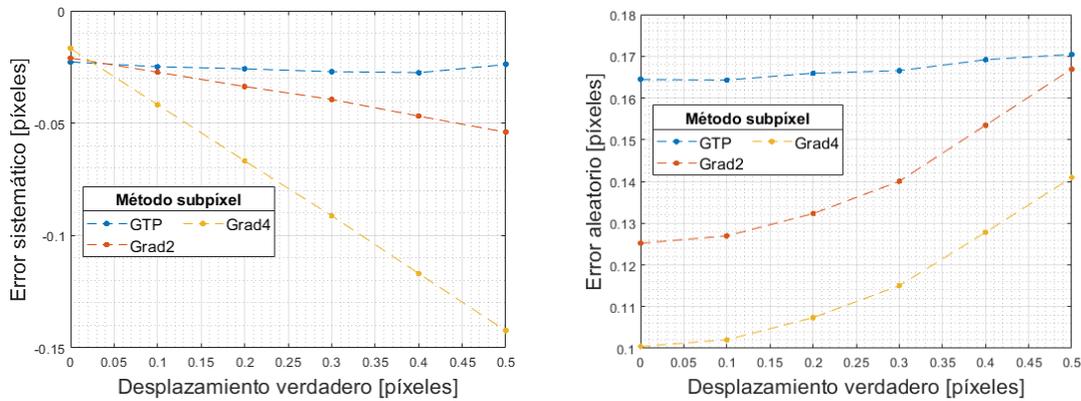


Figura 29 – Evaluación de los errores sistemático y aleatorio en función del desplazamiento verdadero para los métodos de flujo óptico, Grad2 (gradiente con esquema de diferencias finitas de segundo orden), y Grad4 (gradiente con esquema de diferencias finitas de cuarto orden). Se incluye el ajuste gaussiano de tres puntos como referencia.

V.9. ERROR ASOCIADO A LOS COMPONENTES ESTACIONARIOS DE LA IMAGEN

En esta sección se analizan los efectos asociados a la presencia de componentes estacionarios en las imágenes utilizadas en PIV. Con frecuencia, componentes estacionarios aparecen en las imágenes naturalmente debido a la presencia de paredes y cuerpos sólidos que forman parte del sistema experimental. Particularmente en este trabajo, en el que se desea medir la velocidad del fluido dentro de un medio poroso, la presencia de componentes estacionarios es inevitable. Por lo tanto, es de interés cuantificar estos errores.

Para la estimación del error asociado a la presencia de componentes estacionarios en la imagen, se generaron ventanas sintéticas tal como se describe en el anexo IX.10, variando aleatoriamente los parámetros de generación según los límites de la Tabla 1, pero agregando además un componente estacionario. En un primer escenario analizado,

se agregó un componente estacionario en forma de área rectangular de intensidad constante (con la adición de ruido), de manera de ocupar parcialmente la región inferior de la ventana y simulando la presencia de una superficie sólida, mientras que, en un segundo escenario, el componente estacionario se dispuso de manera de ocupar la región derecha. En la Figura 30 se muestran a modo de ejemplo dos ventanas generadas aleatoriamente, correspondientes a cada escenario. El movimiento de las partículas se aplicó de izquierda a derecha y de forma uniforme. No se tuvo en cuenta ninguna interacción entre los componentes estacionarios y de las partículas. Las partículas simplemente se desplazan por detrás de estos, como si se tratase de cuerpos que se encuentran en un plano diferente al del plano de iluminación, y obstruyen parcialmente la imagen. Esta es una situación presentada experimentalmente en el marco de este trabajo al intentar medir el campo de velocidades dentro del medio poroso. Las técnicas consideradas para este estudio fueron el uso de correlación cruzada estándar, el uso de correlación cruzada estándar con enmascaramiento (se aplica una máscara para llevar a 0 el nivel de intensidad de la región ocupada por el componente estacionario), y correlación cruzada estándar con técnica de máscara digital (la normalización de la correlación cruzada se efectúa teniendo en cuenta las máscaras, según Ec. II.6-2). Esta última técnica fue empleada para reducir el error en aplicaciones PIV a flujos bifásicos (Gui, Merzkirch, & Fei, 2000; Gui, Wereley, & Kim, 2003). Se utilizaron pares de ventanas de distinto tamaño (32x32 píxeles para la primera ventana y 64x64 píxeles para la segunda ventana), y para generar las estadísticas de cada caso se procesaron 100.000 pares de ventanas.

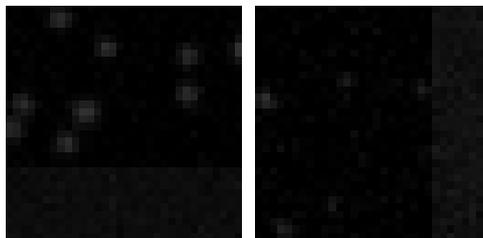


Figura 30 – Ventanas sintéticas generadas aleatoriamente para la evaluación del error asociado a la presencia de componentes estacionarios para los dos casos analizados, mostradas a modo de ejemplo.

En la Figura 31 se muestran los resultados obtenidos para los errores en la estimación del desplazamiento horizontal en función del desplazamiento verdadero, para el escenario con componente estacionario en la región inferior de las ventanas. Como puede observarse, en este caso en el que las partículas se mueven en el sentido paralelo al componente estacionario, no hay una contribución al error sistemático asociado, y los tres métodos producen errores sistemáticos similares. Sin embargo, la presencia de esta región genera un incremento en las variaciones del error aleatorio, tal como se aprecia en la figura. La magnitud de estas variaciones puede reducirse mediante el

enmascaramiento de la región con componente estacionario, o mediante el uso de la técnica de máscara digital. Por otro lado, en la Figura 32 se muestran los resultados para el escenario con el componente estacionario a la derecha de la ventana. En este caso, debido que las partículas se mueven hacia la región ocupada por el componente estacionario, una parte de ellas queda oculta detrás del componente en la segunda ventana, generando un efecto de pérdida de pares, que se manifiesta como un incremento en el error sistemático de las estimaciones. Enmascarar la región en este caso solo genera un incremento en este error. Sin embargo, el uso de la técnica de máscara digital logra efectivamente reducir el error sistemático. El error aleatorio por otro lado presenta un comportamiento similar al del caso previamente analizado. La técnica de máscara digital muestra ser efectiva para reducir tanto el sesgo como las variaciones con respecto a la parte fraccional del desplazamiento verdadero del error aleatorio, producidas por la presencia de un componente estacionario en las imágenes.

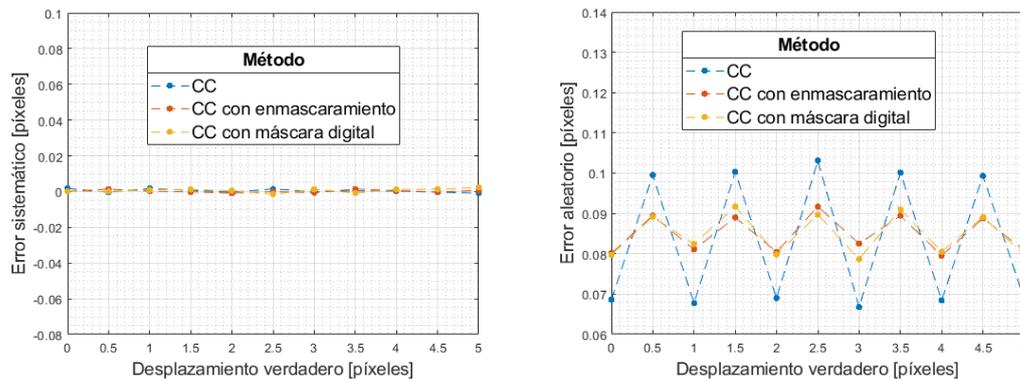


Figura 31 – Evaluación de los errores sistemático y aleatorio en función del desplazamiento verdadero horizontal, para las técnicas de correlación cruzada estándar (CC), correlación cruzada con enmascaramiento y correlación cruzada con máscara digital, para el escenario con componente estacionario en la región inferior de la ventana.

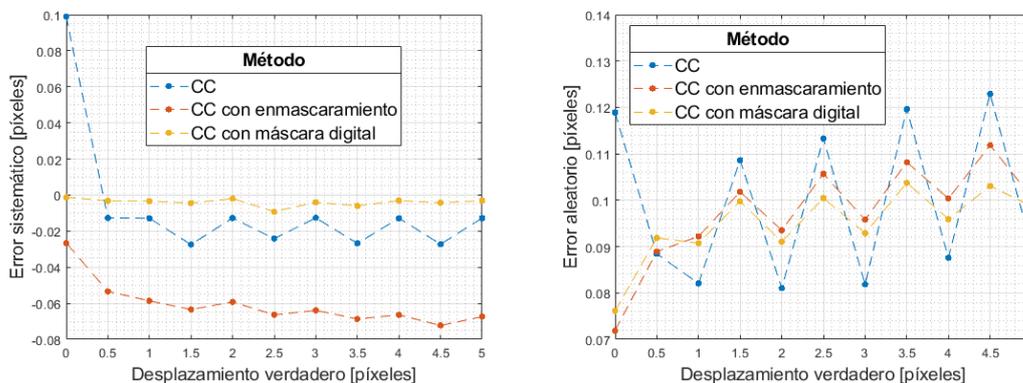


Figura 32 – Evaluación de los errores sistemático y aleatorio en función del desplazamiento verdadero horizontal, para las técnicas de correlación cruzada estándar (CC), correlación cruzada con enmascaramiento y correlación cruzada con máscara digital, para el escenario con componente estacionario en la región derecha de la ventana.

V.10. DESEMPEÑO DEL MÉTODO SUPER RESOLUCIÓN

En esta sección se presenta un análisis del desempeño del método PIV Super Resolución, comparado con el método PIV. Para tal fin, se realizó un ensayo de Montecarlo sobre imágenes generadas sintéticamente. Estas imágenes fueron creadas utilizando una imagen de fondo basada en los experimentos realizados en el marco de esta tesis sobre el medio poroso, con el objetivo de replicar estas condiciones. La imagen de fondo se generó tomando una serie de imágenes capturadas en el laboratorio y aplicando el método de mínimo histórico. Sobre esta imagen de fondo se añadieron partículas con forma gaussiana y se adicionó ruido, tal como se describe en el anexo IX.10. El tamaño, densidad e intensidad de las partículas se aplicó de forma de replicar lo mejor posible las características de las imágenes experimentales. En la Figura 33 se muestra la imagen de fondo utilizada, y se compara una imagen experimental con una generada sintéticamente. El campo de velocidades impuesto para las partículas en la serie de imágenes sintéticas también se determinó de forma de replicar lo mejor posible las condiciones experimentales. Para ello, una serie de imágenes experimentales se procesó mediante la técnica de PIV para obtener un conjunto de campos de velocidad en el tiempo. Estos campos fueron promediados en el tiempo para obtener un campo medio, el cual fue utilizado para la generación de los pares de imágenes sintéticas. En la Figura 34 se muestra la componente horizontal de los campos utilizados para generar las imágenes sintéticas. Se consideran dos casos, un caso 1 con un desplazamiento horizontal máximo de 3,5 píxeles, y otro caso 2 con un desplazamiento horizontal máximo de 17,3 píxeles. Para el procesamiento por medio de la técnica PIV se utilizó un esquema de pasos múltiples con deformación de imagen, en la que se aplicaron dos etapas de procesamiento. En la primera etapa se utilizaron ventanas de 32x32 píxeles con un espaciado entre puntos de grilla de 16 píxeles, mientras que en la segunda etapa se utilizaron ventanas de 16x16 píxeles con espaciado entre puntos de grilla de 8 píxeles. Se aplicaron funciones de ventaneo tipo Hamming y método de correlación cruzada estándar con normalización y aplicación de máscara digital. La estimación subpíxel se efectuó mediante el método de ajuste gaussiano de tres puntos. La validación se realizó mediante la técnica de red neuronal celular con umbral constante. Para la interpolación del campo de desplazamientos y de las imágenes, requeridas para el método de deformación, se aplicó interpolación basada en splines cúbicas. Por otro lado, el método de Super Resolución se aplicó tal como se describe en la sección III, y tomando el campo de desplazamiento obtenido con el método PIV como base.

En la Figura 35 se muestran los resultados obtenidos para el caso 1 mediante la aplicación de la técnica PIV. Se observa un error sistemático cercano a 0 en la zona de flujo libre, mientras que errores mayores se observan en las regiones interiores al medio poroso y cercanas a las paredes, llegando a desvíos de hasta 0,6 píxeles. Por otro lado, el error aleatorio posee valores inferiores a 0,1 píxeles en las regiones de flujo libre, pero alcanzando desvíos de hasta 0,6 píxeles en las regiones cercanas a los límites sólidos. Si se analizan en la Figura 36 los resultados obtenidos para el caso 2 mediante la aplicación de la técnica PIV, se observa un comportamiento similar que para el caso 1 en las zonas de flujo libre. Sin embargo, se presenta un incremento significativo de ambos tipos de

error en las regiones dentro del medio poroso y cercanas a las paredes, alcanzando desvíos mayores a 1 píxel. Por otro lado, se observa que aplicando la técnica de Super Resolución, se logran reducir estos desvíos, alcanzando errores semejantes tanto en la zona interior al medio poroso como fuera de ella. Se concluye que el empleo del método de PIV Super Resolución puede ser superior al método PIV para la determinación del campo de velocidades en las regiones interiores al medio poroso y cercanas a las paredes, sobre todo para velocidades altas.

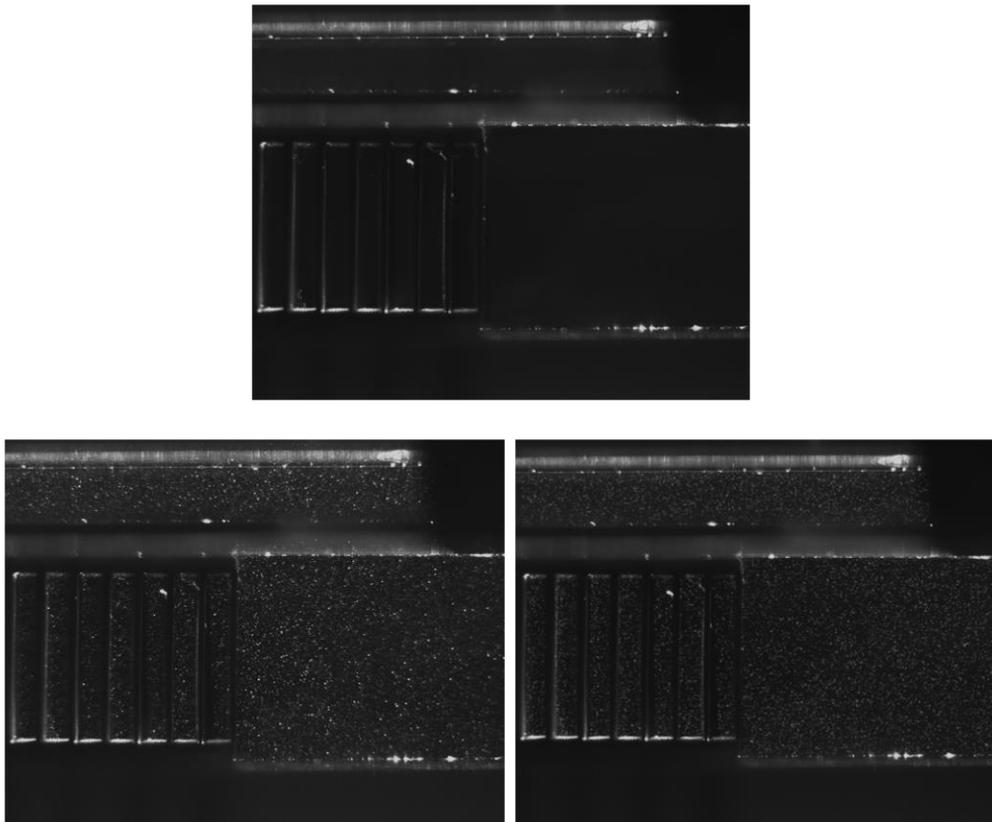


Figura 33 – Generación de imágenes sintéticas para la evaluación de desempeño PIV y PIV SR. Arriba, imagen de fondo utilizada, generada por medio de la técnica de mínimo histórico sobre una serie de imágenes reales. Abajo a la izquierda, imagen real obtenida en el laboratorio. Abajo a la derecha, imagen generada de forma sintética adicionando imágenes de partículas y ruido sobre la imagen de fondo.

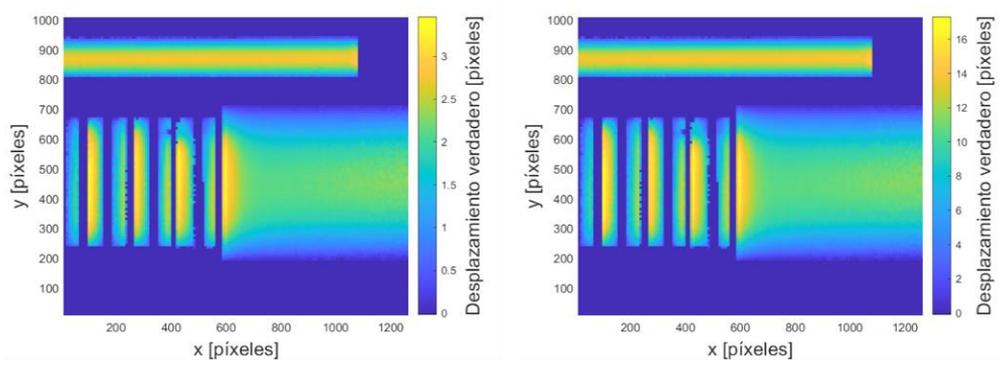


Figura 34 – Componente de desplazamiento horizontal utilizada para la generación de los pares de ventanas sintéticos. A la izquierda se muestra el desplazamiento para el caso 1, mientras que a la derecha se muestra el desplazamiento para el caso 2.

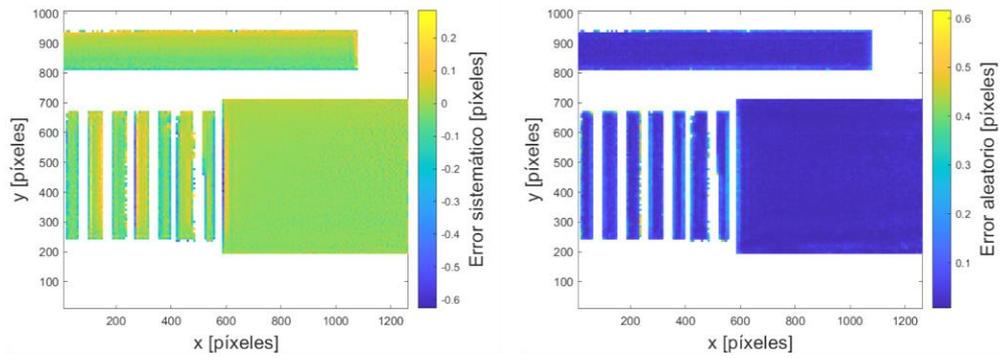


Figura 35 – Errores sistemático y aleatorio para la estimación de la componente horizontal de desplazamiento en la serie de imágenes sintéticas para el caso 1 mediante el método PIV.

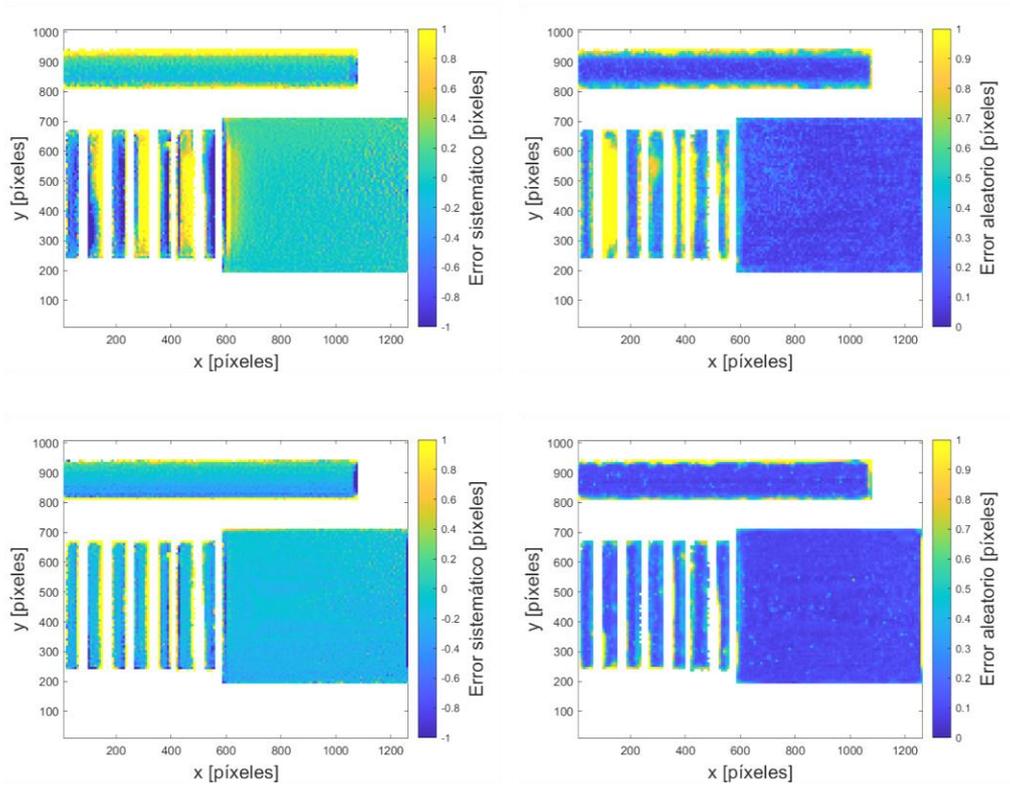


Figura 36 – Errores sistemático y aleatorio para la estimación de la componente horizontal de desplazamiento en la serie de imágenes sintéticas para el caso 2. Arriba, método PIV. Abajo, método PIV con super resolución.

VI. ANÁLISIS CASO PERMEABLE

La motivación del presente trabajo es avanzar en técnicas experimentales para analizar el efecto de recubrimientos porosos anisotrópicos en flujos internos. La cantidad utilizada para caracterizar estos efectos es el coeficiente de permeabilidad.

Los medios para el ensayo fueron diseñados y construidos por manufactura aditiva, utilizando impresión 3D. Estos fueron posteriormente recubiertos con pintura negra opaca para limitar la cantidad de luz reflejada en la superficie de los medios porosos durante la etapa de captura de imágenes. Para medir los efectos de cada medio, estos se dispusieron en la pared de un canal de pruebas.

El montaje del experimento es como el que se muestra en la Figura 37. Este consta de un canal prismático vertical de sección rectangular. El fluido de trabajo es aire, el cual se recircula a través de una cañería conectada en los extremos del canal, conformando un sistema cerrado. El flujo es forzado por medio de un ventilador colocado en el tramo superior, justo en la salida del canal, y el caudal puede ser regulado a partir de la tensión de alimentación del ventilador. Para el sembrado de partículas se utiliza una niebla de aceite de oliva, la cual es generada por medio de un dispositivo de burbujeo fabricado en el laboratorio. La longitud del medio poroso es de 30 cm. Este se coloca sobre la pared, a aproximadamente 110 cm del comienzo del canal. Una vista más detallada del sistema a la altura del medio poroso se muestra en la Figura 38. Sobre la cara del medio poroso que queda expuesta al flujo se adhiere una lámina separadora de 40 cm de longitud, de forma de dividir el flujo de esta zona en dos regiones aisladas, una de flujo libre, y otra de flujo a través del medio poroso. A la misma altura se colocan dos láseres, los cuales, mediante un par de lentes cilíndricas, generan dos haces de luz planos. Estos iluminan tanto el interior del medio poroso como el exterior de este. Finalmente, una cámara se dispone de forma de capturar la región de interés, que comprende la región a la salida del medio poroso, delimitada por el recuadro rojo en la Figura 38.

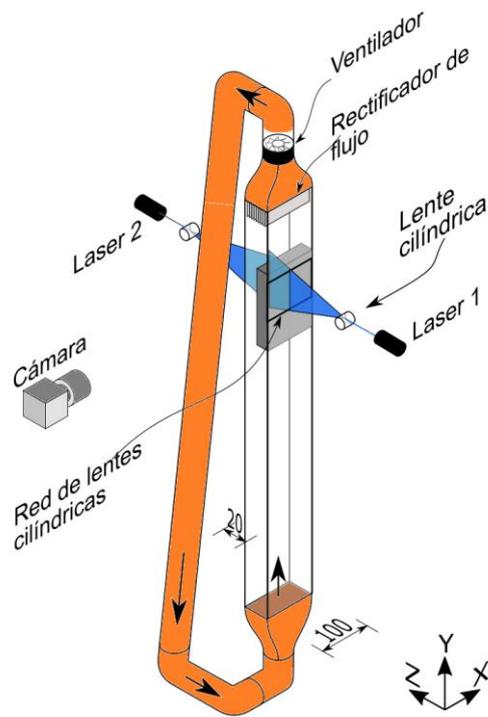


Figura 37 – Esquema del experimento realizado.

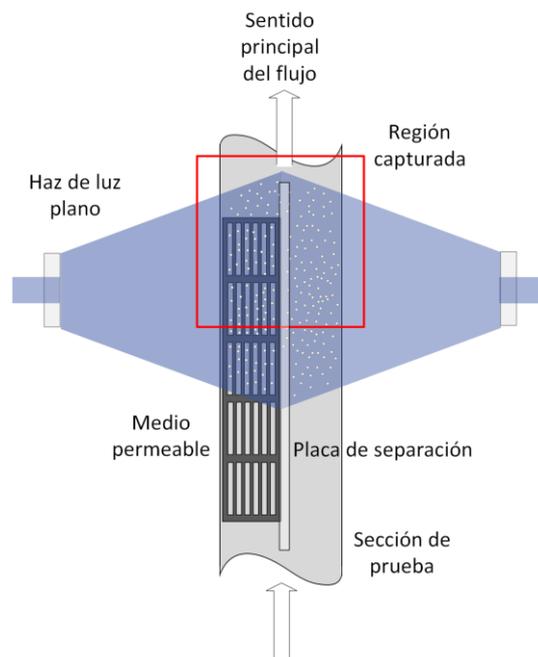


Figura 38 – Vista paralela al plano de la región capturada.

Al realizar el experimento, se obtienen una serie de imágenes de partículas, las cuales, aplicando PIV, permiten obtener el campo de velocidades proyectado en el plano

iluminado. A partir de este campo de velocidades es posible obtener cantidades derivadas, entre las que se encuentra el coeficiente de permeabilidad.

El coeficiente de permeabilidad se obtiene considerando una serie de hipótesis. En primer lugar, se asume que es válida la ley de permeabilidad de Darcy (sección 1.2), la cual establece que el caudal Q circulante a través del medio poroso está relacionado con la caída de presión ΔP a lo largo del mismo por medio de la expresión

$$Q = k \frac{A \Delta P}{\mu L},$$

Siendo A el área de flujo, μ la viscosidad dinámica del fluido, L la longitud del medio poroso en el sentido del flujo y k el coeficiente de permeabilidad.

El caudal circulante dentro del medio puede ser estimado a partir de la velocidad media dentro del medio poroso. Por otro lado, la caída de presión puede determinarse utilizando el perfil de velocidad en la zona de flujo libre, asumiendo de que este flujo es de tipo Poiseuille. Sean x e y las componentes espaciales sobre el plano de iluminación, en el sentido del flujo y perpendicular al sentido del flujo, respectivamente. Sean u y v las componentes de velocidad proyectadas en la dirección x y en la dirección y , respectivamente. Planteando un balance de cantidad de movimiento diferencial en la dirección del flujo para un fluido newtoniano, con régimen de flujo laminar, se tiene

$$\frac{\partial p}{\partial x} = -\rho \left\{ \frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} \right\} + \mu \left\{ \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right\}.$$

Luego, considerando flujo totalmente desarrollado, estado estacionario, y velocidad nula en la dirección transversal al flujo, el balance se reduce a

$$\frac{\partial p}{\partial x} = \mu \frac{\partial^2 u}{\partial y^2}.$$

La velocidad en el sentido del flujo en función de la componente transversal al flujo forma un perfil parabólico, que puede ajustarse mediante la expresión

$$u(y) = a y^2 + b y + c,$$

para ciertas constantes a , b y c . Estas constantes pueden determinarse utilizando un ajuste por mínimos cuadrados. Introduciendo este ajuste en la expresión anterior, se obtiene

$$\frac{\Delta P}{L'} = 2 \mu a,$$

donde L' es la longitud del tramo donde se desarrolla el flujo tipo Poiseuille. Combinando esta expresión con la ley de permeabilidad de Darcy, resulta

$$\frac{Q}{A} = k \frac{2 a L'}{L}.$$

La cantidad Q/A puede estimarse como la velocidad media dentro del medio poroso, $\langle u_m \rangle$, por lo que finalmente, el coeficiente de permeabilidad puede calcularse como

$$k = \frac{\langle u_m \rangle L}{2 a L'}.$$

En el experimento llevado a cabo utilizando PIV en el plano, la cantidad $\langle u_m \rangle$ no puede ser determinada, sino que se obtiene una cantidad $\langle u_m \rangle_p$, que representa la velocidad media a través del medio poroso, calculada en el plano iluminado. Debido a que la velocidad disminuye cerca de las paredes, la cantidad $\langle u_m \rangle_p$ depende de la ubicación del plano de iluminación, siendo máxima en el centro del canal. Para tener en cuenta estos efectos, la cantidad Q puede reemplazarse en su lugar por una cantidad $\langle u_m \rangle_p A_{ef}$, donde A_{ef} es un área efectiva, que representa la sección de flujo que debería tener el canal para producir el caudal Q , si la velocidad media del flujo en toda la sección fuera $\langle u_m \rangle_p$. De esta forma, el coeficiente k puede calcularse como

$$k = c \frac{\langle u_m \rangle_p L}{2 a L'},$$

donde $c = A_{ef}/A$. Este método entonces permite estimar el coeficiente de permeabilidad haciendo uso solamente de factores geométricos (la longitud del tramo libre y del medio poroso y el área de flujo), y cantidades derivables directamente a partir del campo de velocidades obtenible por PIV (la velocidad media del flujo dentro del medio poroso y el parámetro de ajuste a). La cantidad c depende de la ubicación del plano de iluminación y de la forma del perfil de velocidad desarrollado en sus tres dimensiones y no se conoce a priori. Se realizó una simulación numérica del canal para flujo laminar en donde se determinó el valor de $c = 0,907$.

Para la estimación de la incertidumbre en la determinación de k , se analizó la contribución al error de cada una de las cantidades implicadas en su cálculo. A partir de la propagación de errores, la incertidumbre en la estimación de k puede evaluarse como

$$\delta k = k \sqrt{\left(\frac{\delta \langle u_m \rangle_p}{\langle u_m \rangle_p}\right)^2 + \left(\frac{\delta a}{a}\right)^2 + \left(\frac{\delta L}{L}\right)^2 + \left(\frac{\delta L'}{L'}\right)^2},$$

donde δk representa la incertidumbre en la estimación de k , mientras que $\delta \langle u_m \rangle_p$, δa , δL , $\delta L'$ y representan la incertidumbre de cada una de las cantidades involucradas en el cálculo de k . Las fuentes de incertidumbre de $\langle u_m \rangle_p$ se encuentran principalmente en el error asociado de la técnica PIV y al error introducido por los efectos de movimiento fuera de plano. Considerando que el fluido se comporta de manera incompresible, $\langle u_m \rangle_p$ debería ser aproximadamente igual en cada sección perpendicular al flujo para cumplir con el principio de conservación de masa. No obstante, los movimientos fuera

de plano y el error propio de la técnica PIV ocasionaran fluctuaciones en la determinación de $\langle u_m \rangle_p$ sobre distintas secciones del flujo. La incertidumbre en la determinación de $\langle u_m \rangle_p$ puede entonces estimarse a partir del desvío estándar de $\langle u_m \rangle_p$ calculado en distintas secciones del flujo. La incertidumbre de a depende de la bondad del ajuste del perfil de velocidades a la curva cuadrática, y puede tomarse como el ancho del intervalo de confianza del 95% para a . Las incertidumbres para L , L' dependen de la precisión de los instrumentos de medición utilizados para medirlas, y se tomarán como de $\pm 1 \text{ mm}$.

Los medios porosos utilizados en el experimento, de material plástico y producidos mediante impresión 3D, fueron diseñados como una red de barras formadas por dos filamentos paralelos. La matriz tomada para el diseño de los medios se muestra en la Figura 39, junto a las vistas a través de sus ejes principales. Es de interés en particular, en el marco de este trabajo, determinar el coeficiente de permeabilidad de Darcy de este medio cuando está expuesto a flujos en las direcciones de estos tres ejes. Para ello, se diseñaron dos medios porosos, cuyos esquemas son mostrados en la Figura 40. De esta forma, el flujo fue evaluado para cuatro escenarios representados por las orientaciones de los medios porosos denominadas A, B, B' y C. En la Figura 41 se muestran fotografías de uno de los medios impresos para los experimentos.

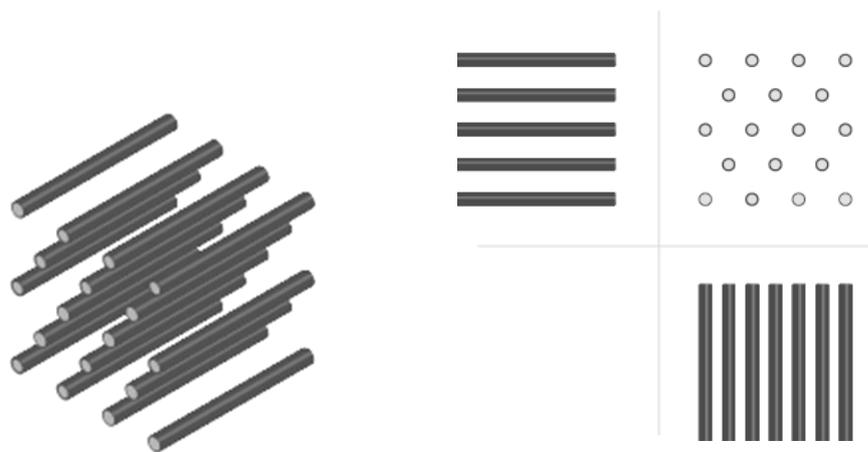


Figura 39 – Matriz tomada para la elaboración de los medios porosos (izquierda) y vista a través de sus ejes principales (derecha).

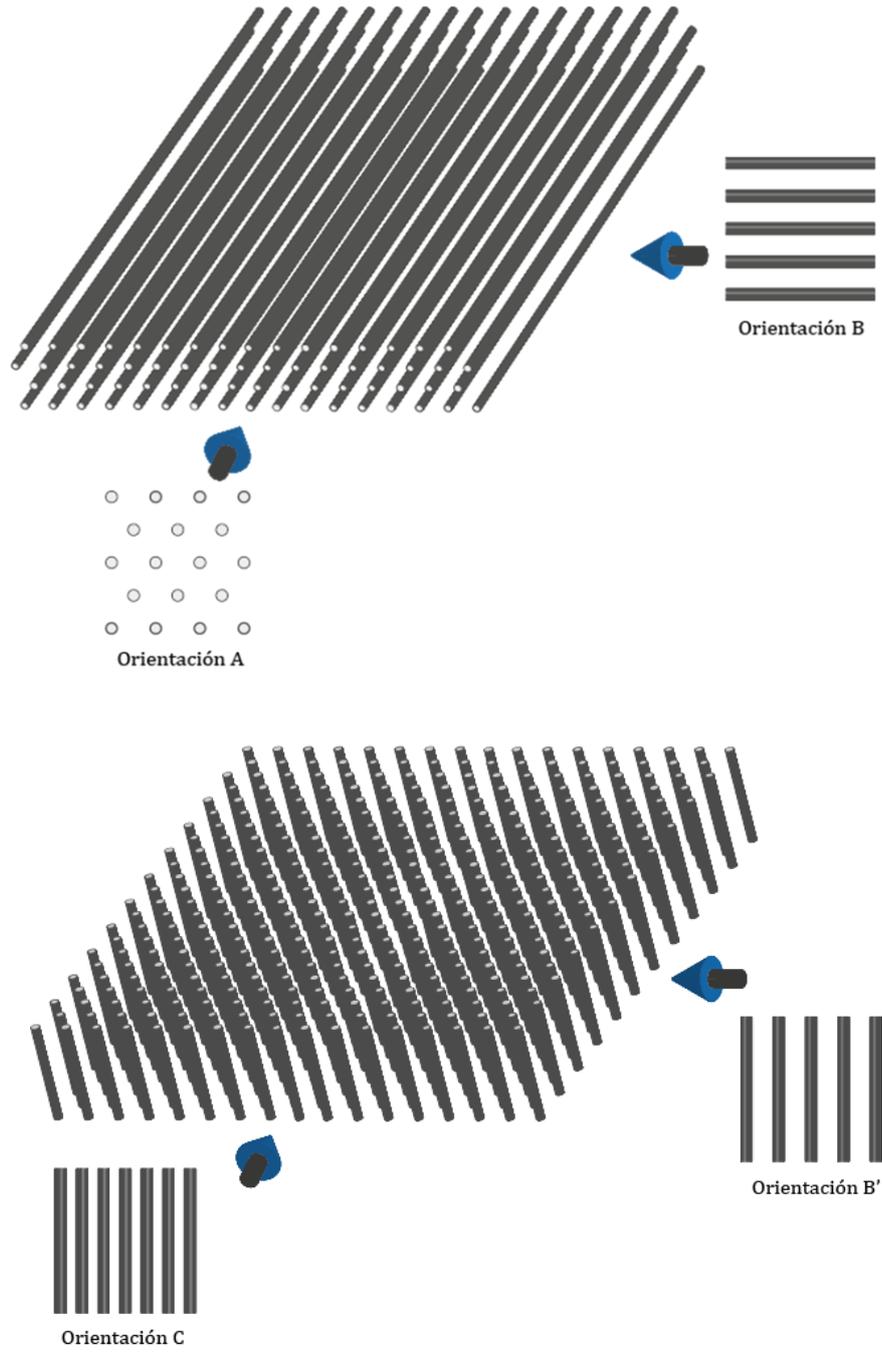


Figura 40 – Esquema de los medios porosos diseñados y las orientaciones ensayadas.

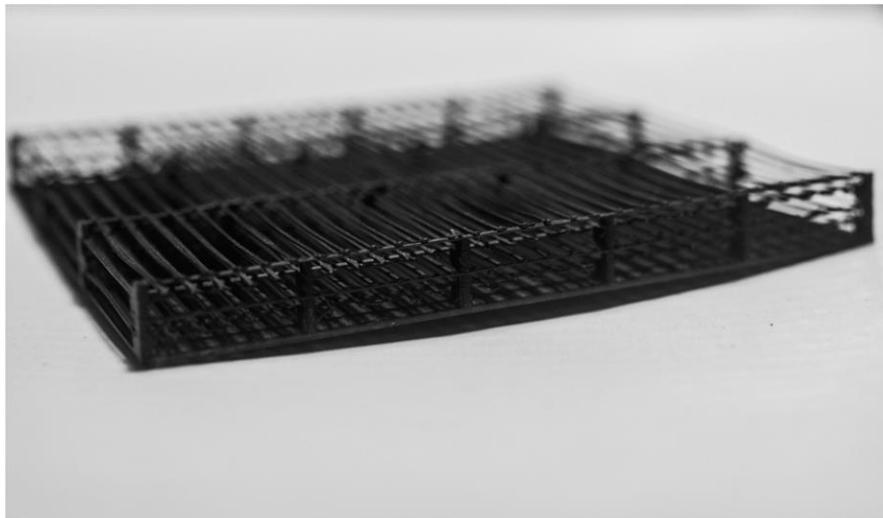
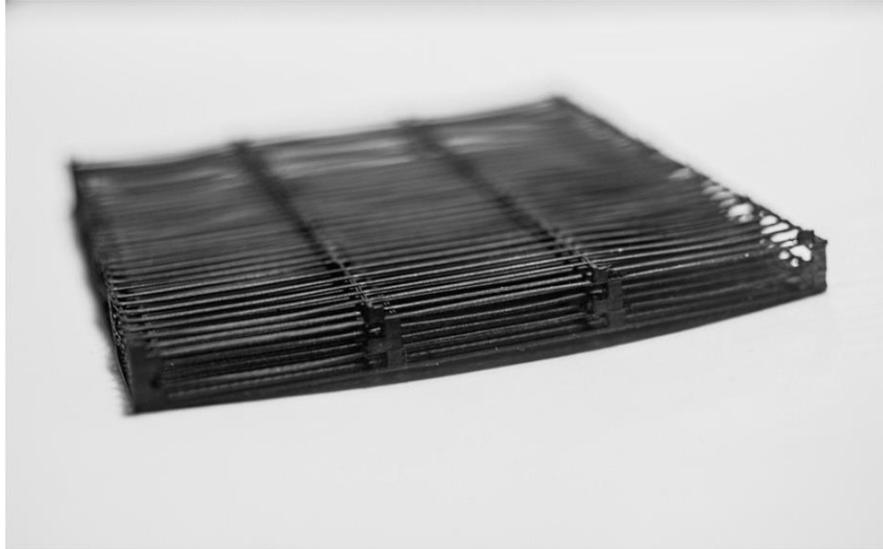


Figura 41 – Fotografías de uno de los medios porosos impresos para llevar a cabo el experimento.

Para cada orientación de los medios porosos, se capturaron series de imágenes utilizando distintas tensiones de alimentación del ventilador, para generar distintas condiciones de flujo. Para cada orientación del medio y cada velocidad ensayada, se analizaron 50 pares de imágenes. En la Figura 42 se muestra la primera imagen de cada serie de capturas PIV tomadas para cada orientación del medio poroso, para el caso de menor velocidad media en los poros. Estas imágenes muestran iluminación de fondo, y reflexiones de luz en las paredes del canal y en la superficie del medio poroso, que representan fuentes de error. Estas reflexiones pueden eliminarse mediante sustracción de fondo. En este trabajo se empleó la técnica de sustracción de fondo basada en el mínimo histórico (sección II.1). Las imágenes de fondo generadas por este método se muestran en la Figura 43. Si a las imágenes de la Figura 42 se les aplica sustracción de fondo, se obtienen las imágenes de la Figura 44. Las imágenes luego de la sustracción

de fondo se encuentran más próximas al ideal de partículas brillantes en un fondo oscuro. Para disminuir el tiempo de procesamiento, se crearon máscaras binarias, utilizadas para indicar sobre qué puntos de la grilla realizar correlación cruzada. Para la generación de estas máscaras se utilizó el método basado en la prueba de normalidad de cada píxel, seguido de un ajuste manual (sección II.2). En la Figura 45 se muestra, a modo de ejemplo, una imagen experimental cruda, en donde se discriminan las zonas de flujo en color azul, y zonas que presentan imágenes reflejadas en rojo. La máscara obtenida empleando el algoritmo de generación automática de máscara sobre la serie de esta imagen se muestra en la imagen de la izquierda de la Figura 46, donde se observa que las regiones correspondientes a las paredes del canal y las partes sólidas del medio poroso fueron correctamente enmascaradas, con la excepción de las regiones donde se producen partículas fantasmas. En la imagen de la derecha de la Figura 46 se muestra la máscara resultante luego del ajuste manual. Las máscaras obtenidas por este método para cada configuración del medio poroso se muestran en la Figura 47.

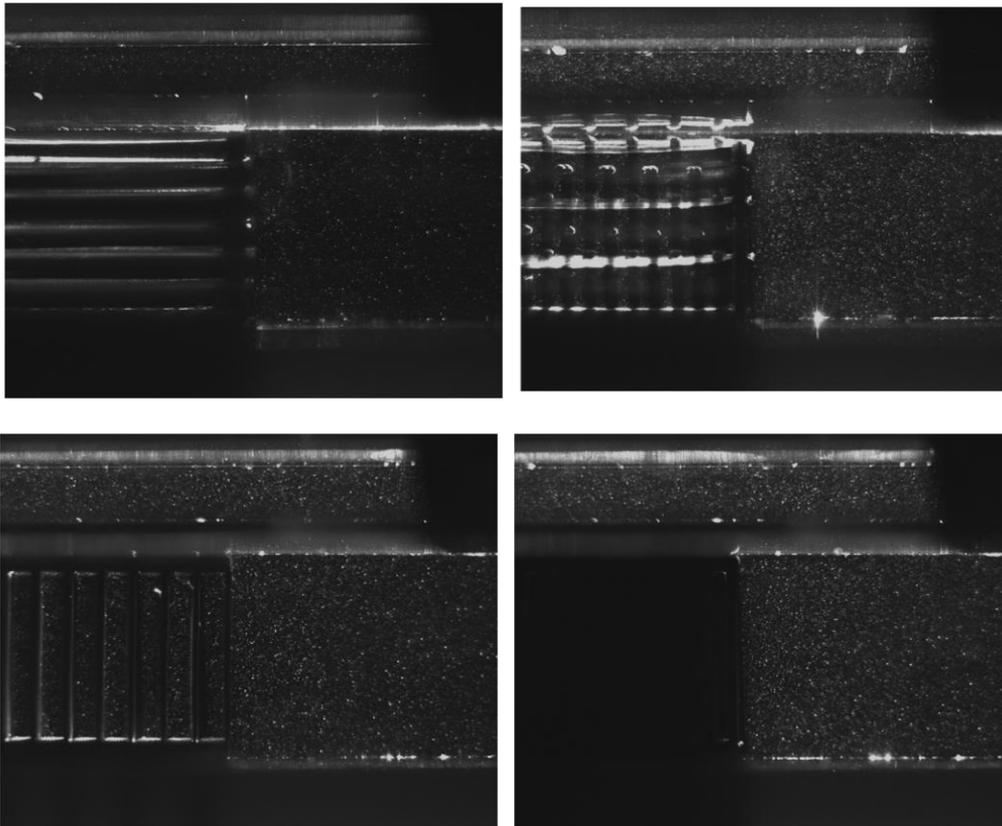


Figura 42 – Primera imagen de cada serie de capturas PIV para las distintas orientaciones de los medios porosos. De izquierda a derecha y de arriba hacia abajo, orientaciones A, B, C y B'.

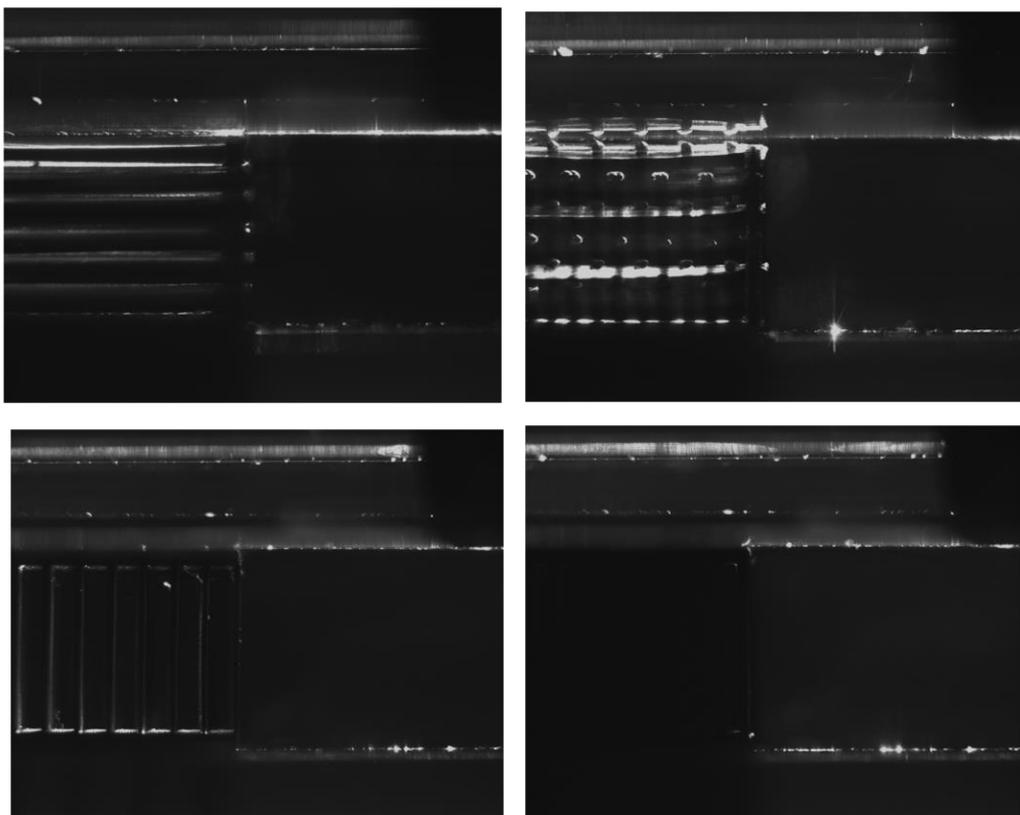


Figura 43 – Imágenes de fondo generadas por el método del mínimo histórico, para las distintas orientaciones de los medios porosos. De izquierda a derecha y de arriba hacia abajo, orientaciones A, B, C y B'.

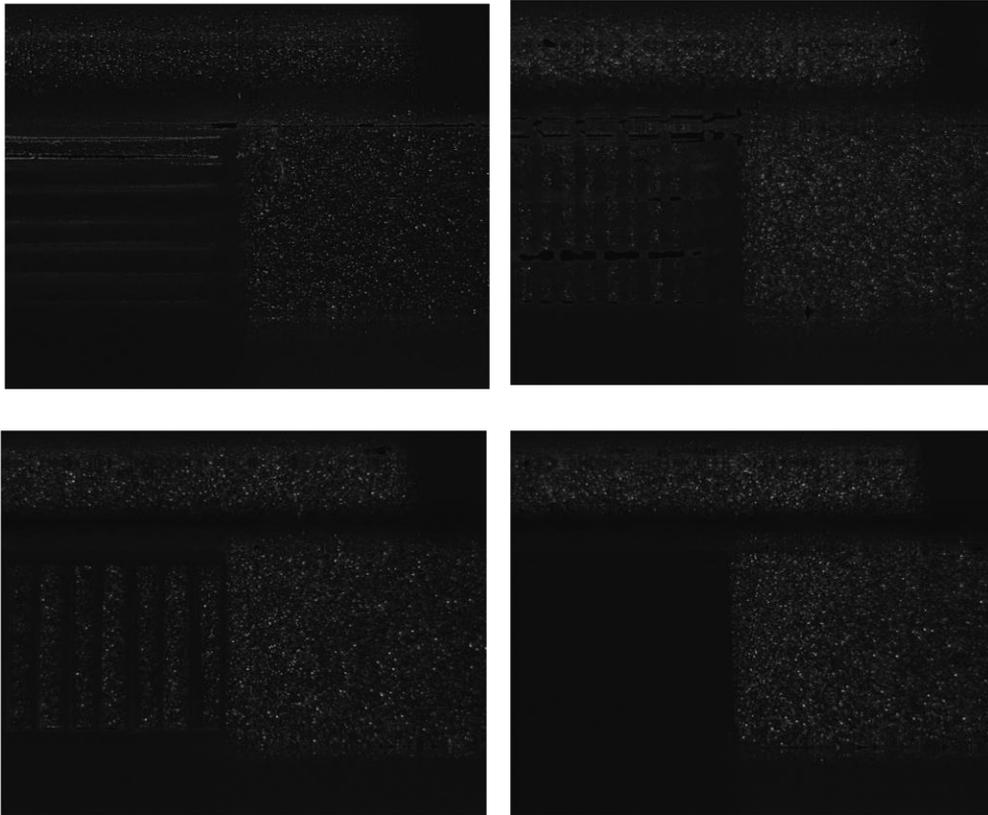


Figura 44 – Primera imagen de cada serie de capturas PIV para las distintas orientaciones de los medios porosos, luego de aplicar sustracción de fondo. De izquierda a derecha y de arriba hacia abajo, orientaciones A, B, C y B'.

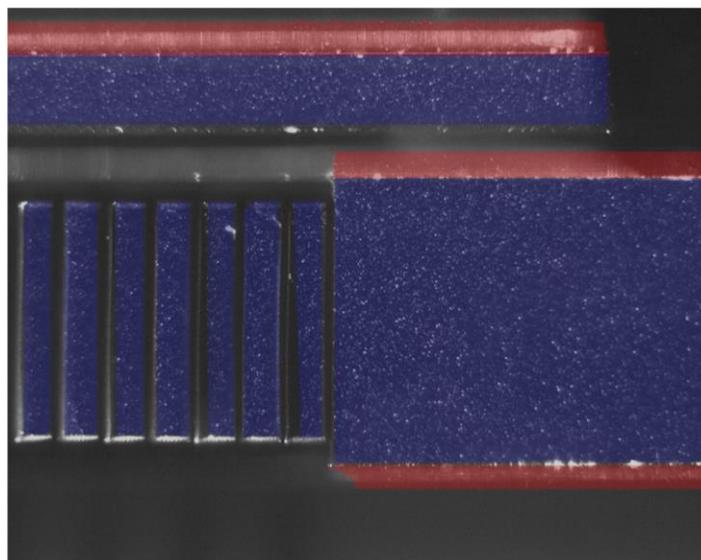


Figura 45 - Regiones dentro de la imagen. En azul se muestran zonas de interés con flujo de partículas. En rojo se muestran zonas con partículas fantasma debido a la reflexión de luz de partículas en las superficies sólidas.

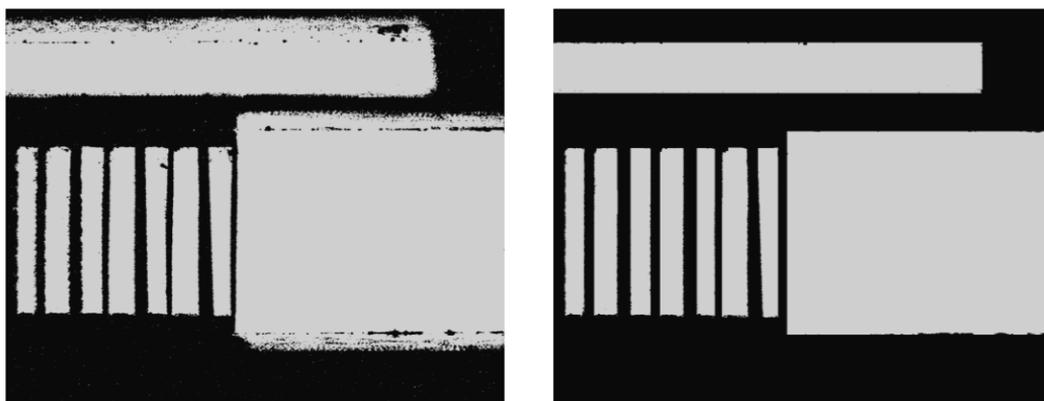


Figura 46 – Máscara generada por medio del algoritmo de generación automática (izquierda) y posterior corrección manual (derecha).

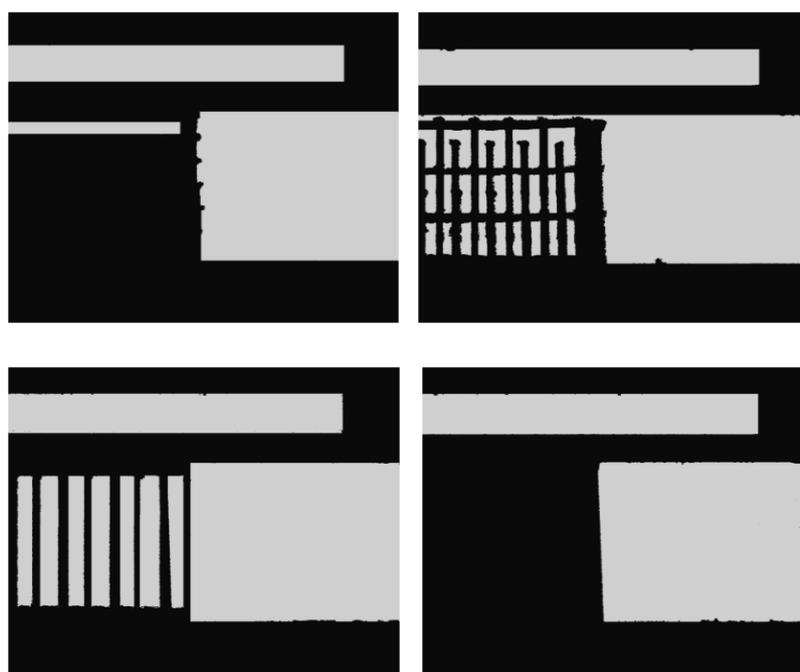


Figura 47 – Máscaras estáticas generadas para cada serie PIV para las distintas orientaciones de los medios porosos. De izquierda a derecha y de arriba hacia abajo, orientaciones A, B, C y B'.

Todas las series capturadas fueron procesadas mediante el mismo esquema. Se empleó una técnica de pasos múltiples con deformación de imagen consistente en dos pasos PIV, el primero utilizando una grilla de 16x16 píxeles, con ventanas de 32x32 píxeles y el segundo utilizando una grilla de 8x8 píxeles con ventanas de 16x16 píxeles. En ambos pasos de procesamiento se aplicó rellenado con ceros y funciones de ventaneo tipo Hamming. Para la deformación de las imágenes, el campo de desplazamientos obtenido a partir del primer paso fue interpolado a cada píxel de la imagen utilizando un esquema de interpolación basado en splines cúbicas. Para la interpolación de las imágenes a la grilla deformada también se aplicó un esquema de

splines cúbicas. La correlación fue implementada por medio de la correlación estándar y la estimación subpíxel se realizó mediante ajuste gaussiano de tres puntos. Para la validación se aplicó el método de red neuronal celular con umbral constante, y los vectores espurios fueron reemplazados por interpolación cúbica. El suavizado se implementó ajustando el campo resultante a una superficie de splines cúbicas, mediante mínimos cuadrados con penalización.

En la Figura 48 se muestran a modo de ejemplo campos de desplazamiento obtenidos para cada orientación de los medios permeables, para los casos de menor y mayor velocidad media en los poros. Los campos de velocidad validados y suavizados obtenidos para cada orientación y para cada tensión del ventilador fueron promediados en el tiempo. En la Figura 49 se muestran los campos de la componente de velocidad en el sentido del flujo, para las distintas orientaciones de los medios porosos y para los casos correspondientes a la velocidad más baja y alta del ventilador. En estas gráficas se observan algunos fenómenos que se analizan a continuación. En primer lugar, como es de esperarse, se observa una velocidad superior en la región de flujo libre, en contraste con el flujo dentro del medio poroso, debido a las restricciones que este impone al paso del fluido. En segundo lugar, se observa la presencia inevitable de sombras dentro del medio poroso, producto de la obstrucción parcial de las barras al pasaje de la luz. Estas sombras actúan reduciendo parcial o totalmente la luz reflejada por las partículas en ciertas regiones dentro del medio poroso, impidiendo determinar la velocidad en dichas zonas. En la orientación A (primera fila de la Figura 49) las sombras impidieron medir la velocidad prácticamente en todos los puntos dentro del medio poroso, mientras que para la orientación B (segunda fila de la Figura 49) estas solo limitaron parcialmente la medición de la velocidad en el interior del medio. En tercer y último lugar, se presentan regiones con canalizaciones en el flujo. Por ejemplo, en la orientación A se presentan zonas de mayor velocidad, en forma de chorros, a la salida del medio poroso. Por otro lado, en la configuración C (tercera fila de la Figura 49) se presentan regiones dentro del medio poroso en donde el fluido se acelera y desacelera. Esta aparente violación al principio de conservación de masa en el plano se explica teniendo en cuenta de que el fluido debe pasar a través de un arreglo de barras paralelas, que implica una disminución de la sección de flujo, y naturalmente un aumento de la velocidad del fluido. Esta velocidad vuelve a disminuir progresivamente en el sentido del flujo una vez atravesados los obstáculos, pero el fluido se acelera nuevamente al atravesar el siguiente arreglo de barras, generando el efecto observado.

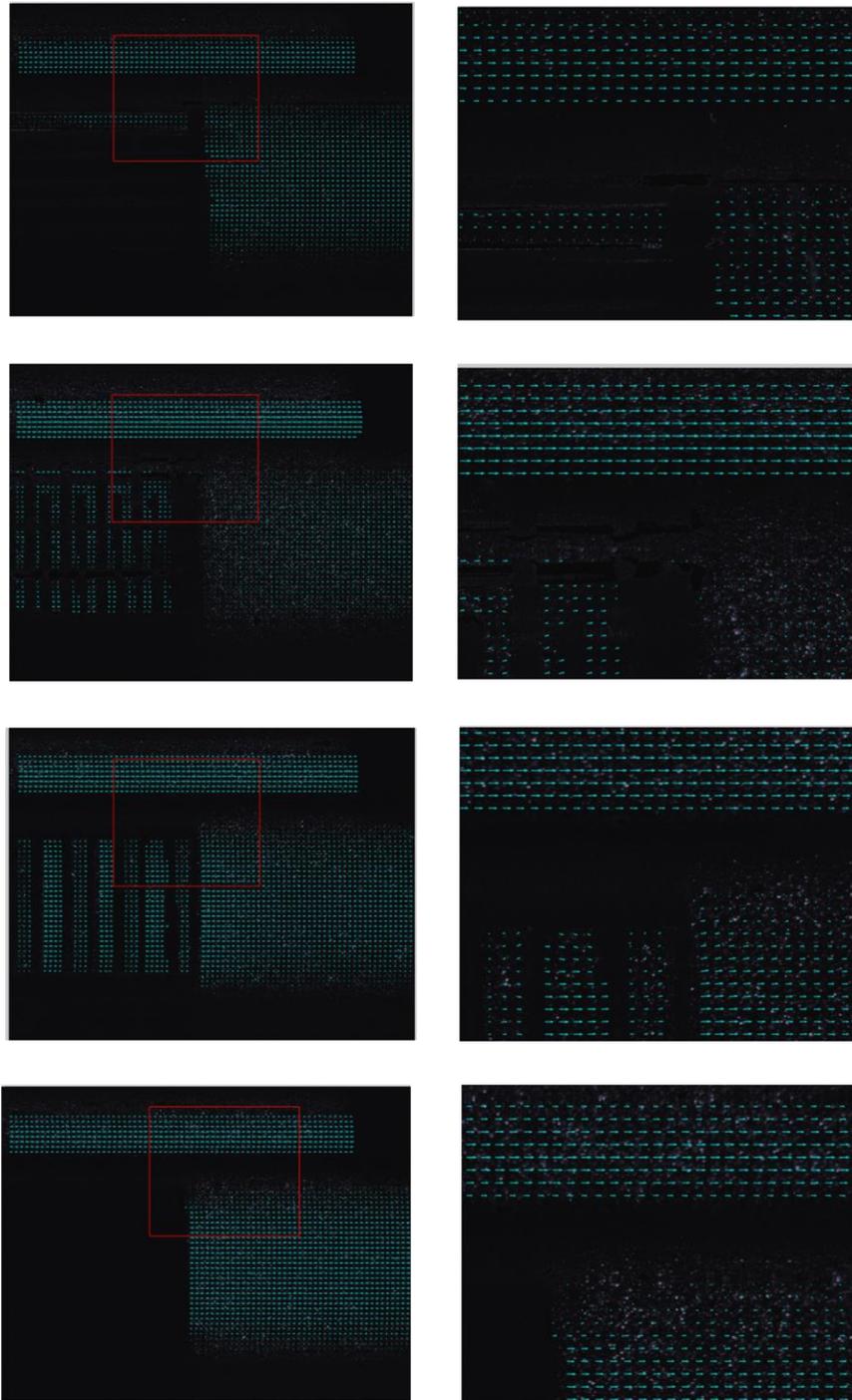


Figura 48 – Campos de velocidad estimados, mostrados a modo de ejemplo, para las distintas configuraciones del medio poroso. De arriba hacia abajo, orientaciones A, B, C y B'.

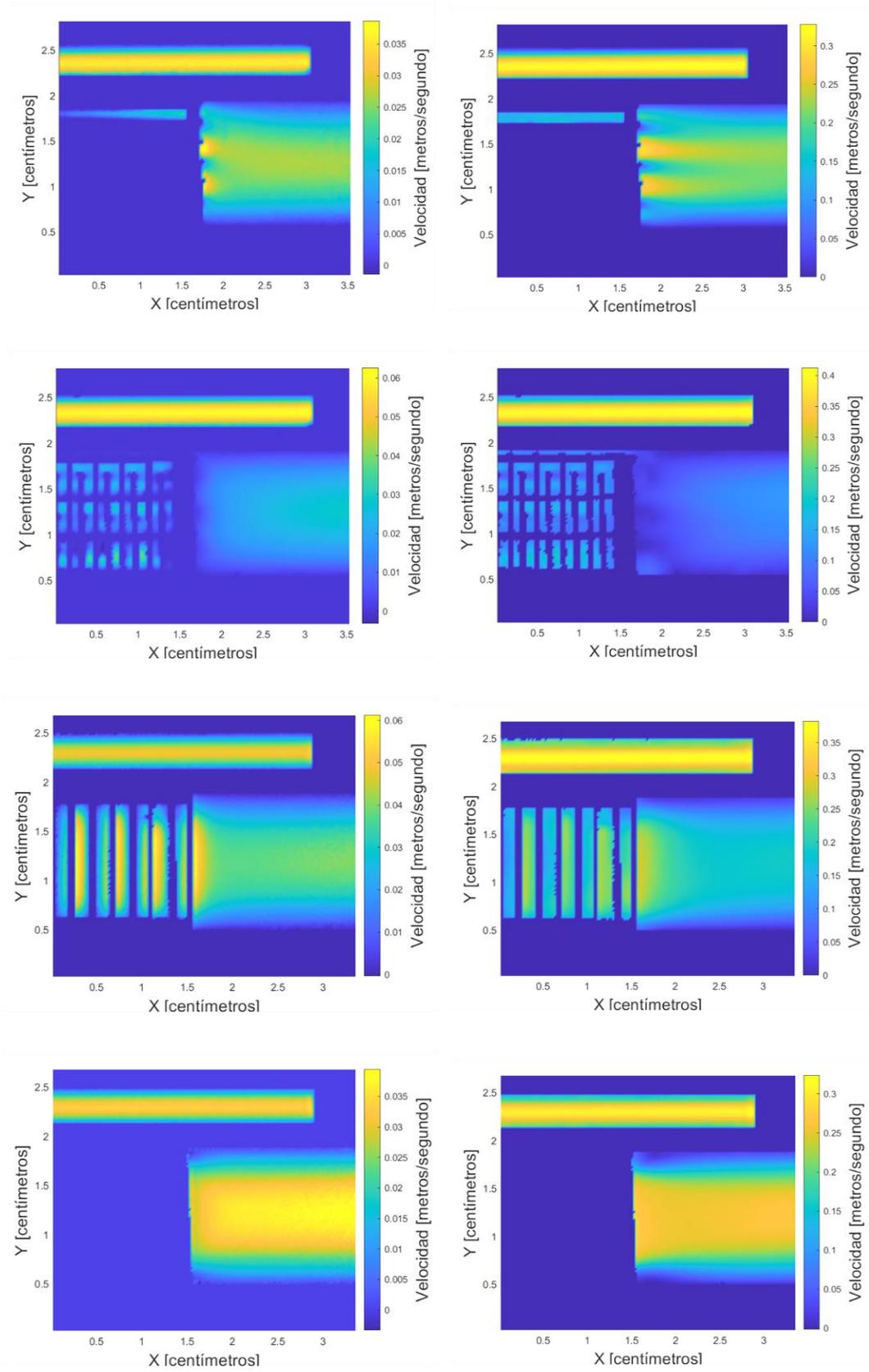


Figura 49 – Gráficas de contorno de la velocidad en el sentido del flujo promediada en el tiempo, para las distintas configuraciones del medio poroso. De arriba hacia abajo, orientaciones A, B, C y B’.

Los campos de velocidades promediados en el tiempo fueron posteriormente utilizados para estimar el coeficiente de permeabilidad. El perfil de velocidades en el área de flujo libre fue ajustada a una curva cuadrática para estimar el gradiente de presión, asumiendo flujo de tipo Poiseuille. En la Figura 50 se muestra como ejemplo el ajuste del perfil de velocidades de la región de flujo libre para el caso de mayor velocidad, para la orientación A. Por otro lado, el perfil de velocidades a la salida del medio poroso fue utilizado para estimar la velocidad media a través del medio. En la Figura 51 se muestran los perfiles de velocidad media en función de la localización de la sección de cálculo, para los casos de menor y mayor velocidad en poros, y las distintas orientaciones de los medios porosos. El comienzo de estas gráficas se corresponde con la sección inmediatamente a la salida del medio poroso. Se observan variaciones importantes en la velocidad media, lo cual puede ser atribuible a efectos de flujo tridimensional debido a la interacción del fluido con el medio poroso y al desarrollo del perfil de velocidades. El flujo parece tender a estabilizarse hacia aguas abajo del medio poroso. A excepción de las mediciones correspondientes a la orientación C, el flujo no parece terminar de desarrollarse por completo dentro de la región de observación. Para minimizar el error de las estimaciones de la permeabilidad, se tendrá en cuenta la velocidad media en la región más alejada del medio poroso. Sin embargo, las fluctuaciones observadas contribuirán a la incertidumbre de las estimaciones.

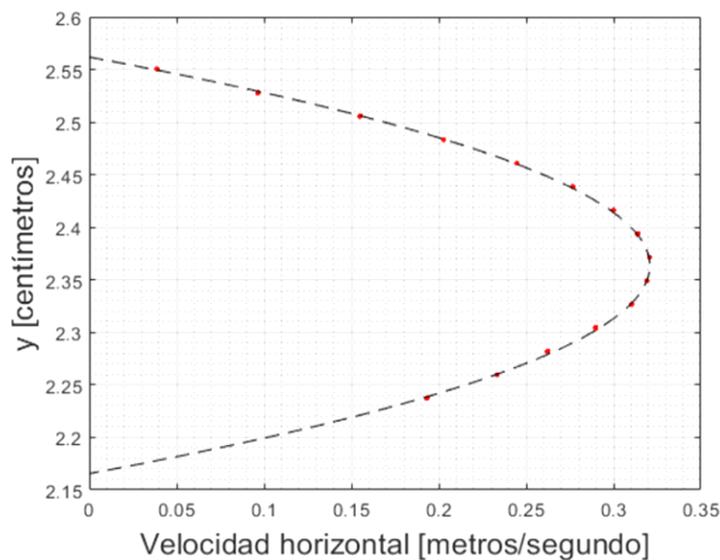


Figura 50 - Ajuste a una curva cuadrática del perfil de velocidades en la región de flujo libre, para el medio poroso con la configuración de barras paralelas al flujo, y el caso de mayor velocidad en poros.

A partir del ajuste cuadrático y la estimación de la velocidad media a través del medio poroso, el coeficiente de permeabilidad fue estimado tal como se describió al principio de la sección. Los resultados obtenidos se muestran en la Figura 52. Se observa que,

dentro del margen de error, el coeficiente de permeabilidad es independiente del caudal aplicado, demostrando la validez en la aplicación del modelo de Darcy en las condiciones estudiadas.

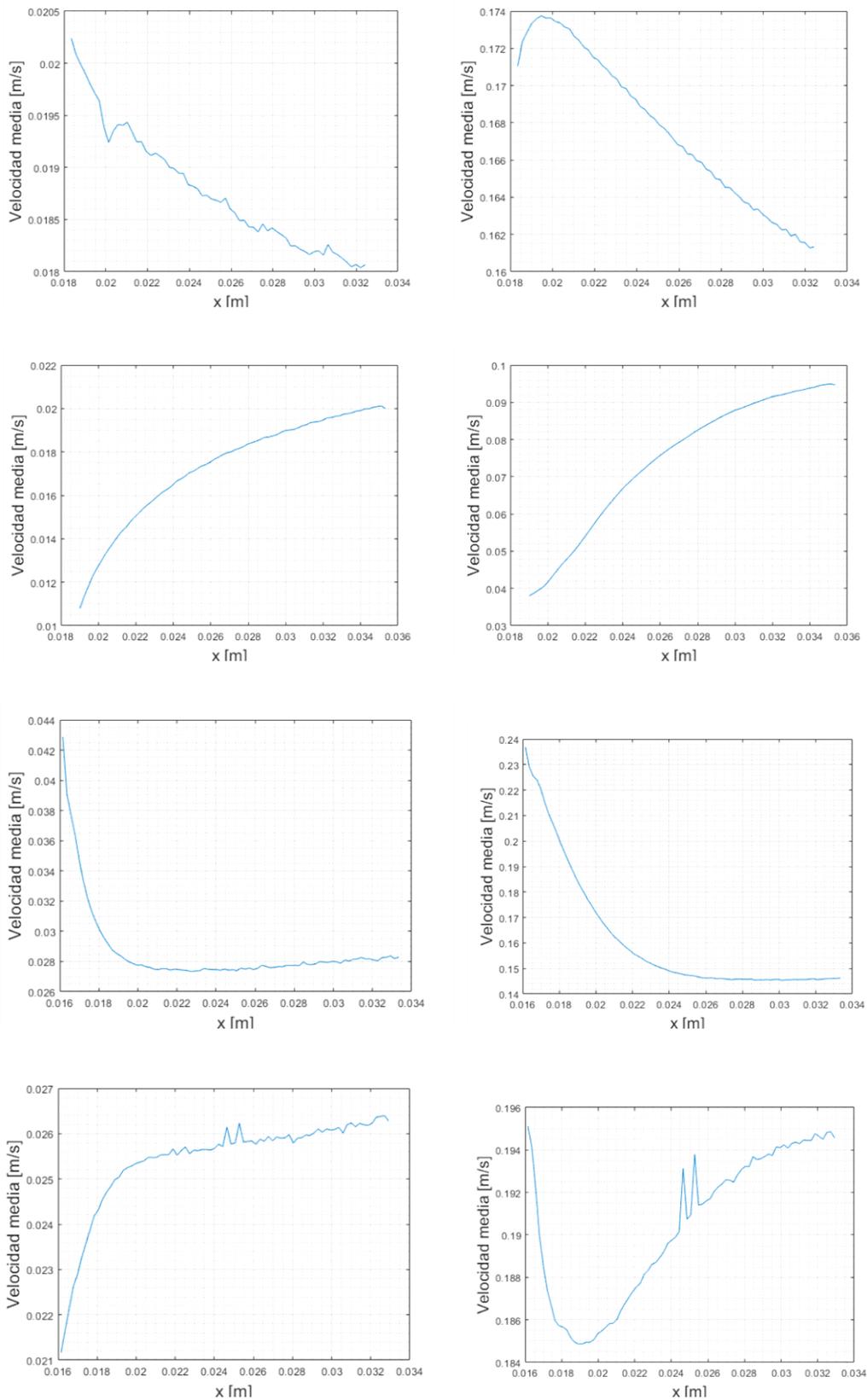


Figura 51 – Perfiles de velocidad media en el sentido del flujo en función de la localización de la sección, para las distintas configuraciones del medio poroso. De arriba hacia abajo, orientaciones A, B, C y B'. En la columna de la izquierda se muestran resultados para los casos de mínima velocidad media en poros, mientras que a la derecha se muestran los resultados para los casos con máxima velocidad media en poros.

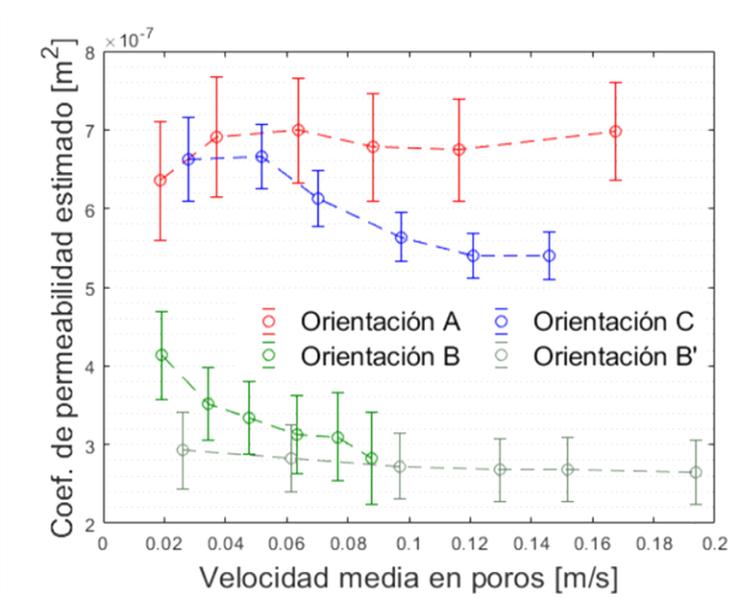


Figura 52 – Resultados de la estimación del coeficiente de permeabilidad para las distintas orientaciones del medio en función de la velocidad media en los poros. La incertidumbre estimada de cada cálculo se muestra en forma de barras de error.

VII. CONCLUSIONES

Durante el desarrollo de este trabajo fue abordada la problemática de la determinación, mediante la técnica de PIV en el plano, del campo de velocidades de un fluido circulante a través de un medio poroso. Se buscó medir la velocidad tanto fuera como dentro de los poros, lo cual presentó una serie de desafíos. En la región dentro del medio poroso se generan sombras y atenuación en la iluminación, dando lugar a regiones con depleción o ausencia de partículas, dificultando la determinación de vectores de velocidad válidos. Esto requiere de métodos de validación eficaces en la clasificación de vectores espurios, sobre todo aquellos concentrados en forma de clústeres. Entre los métodos de validación analizados, el método de validación basado en redes neuronales celulares resultó adecuado para esta tarea. La aplicación de una máscara estática también demostró ser una técnica efectiva para reducir el costo computacional y evitar el procesamiento en zonas con ausencia de partículas. Por otro lado, la presencia del medio poroso genera zonas con componentes estacionarios en las imágenes, las cuales introducen errores sistemáticos en las mediciones. Los ensayos de Montecarlo realizados sobre imágenes del medio poroso generadas sintéticamente demostraron que la combinación de algoritmos sofisticados de PIV (método de pasos múltiples con deformación de imágenes y correlación cruzada estándar con funciones de ventaneo y técnica de máscara digital) permiten la determinación del campo de velocidades con errores bajos tanto dentro como fuera del medio poroso. Sin embargo, se encontraron dificultades para determinar los desplazamientos de las partículas por medio de estos algoritmos cuando el ensayo se realizó aplicando mayores desplazamientos a las partículas, resultando en errores en los desplazamientos dentro del medio poroso superiores a 1 píxel. Esto pudo ser corregido al aplicar la técnica de PIV Super Resolución. A su vez, esta técnica demostró ser más eficaz que la técnica PIV para predecir la velocidad de las partículas cerca de las paredes. Otra de las ventajas de la técnica PIV Super Resolución es que permite alcanzar la máxima resolución espacial teórica posible, dado que mide el desplazamiento de cada partícula individual. Estas observaciones hacen de la técnica PIV Super Resolución un método con aparentes ventajas en la aplicación a este tipo de escenarios, y sirven de motivación para realizar estudios más detallados de la técnica en futuros trabajos.

Con respecto a la estimación del coeficiente de permeabilidad, se demostró la aplicabilidad del modelo de Darcy en las condiciones estudiadas, al evidenciarse la independencia de los valores estimados con el caudal circulante (dentro del margen de error). Sin embargo, se presentaron una serie de inconvenientes asociados a la medición del campo de velocidades solo en el plano iluminado. Debido a que la velocidad media del fluido es máxima en el centro del canal y disminuye hacia las paredes, la velocidad media calculada por medio de PIV depende de la ubicación del plano de iluminación. Esto implica una sobreestimación del caudal verdadero al ser calculado a partir del área de flujo y de la velocidad media del flujo calculada por PIV, por lo que en su lugar se introdujo un área efectiva, A_{ef} , inferior al área de flujo real. Esta cantidad, desconocida a priori, fue estimada a partir de una simulación numérica, asumiendo flujo laminar. Por

último, se evidenciaron variaciones importantes de la velocidad media en distintas secciones del flujo. Esta aparente violación al principio de conservación de masa se atribuyó a posibles efectos de flujo tridimensional no tenidos en cuenta, por ejemplo, estelas generadas por la presencia de las barras del medio poroso. Estas fuentes importantes de incertidumbre generadas por efectos de flujo tridimensional generan una motivación para la implementación de técnicas de medición más avanzadas en futuros trabajos, que impliquen la medición de la tercera componente de velocidad, o la medición del campo en todo el volumen, como las técnicas de PIV estéreo y PIV tomográfico.

VIII. REFERENCIAS

- Adrian, R. J. (1997). Dynamic ranges of velocity and spatial resolution of particle image velocimetry. *Measurement Science and Technology*, 8(12), 1393.
- Agüí, J. C., & Jimenez, J. (1987). On the performance of particle tracking. *Journal of fluid mechanics*, 185, 447-468.
- Al-Nahhal, I., Dobre, O. A., Basar, E., Moloney, C., & Ikki, S. (2019). A fast, accurate, and separable method for fitting a Gaussian function [tips & tricks]. *IEEE Signal Processing Magazine*, 36(6), 157-163.
- Angele, K. P., & Muhammad-Klingmann, B. (2005). A simple model for the effect of peak-locking on the accuracy of boundary layer turbulence statistics in digital PIV. *Experiments in fluids*, 38(3), 341-347.
- Blais, F., & Rioux, M. (1986). Real-time numerical peak detector. *Signal processing*, 11(2), 145-155.
- Brady, M. R. (2006). Subpixel resolution schemes for multiphase flows. (*Doctoral dissertation, Virginia Tech*).
- Brady, M. R. (2009). Methods for digital particle image sizing (DPIS): comparisons and improvements. *Flow Measurement and Instrumentation*, 20(6), 207-219.
- Caruana, R. A., Searle, R. B., Heller, T., & Shupack, S. I. (1986). Fast algorithm for the resolution of spectra. *Analytical chemistry*, 58(6), 1162-1167.
- Chavarin, A., Efstathiou, C., Vijay, S., & Luhar, M. (2020). Resolvent-based design and experimental testing of porous materials for passive turbulence control. *International Journal of Heat and Fluid Flow*, 86, 108722.
- Darcy, H. (1856). *Les fontaines publiques de la ville de Dijon*. Paris: V. Dalmont.
- Duncan, J., Dabiri, D., Hove, J., & Gharib, M. (2010). Universal outlier detection for particle image velocimetry (PIV) and particle tracking velocimetry (PTV) data. *Measurement Science and Technology*, 21(5), 057002.
- Durlofsky, L., & Brady, J. F. (1987). Analysis of the Brinkman equation as a model for flow in porous media. *The Physics of fluids*, 30(11), 3329-3341.
- Eckstein, A., & Vlachos, P. P. (2009). Assessment of advanced windowing techniques for digital particle image velocimetry (DPIV). *Measurement Science and Technology*, 20(7), 075402.
- Fincham, A. M., & Spedding, G. R. (1997). Low cost, high resolution DPIV for measurement of turbulent fluid flow. *Experiments in Fluids*, 23(6), 449-462.
- Fisher, R. B., & Naidu, D. K. (1996). A comparison of algorithms for subpixel peak detection. En *Image technology* (págs. 385-404). Berlin: Springer.
- Garcia, D. (2011). A fast all-in-one method for automated post-processing of PIV data. *Experiments in fluids*, 50(5), 1247-1259.

- Grant, I. (1997). Particle image velocimetry: a review. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, 211(1), 55-76.
- Gui, L. (1997). Evaluation of low image density recordings with the MQD method and application to the flow in a liquid bridge. *Journal of Flow Visualization & Image Processing*, 4, 333-343.
- Gui, L., & Merzkirch, W. (1996a). A method of tracking ensembles of particle images. *Experiments in fluids*, 21(6), 465-468.
- Gui, L., & Merzkirch, W. (1996b). Phase-separation of PIV measurements in two-phase flow by applying a digital mask technique. *ERCOFTAC Bull*, 30, 45-48.
- Gui, L., & Merzkirch, W. (1998). Generating arbitrarily sized interrogation windows for correlation-based analysis of particle image velocimetry recordings. *Experiments in fluids*, 24(1), 66-69.
- Gui, L., & Wereley, S. T. (2002). A correlation-based continuous window-shift technique to reduce the peak-locking effect in digital PIV image evaluation. *Experiments in Fluids*, 32(4), 506-517.
- Gui, L., Merzkirch, W., & Fei, R. (2000). A digital mask technique for reducing the bias error of the correlation-based PIV interrogation algorithm. *Experiments in fluids*, 29(1), 30-35.
- Gui, L., Wereley, S. T., & Kim, Y. H. (2003). Advances and applications of the digital mask technique in particle image velocimetry experiments. *Measurement Science and Technology*, 14(10), 1820.
- Guo, H. (2011). A simple algorithm for fitting a Gaussian function [DSP tips and tricks]. *IEEE Signal Processing Magazine*, 28(5), 134-137.
- Harris, F. J. (1978). On the use of windows for harmonic analysis with the discrete Fourier transform. *Proceedings of the IEEE*, 66(1), 51-83.
- Hildreth, E. C. (1984). Computations underlying the measurement of visual motion. *Artificial Intelligence*, 23(3), 309-354.
- Honkanen, M., & Nobach, H. (2005). Background extraction from double-frame PIV images. *Experiments in fluids*, 38(3), 348-362.
- Horn, B. K., & Schunck, B. G. (1981). Determining optical flow. *Artificial intelligence*, 1981, 185-203.
- Huang, H. T., Fiedler, H. E., & Wang, J. J. (1993). Limitation and improvement of PIV. I: Limitation of conventional techniques due to deformation of particle image patterns. *Experiments in fluids*, 15(3), 168-174.
- Irmay, S. (1958). On the theoretical derivation of Darcy and Forchheimer formulas. *Transactions American Geophysical Union*, 39(4), 702-707.
- Jambunathan, K., Ju, X. Y., Dobbins, B. N., & Ashforth-Frost, S. (1995). An improved cross correlation technique for particle image velocimetry. *Measurement Science and Technology*, 6(5), 507.

- Kay, S. M. (1993). Least Squares. En *Fundamentals of statistical signal processing* (págs. 219-286). Englewood Cliffs, NJ: PTR Prentice-Hall.
- Keane, R. D., & Adrian, R. J. (1990). Optimization of particle image velocimeters. I. Double pulsed systems. *Measurement science and technology*, *1*(11), 1202.
- Keane, R. D., & Adrian, R. J. (1993). Theory of cross-correlation analysis of PIV images. En *Flow visualization and image analysis* (págs. 1-25). Dordrecht: Springer.
- Keane, R. D., Adrian, R., & Zhang, Y. (1995). Super-resolution particle imaging velocimetry. *Measurement Science and Technology*, *6*(6), 754.
- Kemmerich, T., & Rath, H. J. (1994). Multi-level convolution filtering technique for digital laser-speckle-velocimetry. *Experiments in fluids*, *17*(5), 315-322.
- Ledda, P. G., Siconolfi, L., Viola, F., Camarri, S., & Gallaire, F. (2019). Flow dynamics of a dandelion pappus: A linear stability approach. *Physical review fluids*, *4*(7), 071901.
- Levenberg, K. (1944). A method for the solution of certain non-linear problems in least squares. *Quarterly of applied mathematics*, *2*(2), 164-168.
- Liang, D., Jiang, C., & Li, Y. (2003). Cellular neural network to detect spurious vectors in PIV data. *Experiments in fluids*, *34*(1), 52-62.
- Marquardt, D. W. (1963). An algorithm for least-squares estimation of nonlinear parameters. *Journal of the society for Industrial and Applied Mathematics*, *11*(2), 431-441.
- Masullo, A., & Theunissen, R. (2017). Automated mask generation for PIV image analysis based on pixel intensity statistics. *Experiments in Fluids*, *58*(6), 1-11.
- Mejia-Alvarez, R., & Christensen, K. T. (2013). Robust suppression of background reflections in PIV images. *Measurement Science and Technology*, *24*(2), 027003.
- Mendez, M. A., Raiola, M., Masullo, A., Discetti, S., Laniro, A., Theunissen, R., & Buchlin, J. M. (2017). POD-based background removal for particle image velocimetry. *Experimental Thermal and Fluid Science*, *80*, 181-192.
- Mitra, S., de Castro, A. R., & El Mansori, M. (2019). On the rapid manufacturing process of functional 3D printed sand molds. *Journal of Manufacturing Processes*, *42*, 202-212.
- Moin, P. (2010). Numerical Differentiation – Finite Differences. En *Fundamentals of engineering numerical analysis* (págs. 13-29). Cambridge University Press.
- Neuman, S. P. (1977). Theoretical derivation of Darcy's law. *Acta mechanica*, *25*(3), 153-170.
- Nobach, H. &. (2005). Two-dimensional Gaussian regression for sub-pixel displacement estimation in particle image velocimetry or particle position estimation in particle tracking velocimetry. *Experiments in fluids*, *38*(4), 511-515.
- Nogueira, J., Lecuona, A., & Rodriguez, P. A. (1999). Local field correction PIV: on the increase of accuracy of digital PIV systems. *Experiments in fluids*, *27*(2), 107-116.
- Nogueira, J., Legrand, M., Jiménez, R., & De Gregorio, F. (2021). Peak-locking full characterization: PIV error assessment and velocity ensemble measurement correction. *Measurement Science and Technology*, *32*(11), 114005.

- Nuttall, A. (1981). Some windows with very good sidelobe behavior. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 29(1), 84-91.
- Pun, C. S., Susanto, A., & Dabiri, D. (2007). Mode-ratio bootstrapping method for PIV outlier correction. *Measurement Science and Technology*, 18(11), 3511.
- Raffel, M. M., Willert, C. E., Scarano, F., Kähler, C. J., Wereley, S. T., & Kompenhans, J. (2018). *Particle Image Velocimetry*. Springer.
- Ronneberger, O., Raffel, M., & Kompenhans, J. (1998). Advanced evaluation algorithms for standard and dual plane particle image velocimetry. En *Proceedings of the 9th international symposium on applications of laser techniques to fluid mechanics* (págs. 13-16). Lisboa, Portugal.
- Rosti, M. E., Brandt, L., & Pinelli, A. (2018). Turbulent channel flow over an anisotropic porous wall—drag increase and reduction. *Journal of Fluid Mechanics*, 842, 381-394.
- Scarano, F. (2001). Iterative image deformation methods in PIV. *Measurement science and technology*, 13(1), R1.
- Scarano, F. (2003). Theory of non-isotropic spatial resolution in PIV. *Experiments in Fluids*, 35(3), 268-277.
- Scarano, F., & Riethmuller, M. L. (1999). Iterative multigrid approach in PIV image processing with discrete window offset. *Experiments in Fluids*, 26(6), 513-523.
- Sciacchitano, A., & Scarano, F. (2014). Elimination of PIV light reflections via a temporal high pass filter. *Measurement Science and Technology*, 25(8), 084009.
- Sciacchitano, A., & Wieneke, B. (2016). PIV uncertainty propagation. *Measurement Science and Technology*, 27(8), 084006.
- Shinnee, A. M., Bugg, J. D., & Balachandar, R. (2004). Variable threshold outlier identification in PIV data. *Measurement Science and Technology*, 15(9), 1722.
- Soria, J. (1996). An investigation of the near wake of a circular cylinder using a video-based digital cross-correlation particle image velocimetry technique. *Experimental Thermal and Fluid Science*, 12(2), 221-233.
- Spedding, G. R., & Rignot, E. J. (1993). Performance analysis and application of grid interpolation techniques for fluid flows. *Experiments in Fluids*, 15(6), 417-430.
- Stitou, A., & Riethmuller, M. L. (2001). Extension of PIV to super resolution using PTV. *Measurement Science and Technology*, 12(9), 1398.
- Suga, K., Okazaki, Y., Matsuo, T., Taneo, A., & Kuwata, Y. (2019). MEASUREMENT OF TURBULENT SQUARE DUCT FLOWS OVER ANISOTROPIC POROUS MEDIA. En *Eleventh International Symposium on Turbulence and Shear Flow Phenomena (TSFP11)*. Begel House Inc.
- Sugii, Y., Nishio, S., Okuno, T., & Okamoto, K. (2000). A highly accurate iterative PIV technique using a gradient method. *Measurement Science and Technology*, 11(12), 1666.
- Takehara, K., & Etoh, T. (1998). A study on particle identification in PTV particle mask correlation method. *Journal of Visualization*, 1(3), 313-323.

- Theunissen, R., Scarano, F., & Riethmuller, M. L. (2006). An adaptive sampling and windowing interrogation method in PIV. *Measurement Science and Technology*, *18*(1), 275.
- Thevenaz, P., Ruttimann, U. E., & Unser, M. (1998). A pyramid approach to subpixel registration based on intensity. *IEEE transactions on image processing*, *7*(1), 27-41.
- Wereley, S. T., & Meinhart, C. D. (2001). Second-order accurate particle image velocimetry. *Experiments in fluids*, *31*(3), 258-268.
- Wereley, S. T., Gui, L., & Meinhart, C. D. (2002). Advanced algorithms for microscale particle image velocimetry. *AIAA journal*, *40*(6), 1047-1055.
- Westerweel, J. (1994). Efficient detection of spurious vectors in particle image velocimetry data. *Experiments in fluids*, *16*(3), 236-247.
- Westerweel, J. (1995). Digital particle image velocimetry: Theory and application.
- Westerweel, J. (1997). Fundamentals of digital particle image velocimetry. *Measurement science and technology*, *8*(12), 1379.
- Westerweel, J. (2000). Theoretical analysis of the measurement precision in particle image velocimetry. *Experiments in Fluids*, *29*(1), S003-S012.
- Westerweel, J., & Scarano, F. (2005). Universal outlier detection for PIV data. *Experiments in fluids*, *39*(6), 1096-1100.
- Westerweel, J., Dabiri, D., & Gharib, M. (1997). The effect of a discrete window offset on the accuracy of cross-correlation analysis of digital PIV recordings. *Experiments in fluids*, *23*(1), 20-28.
- Whitaker, S. (1986). Flow in porous media I: A theoretical derivation of Darcy's law. *Transport in porous media*, *1*(1), 3-25.
- Willert, C. E. (1997). Stereoscopic digital particle image velocimetry for application in wind tunnel flows. *Measurement science and technology*, *8*(12), 1465.
- Willert, C. E., & Gharib, M. (1991). Digital particle image velocimetry. *Experiments in fluids*, *10*(4), 181-193.

IX. ANEXO

IX.1. TRANSFORMADA DE FOURIER DISCRETA EN EL TIEMPO

Sea una señal discreta definida a partir de una función $x: \mathbb{Z} \rightarrow \mathbb{R}$. La transformada de Fourier discreta en el tiempo de la señal x vendrá dada por,

$$\mathcal{F}\{x; f\} = X(f) = \sum_{n=-\infty}^{\infty} x(n) e^{-2\pi i n f},$$

donde la variable f puede adoptar valores reales. Por otro lado, sea $X: \mathbb{R} \rightarrow \mathbb{C}$ la función que representa la transformada de Fourier discreta en el tiempo de alguna señal discreta x . La transformada de Fourier inversa se computa como

$$\mathcal{F}^{-1}\{X; n\} = x(n) = \int_{-1/2}^{1/2} X(f) e^{2\pi i n f} df.$$

Estas dos operaciones son inversas entre sí, es decir

$$\mathcal{F}^{-1}\{\mathcal{F}\{x\}\} = x.$$

Esto último se puede probar por combinación de ambas expresiones:

$$\begin{aligned} \mathcal{F}^{-1}\{\mathcal{F}\{x; f\}; n\} &= \int_{-1/2}^{1/2} \left(\sum_{k=-\infty}^{\infty} x(k) e^{-2\pi i k f} \right) e^{2\pi i n f} df \\ &= \sum_{k=-\infty}^{\infty} x(k) \int_{-1/2}^{1/2} e^{-2\pi i f(k-n)} df, \end{aligned}$$

La integral vale 1 para $k = n$ y 0 en caso contrario, por lo que puede reemplazarse por $\delta(k - n)$, siendo

$$\delta(n) = \begin{cases} 1 & \text{si } n = 0 \\ 0 & \text{en caso contrario} \end{cases}$$

la función impulso unitario. Se sigue entonces que

$$\mathcal{F}^{-1}\{\mathcal{F}\{x; f\}; n\} = \sum_{k=-\infty}^{\infty} x(k) \delta(k - n) = x(n).$$

IX.2. TRANSFORMADA DE FOURIER DISCRETA

Sea una señal discreta representada por una secuencia de valores, $x = \{x[1], x[2], \dots, x[N]\}$. La transformada de Fourier discreta $\mathcal{F}\{x\}$ de la señal x viene dada por

$$\mathcal{F}\{x; k\} = X(k) = \sum_{n=1}^N x(n) e^{-\frac{2\pi i(k-1)(n-1)}{N}},$$

Mientras que la transformada de Fourier inversa discreta, $\mathcal{F}^{-1}\{X; n\}$, viene dada por

$$\mathcal{F}^{-1}\{X; n\} = x(n) = \frac{1}{N} \sum_{k=1}^N X(k) e^{\frac{2\pi i(k-1)(n-1)}{N}}.$$

Ambas operaciones son inversas entre sí, es decir,

$$\mathcal{F}^{-1}\{\mathcal{F}\{x\}\} = x.$$

Esto se puede probar directamente combinando ambas expresiones:

$$\begin{aligned} \mathcal{F}^{-1}\{\mathcal{F}\{x; k\}; n\} &= \frac{1}{N} \sum_{k=1}^N \sum_{j=1}^N x(j) e^{-\frac{2\pi i(k-1)(j-1)}{N}} e^{\frac{2\pi i(k-1)(n-1)}{N}} \\ &= \frac{1}{N} \sum_{j=1}^N x(j) \sum_{k=1}^N e^{\frac{2\pi i(n-j)(k-1)}{N}}. \end{aligned}$$

La sumatoria en k de la última expresión es igual a N cuando $n = j$ y es igual a cero en caso contrario, por lo tanto, se puede reemplazar por $N \delta(n - j)$, resultando

$$\mathcal{F}^{-1}\{\mathcal{F}\{x; k\}; n\} = \sum_{j=1}^N x(j) \delta(n - j) = x(n).$$

El cálculo directo de la Transformada de Fourier requiere del orden de N^2 operaciones.

IX.3. NATURALEZA PERIÓDICA DE LA TRANSFORMADA DE FOURIER

La transformada de Fourier discreta de una señal de tamaño N tiene un comportamiento periódico con periodicidad N , es decir

$$X(k + N) = X(k).$$

Esto puede probarse directamente a partir de la definición de la transformada de Fourier:

$$\begin{aligned} X(k + N) &= \sum_{n=1}^N x(n) e^{-\frac{2\pi i(k+N-1)(n-1)}{N}} = \sum_{n=1}^N x(n) e^{-\frac{2\pi i(k-1)(n-1)}{N}} e^{-2\pi i(n-1)} \\ &= \sum_{n=1}^N x(n) e^{-\frac{2\pi i(k-1)(n-1)}{N}} = X(k). \end{aligned}$$

IX.4. TEOREMA DE LA CONVOLUCIÓN

Sean dos señales discretas de valores reales y de longitud N , x e y . El teorema de convolución establece que la correlación cruzada entre las señales x e y puede calcularse como

$$C = \mathcal{F}^{-1}\{ \overline{\mathcal{F}\{x\}} \mathcal{F}\{y\} \},$$

donde $\overline{\mathcal{F}\{x\}}$ denota el complejo conjugado de $\mathcal{F}\{x\}$. Esto puede probarse por sustitución:

$$\begin{aligned} & \mathcal{F}^{-1}\{ \overline{\mathcal{F}\{x\}} \mathcal{F}\{y\} \} \\ &= \frac{1}{N} \sum_{m=1}^N \left(\sum_{k=1}^N x(k) e^{\frac{2\pi i(m-1)(k-1)}{N}} \sum_{j=1}^N y(j) e^{-\frac{2\pi i(m-1)(j-1)}{N}} \right) e^{\frac{2\pi i(m-1)(n-1)}{N}} \\ &= \frac{1}{N} \sum_{k=1}^N x(k) \left(\sum_{j=1}^N y(j) \sum_{m=1}^N e^{\frac{2\pi i(k-j+n-1)(m-1)}{N}} \right) \\ &= \sum_{k=1}^N x(k) \left(\sum_{j=1}^N y(j) \delta(k-j+n-1) \right) = \sum_{k=1}^N x(k) y(k+n-1). \end{aligned}$$

Para efectuar esta operación, es necesario que ambas señales x e y tengan el mismo tamaño. Además, por la naturaleza periódica de la transformada de Fourier discreta, la función de correlación obtenida es equivalente a la que se obtendría mediante la correlación directa asumiendo de que las señales son periódicas con período N , es decir, se obtiene la correlación cruzada cíclica.

IX.5. TRANSFORMADA DE FOURIER EN DOS DIMENSIONES

La transformada de Fourier discreta de una señal x de tamaño $N_i \times N_j$ viene dada por

$$\mathcal{F}\{x; k_i, k_j\} = X(k_i, k_j) = \sum_{j=1}^{N_j} \sum_{i=1}^{N_i} x(i, j) e^{-\frac{2\pi i(k_i-1)(i-1)}{N_i}} e^{-\frac{2\pi i(k_j-1)(j-1)}{N_j}}.$$

Esto es equivalente a aplicar la transformada de Fourier sobre una dimensión y luego sobre la otra. Aplicar la transformada de Fourier sobre las filas de la matriz $x(i, j)$ tiene un costo computacional del orden de $N_i^2 N_j$, mientras que hacerlo sobre las columnas tiene un costo computacional del orden de $N_i N_j^2$, por lo que el costo computacional de la operación es del orden de $N_i N_j (N_i + N_j)$.

La transformada de Fourier para señales de dimensiones arbitrarias se define de forma análoga. Las propiedades de periodicidad y el teorema de convolución son igualmente válidos en estos casos.

IX.6. EXPRESIÓN ANALÍTICA PARA LA FUNCIÓN DE CORRELACIÓN DE UNA PARTÍCULA CON DISTRIBUCIÓN GAUSSIANA

Sea

$$g_1: \mathbb{R}^2 \rightarrow \mathbb{R} \mid g_1(x, y) = I e^{-\frac{1}{2\sigma^2}[(x-x_0)^2+(y-y_0)^2]}$$

la función que representa la imagen de una partícula con distribución de intensidades gaussiana, centrada en el punto (x_0, y_0) con respecto a cierto marco de referencia, con pico de intensidad I y factor de forma σ , y sea

$$g_2: \mathbb{R}^2 \rightarrow \mathbb{R} \mid g_2(x, y) = I e^{-\frac{1}{2\sigma^2}[(x-x_f)^2+(y-y_f)^2]}$$

la función que representa la imagen de la partícula luego que se ha desplazado hacia un punto (x_f, y_f) con respecto al mismo marco de referencia. El desplazamiento neto de la partícula viene dado por

$$\bar{d} = (d_x, d_y) = (x_f - x_0, y_f - y_0).$$

La función de correlación entre las funciones g_1 y g_2 puede determinarse de forma analítica como

$$\begin{aligned} C(\delta_x, \delta_y) &= I^2 \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} e^{-\frac{1}{2\sigma^2}[(x-x_0)^2+(y-y_0)^2]} e^{-\frac{1}{2\sigma^2}[(x+\delta_x-x_f)^2+(y+\delta_y-y_f)^2]} dx dy \\ &= I^2 \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} e^{-\frac{1}{2\sigma^2}[(x-x_0)^2+(y-y_0)^2]} e^{-\frac{1}{2\sigma^2}[(x-x_0+\delta_x-d_x)^2+(y-y_0+\delta_y-d_y)^2]} dx dy \\ &= I^2 e^{-\frac{1}{2\sigma^2}[(\delta_x-d_x)^2+(\delta_y-d_y)^2]} \\ &\quad \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} e^{-\frac{1}{2\sigma^2}[2(x-x_0)^2+2(y-y_0)^2+2(\delta_x-d_x)(x-x_0)+2(\delta_y-d_y)(y-y_0)]} dx dy \\ &= I^2 e^{-\frac{1}{2(\sqrt{2}\sigma)^2}[(\delta_x-d_x)^2+(\delta_y-d_y)^2]} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} e^{-\frac{1}{2\left(\frac{\sigma}{\sqrt{2}}\right)^2}\left[\left(x+\frac{\delta_x}{2}-\frac{x_f+x_0}{2}\right)^2+\left(y+\delta_y-\frac{y_f+y_0}{2}\right)^2\right]} dx dy \\ &= \pi\sigma^2 I^2 e^{-\frac{1}{2(\sqrt{2}\sigma)^2}[(\delta_x-d_x)^2+(\delta_y-d_y)^2]}. \end{aligned}$$

Se observa que la función de correlación cruzada entre las funciones g_1 y g_2 tiene también distribución gaussiana, con su máximo en (d_x, d_y) , por lo que a partir del

máximo de esta función es posible determinar el desplazamiento de la partícula. Por otro lado, la intensidad máxima de este pico es de $\pi\sigma^2 I^2$, y su factor de forma esta ampliado por un factor de $\sqrt{2}$ con respecto al factor de forma de la partícula.

IX.7. IMPLEMENTACIÓN DEL ALGORITMO DE LEVENBERG-MARQUARDT PARA EL AJUSTE A UNA SUPERFICIE GAUSSIANA

Sean $\mathbf{x} = [x(1), x(2), \dots, x(N)]^T$ e $\mathbf{y} = [y(1), y(2), \dots, y(N)]^T$ las componentes horizontal y vertical, respectivamente, de ciertos puntos del plano, y sea

$$\mathbf{C} = [C(x(1), y(1)), C(x(2), y(2)), \dots, C(x(N), y(N))]^T.$$

los valores de una función evaluada en dichos puntos. Estos puntos desean ajustarse a una superficie gaussiana en el sentido de mínimos cuadrados. La superficie gaussiana viene descrita por medio de la función

$$f(x, y; \boldsymbol{\theta}) = I e^{-\frac{1}{2\sigma_x^2}(x-\delta_x)^2 - \frac{1}{2\sigma_y^2}(y-\delta_y)^2},$$

para cierto arreglo de parámetros $\boldsymbol{\theta} = [\theta_1, \theta_2, \theta_3, \theta_4, \theta_5]^T = [I, \sigma_x^2, \sigma_y^2, \delta_x, \delta_y]^T$. El ajuste por mínimos cuadrados implica establecer el valor de los parámetros tal que minimicen la función E definida por

$$E(\boldsymbol{\theta}) = \sum_{n=1}^N W(n) [C(x(n), y(n)) - f(x(n), y(n); \boldsymbol{\theta})]^2,$$

donde W es una función de peso con valores positivos que sirve para ponderar la contribución de los distintos términos en esta suma. A continuación, se define el vector $\mathbf{f}(\boldsymbol{\theta})$, cuyas componentes son el valor de la superficie f evaluada en los distintos puntos a ajustar, es decir

$$\mathbf{f}(\boldsymbol{\theta}) = [f(x(1), y(1); \boldsymbol{\theta}), f(x(2), y(2); \boldsymbol{\theta}), \dots, f(x(N), y(N); \boldsymbol{\theta})]^T.$$

Esto permite escribir la función de error en forma matricial como

$$E(\boldsymbol{\theta}) = (\mathbf{C} - \mathbf{f}(\boldsymbol{\theta}))^T \mathbf{W} (\mathbf{C} - \mathbf{f}(\boldsymbol{\theta})),$$

donde \mathbf{W} es la matriz diagonal de $N \times N$ que contiene los valores de la función de pesos W en su diagonal.

La función \mathbf{f} es una función no lineal de los parámetros de ajuste. Sin embargo, puede linealizarse por medio del desarrollo en series de Taylor. Sea $\boldsymbol{\theta}_0$ una combinación inicial de parámetros. Del desarrollo en series de Taylor de \mathbf{f} en torno a $\boldsymbol{\theta}_0$, despreciando los términos de mayor orden, resulta

$$\mathbf{f}(\boldsymbol{\theta}_0 + \Delta\boldsymbol{\theta}) \cong \mathbf{f}(\boldsymbol{\theta}_0) + \mathbf{J}(\boldsymbol{\theta}_0)\Delta\boldsymbol{\theta},$$

donde \mathbf{J} es la matriz jacobiana, dada por

$$J(\boldsymbol{\theta}) = \begin{bmatrix} \frac{\partial f}{\partial \theta_1}(x(1), y(1); \boldsymbol{\theta}) & \frac{\partial f}{\partial \theta_2}(x(1), y(1); \boldsymbol{\theta}) & \cdots & \frac{\partial f}{\partial \theta_5}(x(1), y(1); \boldsymbol{\theta}) \\ \frac{\partial f}{\partial \theta_1}(x(2), y(2); \boldsymbol{\theta}) & \frac{\partial f}{\partial \theta_2}(x(2), y(2); \boldsymbol{\theta}) & \cdots & \frac{\partial f}{\partial \theta_5}(x(2), y(2); \boldsymbol{\theta}) \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial f}{\partial \theta_1}(x(N), y(N); \boldsymbol{\theta}) & \frac{\partial f}{\partial \theta_2}(x(N), y(N); \boldsymbol{\theta}) & \cdots & \frac{\partial f}{\partial \theta_5}(x(N), y(N); \boldsymbol{\theta}) \end{bmatrix}.$$

Al introducir esta expresión en la función de error, se obtiene una expresión aproximada dada por

$$\tilde{E}(\boldsymbol{\theta}_0 + \Delta\boldsymbol{\theta}) = (\mathbf{R}(\boldsymbol{\theta}_0) - \mathbf{J}(\boldsymbol{\theta}_0)\Delta\boldsymbol{\theta})^T \mathbf{W}(\mathbf{R}(\boldsymbol{\theta}_0) - \mathbf{J}(\boldsymbol{\theta}_0)\Delta\boldsymbol{\theta}),$$

donde $\mathbf{R} = \mathbf{C} - \mathbf{f}(\boldsymbol{\theta}_0)$ representa el valor residual entre la diferencia de \mathbf{C} y la superficie parametrizada por $\boldsymbol{\theta}_0$. La minimización de \tilde{E} respecto de $\Delta\boldsymbol{\theta}$ equivale a un problema de mínimos cuadrados lineal cuya solución para $\Delta\boldsymbol{\theta}$ viene dada por

$$\Delta\boldsymbol{\theta} = \left(\mathbf{J}(\boldsymbol{\theta}_0)^T \mathbf{W} \mathbf{J}(\boldsymbol{\theta}_0) \right)^{-1} \mathbf{J}(\boldsymbol{\theta}_0)^T \mathbf{W} \mathbf{R}(\boldsymbol{\theta}_0).$$

La cantidad $\Delta\boldsymbol{\theta}$ es un vector que minimiza $\tilde{E}(\boldsymbol{\theta}_0 + \Delta\boldsymbol{\theta})$, y que apunta en la dirección en la que se reduce E . La minimización de E puede conseguirse de forma iterativa, calculando $\Delta\boldsymbol{\theta}$, y repitiendo el procedimiento con $\boldsymbol{\theta}_0 = \boldsymbol{\theta}_0 + \Delta\boldsymbol{\theta}$. Este método es el conocido como el método de Gauss-Newton.

El algoritmo de Levenberg-Marquardt introduce una modificación de esta expresión a través de un factor de amortiguamiento, λ , como

$$\Delta\boldsymbol{\theta} = (\mathbf{J}(\boldsymbol{\theta}_0)^T \mathbf{W} \mathbf{J}(\boldsymbol{\theta}_0) + \lambda \mathbf{I})^{-1} \mathbf{J}(\boldsymbol{\theta}_0)^T \mathbf{W} \mathbf{R}(\boldsymbol{\theta}_0),$$

siendo \mathbf{I} la matriz identidad. El algoritmo se implementa como sigue:

1. Proponer una combinación inicial de parámetros, $\boldsymbol{\theta}_0$. Esta se puede estimar a partir de la aplicación de otro método de estimación, como el método de ajuste gaussiano de tres puntos.
2. Calcular el error $E(\boldsymbol{\theta}_0)$, el residuo $\mathbf{R}(\boldsymbol{\theta}_0)$ y la matriz jacobiana, $\mathbf{J}(\boldsymbol{\theta}_0)$.
3. Calcular $\Delta\boldsymbol{\theta}$. Si se obtiene $E(\boldsymbol{\theta}_0 + \Delta\boldsymbol{\theta}) < E(\boldsymbol{\theta}_0)$, continuar al paso 3.2. Caso contrario, continuar al paso 3.1.
 - 3.1. Incrementar el valor del factor de amortiguamiento, λ , y recalculer el valor de $\Delta\boldsymbol{\theta}$. Repetir este paso hasta que se cumpla $E(\boldsymbol{\theta}_0 + \Delta\boldsymbol{\theta}) < E(\boldsymbol{\theta}_0)$. Luego pasar al paso 3.2.
 - 3.2. Reducir el factor de amortiguamiento, λ , y hacer $\boldsymbol{\theta}_0 = \boldsymbol{\theta}_0 + \Delta\boldsymbol{\theta}$.

4. Repetir el procedimiento a partir del paso 2. Continuar el procesamiento hasta cumplir con algún criterio de convergencia o finalización.

Dentro de los criterios de convergencia y finalización se encuentran, establecer una cantidad límite de iteraciones, un valor mínimo $\|\Delta\theta\|/\|\theta\|$, o un valor mínimo para $\|J(\theta)W R\|$.

IX.8. DERIVACIÓN DEL MÉTODO DE ESTIMACIÓN SUBPÍXEL BASADO EN EL FLUJO ÓPTICO

La imagen de partículas en PIV representa un patrón de intensidades en el plano, el cual puede ser descrito a través de una función $I(x, y, t)$, en donde x e y constituyen las componentes horizontal y vertical del plano, respectivamente, y t representa el tiempo. Debido al movimiento de las partículas, este patrón se desplaza con el tiempo. Si se considera que en un pequeño intervalo de tiempo Δt , el patrón de intensidades se desplaza una cantidad Δx horizontalmente, y una cantidad Δy verticalmente, y además se asume que las intensidades de este patrón no varían (la intensidad de las partículas permanece constante en el tiempo), entonces debe tenerse

$$I(x, y, t) = I(x + \Delta x, y + \Delta y, t + \Delta t).$$

Si el miembro de la derecha se desarrolla en series de Taylor, se obtiene

$$I(x, y, t) = I(x, y, t) + \Delta x \frac{\partial I}{\partial x} + \Delta y \frac{\partial I}{\partial y} + \Delta t \frac{\partial I}{\partial t} + \varepsilon,$$

donde ε contiene términos de orden iguales y mayores a 2 en Δx , Δy y Δt . Restando $I(x, y, t)$, y dividiendo por Δt , se obtiene

$$0 = \frac{\Delta x}{\Delta t} \frac{\partial I}{\partial x} + \frac{\Delta y}{\Delta t} \frac{\partial I}{\partial y} + \frac{\partial I}{\partial t} + O(\Delta t),$$

donde $O(\Delta t)$ es un término del orden de Δt (se asume que Δx y Δy varían con Δt). Finalmente, en el límite con $\Delta t \rightarrow 0$ resulta

$$u \frac{\partial I}{\partial x} + v \frac{\partial I}{\partial y} + \frac{\partial I}{\partial t} = 0,$$

siendo $u = \frac{dx}{dt}$ y $v = \frac{dy}{dt}$ las componentes de velocidad horizontal y vertical del patrón de intensidades, respectivamente. Esta es la ecuación del flujo óptico.

La ecuación del flujo óptico puede emplearse para estimar el desplazamiento de partículas con precisión subpíxel. Sean las ventanas de interrogación g_1 y g_2 , representadas por medio de dos matrices de tamaño $N_i \times N_j$. Los valores de intensidad para cada píxel de estas ventanas vienen dados por $g_1(i, j)$ y $g_2(i, j)$, con $i = 1, 2, \dots, N_i$ y $j = 1, 2, \dots, N_j$. Si se permite que g_1 y g_2 representen el patrón de intensidades a tiempo t y $t + \Delta t$, respectivamente, entonces se tiene

$$I(i, j, t) = g_1(i, j),$$

$$I(i, j, t + \Delta t) = g_2(i, j).$$

Por conveniencia, se introducen las ventanas expresadas en forma de vector columna, \mathbf{g}_1 y \mathbf{g}_2 , definidas por

$$\begin{cases} \mathbf{g}_1(k) = g_1(\text{mod}(k-1, N_i) + 1, \text{div}(k, N_i)), \\ \mathbf{g}_2(k) = g_2(\text{mod}(k-1, N_i) + 1, \text{div}(k, N_i)), \end{cases} \quad \text{con } k = 1, \dots, N = N_i N_j$$

siendo $\text{mod}(x, y)$ y $\text{div}(x, y)$ las funciones que devuelven, respectivamente, el resto y el cociente de la división euclídea entre x e y . Análogamente, se hace

$$\mathbf{I}(k, t) = \mathbf{g}_1(k),$$

$$\mathbf{I}(k, t + \Delta t) = \mathbf{g}_2(k).$$

La derivada parcial de \mathbf{I} con respecto al tiempo puede aproximarse por un esquema de diferencias finitas como

$$\frac{\partial \mathbf{I}}{\partial t} \approx \frac{\mathbf{g}_2 - \mathbf{g}_1}{\Delta t}.$$

Por otro lado, las derivadas parciales con respecto a las componentes espaciales se pueden aproximar como

$$\frac{\partial \mathbf{I}}{\partial x} \approx \frac{1}{2} \left(\left. \frac{\partial \mathbf{I}}{\partial x} \right|_t + \left. \frac{\partial \mathbf{I}}{\partial x} \right|_{t+\Delta t} \right) \approx \frac{1}{2} (\mathbf{D}_x \mathbf{g}_1 + \mathbf{D}_x \mathbf{g}_2),$$

$$\frac{\partial \mathbf{I}}{\partial y} \approx \frac{1}{2} \left(\left. \frac{\partial \mathbf{I}}{\partial y} \right|_t + \left. \frac{\partial \mathbf{I}}{\partial y} \right|_{t+\Delta t} \right) \approx \frac{1}{2} (\mathbf{D}_y \mathbf{g}_1 + \mathbf{D}_y \mathbf{g}_2),$$

donde \mathbf{D}_x y \mathbf{D}_y son matrices de $N \times N$ que representan operadores de diferencias finitas para la aproximación de las derivadas espaciales de \mathbf{g}_1 y \mathbf{g}_2 . A partir de la ecuación de flujo óptico se obtiene la relación

$$\frac{\mathbf{u}\Delta t}{2} (\mathbf{D}_x \mathbf{g}_1 + \mathbf{D}_x \mathbf{g}_2) + \frac{\mathbf{v}\Delta t}{2} (\mathbf{D}_y \mathbf{g}_1 + \mathbf{D}_y \mathbf{g}_2) + \mathbf{g}_2 - \mathbf{g}_1 = 0,$$

donde \mathbf{u} y \mathbf{v} son matrices diagonales de $N \times N$, cuyos elementos diagonales son los componentes de velocidad correspondientes a cada punto discreto sobre el área de interrogación. De esta forma, queda conformado un sistema de N ecuaciones con $2N$ incógnitas, por lo que se requieren condiciones adicionales para poder resolverlo. Sugii, Nishio, Okuno, & Okamoto (2000) proponen asumir que la velocidad es uniforme en toda la ventana de interrogación, reduciendo el número de incógnitas a 2, y aplicar un esquema de mínimos cuadrados, buscando minimizar la función de error dada por

$$E(\delta_x, \delta_y) = \left(\frac{\delta_x}{2} (\mathbf{D}_x \mathbf{g}_1 + \mathbf{D}_x \mathbf{g}_2) + \frac{\delta_x}{2} (\mathbf{D}_y \mathbf{g}_1 + \mathbf{D}_y \mathbf{g}_2) + \mathbf{g}_2 - \mathbf{g}_1 \right)^T \mathbf{W} \left(\frac{\delta_x}{2} (\mathbf{D}_x \mathbf{g}_1 + \mathbf{D}_x \mathbf{g}_2) + \frac{\delta_x}{2} (\mathbf{D}_y \mathbf{g}_1 + \mathbf{D}_y \mathbf{g}_2) + \mathbf{g}_2 - \mathbf{g}_1 \right),$$

donde \mathbf{W} es una matriz de pesos.

Si se hace $\boldsymbol{\delta} = [\delta_x, \delta_y]^T$, y

$$\mathbf{H} = \left[\frac{\mathbf{D}_x \mathbf{g}_1 + \mathbf{D}_x \mathbf{g}_2}{2}, \frac{\mathbf{D}_y \mathbf{g}_1 + \mathbf{D}_y \mathbf{g}_2}{2} \right],$$

entonces la función de error cuadrático se puede escribir como

$$E(\boldsymbol{\delta}) = (\mathbf{g}_2 - \mathbf{g}_1 + \mathbf{H}\boldsymbol{\delta})^T \mathbf{W} (\mathbf{g}_2 - \mathbf{g}_1 + \mathbf{H}\boldsymbol{\delta}).$$

La solución a esta ecuación viene dada por

$$\boldsymbol{\delta} = -(\mathbf{H}^T \mathbf{W} \mathbf{H})^{-1} \mathbf{H}^T \mathbf{W} (\mathbf{g}_2 - \mathbf{g}_1).$$

IX.9. APLICACIÓN DEL MÉTODO DE VALIDACIÓN TODO EN UNO

Sea un campo vectorial $\check{\mathbf{V}}: [1, 2, \dots, N_i] \times [1, 2, \dots, N_j] \rightarrow \mathbb{R}^2$, definido en unos puntos discretos que conforman una grilla regular de dimensiones $N_i \times N_j$. Este campo vectorial puede ser representado por medio de dos matrices u y v , de dimensiones $N_i \times N_j$, tales que $u(i, j)$ y $v(i, j)$ son, respectivamente, las componentes horizontal y vertical del vector $\check{\mathbf{V}}(i, j)$. El análisis precedente se continua para u , siendo análogamente aplicable para v .

Se desea encontrar una matriz u_s , tal que guarde cierto grado de proximidad con la matriz u , y que además presente un nivel de suavizado. Para la medida de proximidad, se tendrá en cuenta la suma de la diferencia de cuadrados entre los elementos de u y u_s , mientras que, para el nivel de suavizado, se calculará un factor de rugosidad en función de las derivadas de segundo orden, expresadas en forma de diferencias finitas, de la matriz u_s . Se plantea entonces un problema de mínimos cuadrados por penalización, dado por

$$E(u_s) = \sum_{i,j} w(i, j) [u(i, j) - u_s(i, j)]^2 + s P(u_s),$$

donde w es una matriz de pesos, $s \geq 0$ es un parámetro de suavizado, y $P(u_s)$ es el factor de rugosidad dado por

$$P(u_s) = \sum_{i,j} \left(\left[\frac{\partial^2 u_s}{\partial x^2}(i, j) \right]^2 + \left[\frac{\partial^2 u_s}{\partial y^2}(i, j) \right]^2 + 2 \left[\frac{\partial^2 u_s}{\partial x \partial y}(i, j) \right]^2 \right),$$

siendo las derivadas en esta expresión estimadas por medio de diferencias finitas sobre los valores de la matriz u_s . La matriz u_s deseada es aquella matriz tal que $E(u_s)$ resulta mínimo. Si se hace $s = 0$, está claro que la solución es trivial con $u_s = u$, y la matriz u_s resultante es completamente interpolante de u , sin ofrecer ningún grado de suavizado. Para valores mayores de s , el término de rugosidad cobra peso y la solución u_s presenta un compromiso entre la rugosidad y su proximidad a u .

A continuación, se introduce la representación de la matriz u en forma de vector columna \mathbf{u} , de $N = N_i N_j$ elementos, dados por

$$\mathbf{u}(k) = u(\text{mod}(k - 1, N_i) + 1, \text{div}(k, N_i)) \quad \text{con } k = 1, \dots, N,$$

que equivale a formar un vector columna disponiendo cada una de las columnas de la matriz u , una sobre la otra. De forma análoga se introduce el vector columna \mathbf{u}_s como el vector que surge de reordenar a la matriz u_s de la misma forma. Luego, se define la matriz W como la matriz diagonal de dimensiones $N \times N$ tal que

$$W(k, k) = w(\text{mod}(k - 1, N_i) + 1, \text{div}(k, N_i)) \quad \text{con } k = 1, \dots, N,$$

que equivale a reordenar la matriz de pesos en forma de vector columna de la misma forma que u y u_s , y luego formar una matriz diagonal disponiendo los elementos de este vector en su diagonal. Finalmente, se introducen las matrices D_{xx} , D_{yy} , y D_{xy} , de dimensiones $N \times N$ tales que los productos matriciales $D_{xx}\mathbf{u}_s$, $D_{yy}\mathbf{u}_s$ y $D_{xy}\mathbf{u}_s$ produzcan, respectivamente, las aproximaciones en diferencias finitas para $\partial^2 \mathbf{u}_s / \partial x^2$, $\partial^2 \mathbf{u}_s / \partial y^2$ y $\partial^2 \mathbf{u}_s / \partial x \partial y$. Introduciendo estas cantidades, la función de error se puede expresar de forma vectorial como

$$E(\mathbf{u}_s) = (\mathbf{u} - \mathbf{u}_s)^T W (\mathbf{u} - \mathbf{u}_s) + s \mathbf{u}_s^T M \mathbf{u}_s,$$

donde M es la matriz que viene dada por

$$M = D_{xx}^T D_{xx} + D_{yy}^T D_{yy} + 2D_{xy}^T D_{xy}.$$

la solución de este problema de mínimos cuadrados es

$$\mathbf{u}_s = (W + sM)^{-1} W \mathbf{u}. \quad \text{Ec. IX.9-1}$$

Para la aproximación de las derivadas, se utilizan diferencias centradas de segundo orden, de forma que resulta

$$\frac{\partial^2 u_s}{\partial x^2}(i, j) = \frac{u_s(i, j - 1) - 2u_s(i, j) + u_s(i, j + 1)}{\Delta x^2},$$

$$\frac{\partial^2 u_s}{\partial y^2}(i, j) = \frac{u_s(i - 1, j) - 2u_s(i, j) + u_s(i + 1, j)}{\Delta y^2},$$

$$\frac{\partial^2 u_s}{\partial x \partial y}(i, j) = \frac{u_s(i+1, j+1) + u_s(i, j) - u_s(i+1, j) - u_s(i, j+1)}{\Delta x \Delta y},$$

donde Δx y Δy representan el espaciamiento de la grilla en el sentido horizontal y vertical, respectivamente. Por otro lado, las derivadas en los bordes de la grilla se aproximan extrapolando el valor del campo fuera de la grilla como el valor más cercano, es decir, $u_s(0, j) = u_s(1, j)$, $u_s(i, 0) = u_s(i, 1)$, $u_s(N_i + 1, j) = u_s(N_i, j)$, y $u_s(i, N_j + 1) = u_s(i, N_j)$. De esta forma, resulta

$$\frac{\partial^2 u_s}{\partial x^2}(i, 1) = \frac{-u_s(i, 1) + u_s(i, 2)}{\Delta x^2},$$

$$\frac{\partial^2 u_s}{\partial x^2}(i, N_j) = \frac{u_s(i, N_j - 1) - u_s(i, N_j)}{\Delta x^2},$$

$$\frac{\partial^2 u_s}{\partial y^2}(1, j) = \frac{-u_s(1, j) + u_s(2, j)}{\Delta y^2},$$

$$\frac{\partial^2 u_s}{\partial y^2}(N_i, j) = \frac{u_s(N_i - 1, j) - u_s(N_i, j)}{\Delta y^2},$$

$$\frac{\partial^2 u_s}{\partial x \partial y}(N_i, j) = \frac{\partial^2 u_s}{\partial x \partial y}(i, N_j) = 0.$$

Al emplear estas expresiones, las matrices D_{xx} , D_{yy} y D_{xy} pueden computarse como

$$D_{xx} = I_{N_i} \otimes K_{N_j},$$

$$D_{xy} = K_{N_i} \otimes I_{N_j},$$

$$D_{yy} = D_{N_i} \otimes D_{N_j},$$

donde \otimes denota el producto de Kronecker, I_n representa la matriz identidad de tamaño $n \times n$, mientras que K_n y D_n son las matrices de $n \times n$ dadas por

$$K_n = \begin{bmatrix} -1 & 1 & & & \\ 1 & -2 & 1 & & \\ & 1 & -2 & 1 & \\ & & \cdots & \cdots & \cdots \\ & & & 1 & -1 \end{bmatrix},$$

$$D_n = \begin{bmatrix} 1 & -1 & & & \\ & 1 & -1 & & \\ & & \ddots & \ddots & \\ & & & 1 & -1 \end{bmatrix}.$$

Este algoritmo se puede utilizar para eliminar la influencia de vectores espurios como sigue. Para un dado valor del parámetro s , se calcula \mathbf{u}_s (Ec. IX.9-1). Luego, se determina el residual estudentizado \mathbf{r} dado por

$$\mathbf{r} = \frac{\mathbf{u} - \mathbf{u}_s}{1,4826 \text{MAD}(\mathbf{u} - \mathbf{u}_s) \sqrt{1 - h}},$$

donde $\text{MAD}(\mathbf{u} - \mathbf{u}_s)$ representa la desviación media absoluta de los valores de $\mathbf{u} - \mathbf{u}_s$, y h es la traza de la matriz $(I_N + sM)^{-1}$.

A partir de los valores residuales, se recalcula la diagonal de la matriz de pesos W como

$$W(n, n) = \begin{cases} \left(1 - \left(\frac{\mathbf{r}(n)}{4,685}\right)^2\right)^2 & \text{si } \left|\left(\frac{\mathbf{r}(n)}{4,685}\right)\right| < 1. \\ 0 & \text{en caso contrario} \end{cases}$$

De esta forma, los valores de \mathbf{u} que tengan un desvío muy grande con respecto a \mathbf{u}_s en el sentido del residual estudentizado, serán interpretados como valores espurios, asignándoles un valor bajo o nulo para la matriz de pesos. Si se repite el cálculo de \mathbf{u}_s con esta nueva matriz de pesos, los vectores espurios tendrán una menor contribución sobre la forma de \mathbf{u}_s . El procedimiento puede repetirse durante una determinada cantidad de etapas, o hasta que el número de vectores espurios (dados por la cantidad de ceros en la diagonal de W) no cambie entre dos iteraciones sucesivas.

En la bibliografía también se propone una forma de determinar un valor óptimo para el parámetro s , mediante el método de validación cruzada generalizada o GCV (del inglés, Generalized Cross Validation). Se propone el indicador GCV dado por

$$GCV(s) = \frac{\frac{1}{N}(\mathbf{u} - \mathbf{u}_s)^T W (\mathbf{u} - \mathbf{u}_s)^T}{\left(1 - \frac{\text{Tr}[(I_N + sM)^{-1}]}{N}\right)^2}$$

El valor óptimo para s se determina como aquel que minimiza la función GCV .

El algoritmo se puede resumir como sigue:

1. Inicialmente hacer $W = I_N$, y tomar un valor inicial para s .
2. Calcular el valor de \mathbf{u}_s (ecuación Ec. IX.9-1).
3. Determinar el vector de residuos estudentizados, \mathbf{r} , y recalculer los valores de la matriz W .
4. Determinar el nuevo valor óptimo para s a partir del mínimo de la función $GCV(s)$.

5. Repetir el paso 2 con el nuevo valor de s y la nueva matriz W . Continuar la iteración hasta completar cierto número de iteraciones o hasta que no se detecten nuevos vectores espurios.

IX.10. GENERACIÓN DE IMÁGENES SINTÉTICAS Y MÉTODO DE MONTECARLO

Las ventanas sintéticas para simulaciones numéricas en este trabajo fueron generadas superponiendo imágenes de partículas posicionadas aleatoriamente en un fondo oscuro, con la adición posterior de ruido. Las partículas fueron modeladas con distribución gaussiana, considerando el diámetro de la partícula como el diámetro del círculo que comprende los puntos de la partícula con una intensidad superior a e^{-2} veces la intensidad máxima de la misma. De esta forma, las ventanas quedan determinadas por medio de las expresiones

$$g_1(i, j) = \sum_{n=1}^{Np} I_1(n) e^{-\frac{8}{Dp(n)^2}[(i-x_0(n))^2+(j-y_0(n))^2]} + w_1(i, j),$$

$$g_2(i, j) = \sum_{n=1}^{Np} I_2(n) e^{-\frac{8}{Dp(n)^2}[(i-x_f(n))^2+(j-y_f(n))^2]} + w_2(i, j),$$

donde Np es el número de partículas, Dp es el diámetro de las partículas, I_1 y I_2 son las intensidades máximas de las partículas en la primera y segunda ventana, respectivamente, $[x_0, y_0]$ y $[x_f, y_f]$ son las posiciones inicial y final de las partículas, y w_1 y w_2 son señales de ruido aleatorio añadidas sobre las ventanas. En principio, las intensidades de las partículas podrían ser diferentes entre el par de ventanas, para tener en cuenta efectos de movimientos fuera de plano o iluminación no uniforme del haz de luz. Sin embargo, en las simulaciones realizadas en este trabajo no se tuvieron en cuenta dichos efectos y se tomó $I_1(n) = I_2(n) = I(n)$ para todo n .

Para cada par de ventanas, la cantidad de partículas Np es determinada por la densidad de partículas por unidad de área, ρ_p . Las posiciones iniciales de las partículas, $[x_0, y_0]$, son seleccionadas de forma aleatoria mediante una distribución uniforme. La posición final de las partículas, $[x_f, y_f]$, es determinada a partir de las posiciones iniciales y el campo impuesto. En estas simulaciones se consideró un perfil de velocidades lineal dentro de la ventana, por lo que el desplazamiento de las partículas queda definido únicamente por medio de un desplazamiento medio horizontal, u_m , un desplazamiento medio vertical, v_m , y las cantidades relacionadas a los gradientes de desplazamiento dentro de la ventana en la dirección horizontal, du/dx y dv/dx , y en la dirección vertical, du/dy y dv/dy . De esta forma, las posiciones finales para las partículas en la segunda ventana quedan dadas por

$$[x_f, y_f] = \left[x_0 + u_m + \frac{du}{dx} x_0 + \frac{du}{dy} y_0, y_0 + v_m + \frac{dv}{dx} x_0 + \frac{dv}{dy} y_0 \right].$$

Los diámetros de las partículas Dp dentro de cada ventana son seleccionados a partir de una distribución normal, con media μ_{Dp} y desvío estándar σ_{Dp} . Las señales de ruido, w_1 y w_2 son generadas de forma aleatoria con distribución normal, con media μ_R y σ_R . La intensidad I de las partículas dentro de la ventana se toma a partir de una distribución normal con media μ_I y desvío estándar σ_I . La intensidad media de las partículas dentro de la ventana se determina como $\mu_I = SR \sigma_R$ para $SR > 0$, o $\mu_I = 0,1 \sigma_R$ para $SR = 0$, donde SR es una medida de la relación señal-ruido para la ventana. Por otro lado, el desvío estándar para la intensidad de las partículas dentro de la ventana, σ_I , se toma como $\sigma_I = SR \bar{\sigma}_I$, donde $\bar{\sigma}_I$ es un parámetro de variación.

IX.11. ANÁLISIS DEL CÓDIGO PRANA

En esta sección del anexo se muestra un análisis efectuado sobre el código Prana en su versión r.2013-04-22, en el cual se describen todas las funciones que el software utiliza en lo que respecta a procesamiento PIV, detallando sus entradas y salidas.

Prana.m

Entradas
-
Salidas
-
Descripción
Función que inicializa el programa. Abre la interfaz de usuario por medio de la cual se ingresan los parámetros de la simulación. Gestiona todos los parámetros ingresados por el usuario dentro de la interfaz gráfica, almacenándolos en una variable tipo struct llamada <i>Data</i> . Los campos de esta estructura se muestran en la Tabla 2. En primer lugar, la función ejecuta la subrutina <i>buildDefaultPranaJob.m</i> , que genera una variable tipo struct llamada <i>defaultdata</i> , con la misma estructura que <i>Data</i> , que contiene valores predeterminados para los parámetros de simulación. Si el usuario genera un nuevo trabajo a partir del botón “New Job”, se genera un trabajo por defecto con los parámetros contenidos en <i>defaultdata</i> . Por otro lado, si el usuario carga un trabajo por medio de “Load Job”, se ejecuta la función <i>jobfile_validator.m</i> , que se encarga de validar los datos contenidos en el fichero cargado. Finalmente, cuando el usuario ejecuta los trabajos a partir de “Run Current Job” o “Run All Jobs”, la función llama a las subrutinas <i>pranaPIVcode.m</i> , en el caso de que se haya seleccionado la opción de “Particle Image Velocimetry”, y <i>pranaPTVcode.m</i> , en el caso de que se hayan seleccionado las opciones de “Particle Identification”, “Particle Sizing” o “Particle Tracking”.
Llamadas a otras funciones
<i>buildDefaultPranaJob.m</i>
<i>pranaPIVcode.m</i>
<i>pranaPTVcode.m</i>
<i>jobfile_validator.m</i>

Tabla 2 – Campos de la estructura Data

Campo	Descripción
clientversion	Versión de Prana.
imdirec	Directorio donde se encuentran las imágenes a ser procesadas.
outdirec	Directorio en donde se almacenarán los resultados.
maskdirec	Directorio donde se encuentra la máscara.
imbase	Nombre base de las imágenes a ser procesadas.
imzeros	cantidad de ceros que contiene la indexación de las imágenes.
Imext	Extensión de las imágenes.
imcstep	Salto en los índices de las imágenes del par de imágenes a ser procesadas.
imfstep	Salto en los índices entre pares de imágenes.
imfstart	Índice de la primera imagen a ser procesada.
imfend	Índice de la última imagen a ser procesada.
wrmag	Resolución de la imagen en $\mu m/píxel$.
wrsamp	Frecuencia de muestreo en Hz.
Wrsep	Separación de pulsos láser en segundos.
batchname	Nombre del trabajo siendo ejecutado.
datout	Especifica si los resultados se almacenan en archivo de extensión .dat. Su valor es 0 o 1.
multiplematout	Especifica si los resultados se almacenan en archivo de extensión .mat. Su valor es 0 o 1.
exp_date	Fecha del experimento.
exp_L	Longitud característica.
exp_v0	Velocidad característica.
exp_notes	Notas del usuario. Cada línea de texto se guarda como char en un elemento de la celda.
exp_density	Densidad del fluido en kg/m^3 .
exp_viscosity	Viscosidad dinámica en Pa s.
exp_surfacetension	Tensión superficial del fluido en N/m.
exp_partD	Diámetro de partícula en micrómetros.
exp_partdensity	Densidad de la partícula en kg/m^3 .
exp_wavelength	Longitud de onda del láser en micrómetros.

exp_pixelsize	Tamaño físico del píxel en micrómetros.
exp_lensfocal	Longitud focal del lente en milímetros.
exp_lensfnum	Número focal.
exp_micro	Indica si se realiza microPIV. Vale 0 o 1.
exp_NA	Apertura numérica.
exp_n	Índice de refracción.
exp_Re	Número de Reynolds. Calculado a partir de los datos ingresados.
exp_St	Número de Stokes de la partícula. Calculado a partir de los datos ingresados.
exp_M	Magnificación. Calculado a partir de los datos ingresados.
exp_ROI	Región de interés en metros. Calculado a partir de los datos ingresados.
exp_diffractiondiameter	Diámetro de difracción en metros. Calculado a partir de los datos ingresados.
exp_depthffocus	Profundidad del foco en micrómetros. Calculado a partir de los datos ingresados.
masktype	Tipo de máscara. Puede ser nula ("none"), estática ("static") o dinámica ("dynamic").
staticmaskname	Nombre completo del archivo de máscara, que contiene el directorio y la extensión del archivo.
maskbase	Nombre base de los archivos de máscara. Solo aplica con máscara dinámica.
maskzeros	Cantidad de ceros en la indexación de los archivos de máscara.
maskext	Extensión de los archivos de máscara.
maskstep	Salto en los índices de los archivos de máscara.
maskstart	Índice inicial de los archivos de máscara.
runPIV	Indica si se ejecuta el procesamiento PIV del trabajo actual. Vale 0 o 1. Si vale 1 se ejecuta la función pranaPIVcode.m con la estructura Data como argumento.
PIV0	Estructura de datos por defecto.
PIV1	Estructura que contiene los parámetros del paso 1. Por cada paso que se agrega se adiciona un campo PIV#n, donde #n es el número del paso. Los campos de esta estructura se muestran en la Tabla 3.

passes	Cantidad de pasos PIV para cada par de imágenes.
method	Método PIV: 1 - Multipass DWO, 2 – Multigrid DWO, 3 – Multigrid Deform CWO, 4 – Multigrid Ensemble DWO, 5 – Multigrid Ensemble with deform CWO, 6 – Multigrid multiframe DWO.
velinterp	Método de interpolación multigrid. 1 – Nearest Neighbor, 2 – Bilinear, 3 – Bicubic.
iminterp	Método de deformación de imagen. 1 – Función cardinal, 2 – Función cardinal con filtro de Blackman.
framestep	Máximo salto de cuadros para el método “multiframe”.
PIVerror	Error para el método “multiframe”.
channel	Canal de color. 1 – Rojo (escala de gris), 2 – verde, 3 – azul, 4 – Media ponderada, 5 – Media, 6 – Color ensemble.
ID	Estructura que contiene los datos para “Particle Identification”.
Size	Estructura que contiene los datos para “Particle Sizing”.
Track	Estructura que contiene los datos para “Particle Tracking”.
Splash	Valor asignado desde buildDefaultPranaJob.m, fijado en 1. Si su valor es 0, la función Prana.m muestra una ventana con las novedades de la última versión.
outputpassbase	Nombre base para el archivo de resultados.
Versión	Versión de Prana.
ptv_version	Versión de “Particle Tracking”.
Par	Activa el procesamiento de imágenes en paralelo. Vale 0 o 1.
parprocessors	Cantidad de procesadores trabajando en paralelo.
Cpass	Paso actual.

Tabla 3 - Campos de la estructura PIV#n

Campo	Descripción
Winres	Resolución de las ventanas en la primera y segunda imagen a ser interrogadas. Expresado en píxeles.
Winsize	Tamaño real de ventana luego de aplicar rellenado de ceros y ventaneo. Expresado en píxeles. Si se activa la casilla auto, esta se calcula de forma automática, tomando las dimensiones x e y más

	<p>altas dadas en winres y calculando la ventana final como las siguientes potencias de 2. Por otro lado, si el método de correlación es correlación cruzada directa, esta se calcula como la suma de las dimensiones de la primera y segunda ventanas menos 1.</p>
Winauto	Indica si el tamaño final de ventana se ingresa por el usuario o se calcula de forma automática. Vale 0 o 1.
Gridres	Resolución del campo de velocidades dado en píxeles.
Winoverlap	Solapamiento de ventanas dado en porcentaje.
Gridtype	Tipo de grilla. Vale 1 si el método es "multipass" o 2 en cualquier otro caso.
Gridbuf	"Buffer" de grilla dado en píxeles. El "buffer" de la grilla es el margen en píxeles desde el borde de la región de interés hasta el punto de la grilla más cercano.
BWO	"Bulk window offset" en píxeles. Contiene dos valores para el offset en x y en y. Esto implica agregar un desplazamiento de las ventanas entre imágenes en un primer paso de PIV para compensar los flujos con velocidades medias significativas. En el segundo paso y posteriores se utiliza el campo de velocidades estimado en el paso anterior para este propósito.
Corr	Tipo de método de correlación. 'SCC' – Standard Cross Correlation, 'RPC' – Robust Phase Correlation, 'DRPC' – Dynamic Robust Phase Correlation, 'GCC' - Generalized Cross Correlation, 'FWC' – Fraction-Filter Weighted Correlation, 'SPC' – Spectral Phase Correlation, 'DCC' – Direct Cross Correlation.
RPCd	Diámetro de partícula para los métodos RPC y SPC.
frac_filt	Peso para el filtro fraccional del método FWC.
Zeromean	Indica ventanas de media cero. Puede valer 0 o 1.
Peaklocator	Método de estimación de máximo con resolución subpíxel. 1 – Estimador gaussiano de tres puntos. 2 – Estimador gaussiano de cuatro puntos. 3 – Regresión de mínimos cuadrados gaussiana.
Velsmooth	Indica si se le aplica un suavizado gaussiano al campo de velocidades calculado. Puede valer 0 o 1.

Velsmoothfilt	Peso del filtro gaussiano aplicado para el suavizado.
deform_min	Número de iteraciones mínimo para el método de deformación.
deform_max	Número de iteraciones máximo para el método de deformación.
deform_conv	Factor de convergencia para el método de deformación.
Val	Indica si se realiza un paso de validación sobre los vectores calculados. Puede valer 0 o 1.
Uod	Indica si se aplica UOD (Universal Outlier Detection). Puede valer 0 o 1.
Bootstrap	Indica si se aplica "Bootstrapping". Puede valer 0 o 1.
Thresh	Indica si se aplica método de "Threshold" sobre los vectores de velocidad calculados. Puede valer 0 o 1.
uod_type	Indica el tipo de UOD. 1 – media. 2 – mediana.
uod_window	Tamaño de ventana para el UOD en cantidad de puntos de grilla.
uod_thresh	Valor umbral para el método UOD, expresado en desviaciones estándar de los datos dentro de la ventana UOD.
bootstrap_percentsampled	Porcentaje muestreado para el método de "Bootstrapping".
bootstrap_iterations	Número de iteraciones por "frame" para el método de "Bootstrapping".
bootstrap_passes	Número de pasadas para el método de "Bootstrapping".
Valuthresh	Límites para el componente U de velocidad para el método de "Threshold" de velocidad.
Valvthresh	Límites para el componente V de velocidad para el método de "Threshold" de velocidad.
Valextrapeaks	Indica si se evalúan picos de correlación adicionales si falla la validación utilizando el pico principal. Puede valer 0 o 1.
Savepeakinfo	Indica si se guarda información de picos adicionales. Puede valer 0 o 1.

Corrpeaknum	Indica la cantidad de picos adicionales de los que se guarda información. 1 – Pico 1; 2 – Pico 1 y 2; 3 – Pico 1, 2 y 3.
Savepeakmag	Indica que se guarde la magnitud de los picos. Puede valer 0 o 1.
Savepeakvel	Indica que se guarde la velocidad resultante de los picos. Puede valer 0 o 1.
Saveplane	Indica si se desean guardar los planos de correlación. Puede valer 0 o 1. Esta opción resulta útil en etapa de ajuste de parámetros, pero consume mucha memoria.
Outbase	Nombre base para el archivo donde se guardan los resultados de cada paso.
Write	Indica si se guardan los resultados de cada paso. Puede valer 0 o 1.

bootstrapping.m

Entradas

X: matriz de dos dimensiones con las componentes en *x* de la grilla, dadas en píxeles, sobre las que se calculó el campo de velocidades.

Y: matriz de dos dimensiones con las componentes en *y* de la grilla, dadas en píxeles, sobre las que se calculó el campo de velocidades.

u: matriz de las mismas dimensiones que *X*, que contiene los desplazamientos en *x* estimados a ser validados por el método de “Bootstrapping”.

v: matriz de las mismas dimensiones que *X*, que contiene los desplazamientos en *y* estimados a ser validados por el método de “Bootstrapping”.

per: porcentaje de muestreo.

iter: cantidad de iteraciones del método “Bootstrapping”.

kmax: número de pasos del método “Bootstrapping”.

Eval: Matriz de las mismas dimensiones que *X*. El contenido de esta matriz se detalla en la Tabla 4.

Salidas

U: matriz de desplazamientos *u* validada.

V: matriz de desplazamientos *v* validada.

Eval: matriz *Eval* validada. La matriz toma distintos valores según como haya pasado el punto de grilla el proceso de validación. -1 – el punto de grilla no fue procesado debido a que estaba enmascarado; 0 – el punto de grilla fue procesado y pasó la validación; 200 – el punto de grilla fue procesado, pero falló en la validación por “Bootstrapping”.

Descripción

Función que valida mediante el método de “Bootstrapping” el campo de velocidades calculado por el método PIV. Los vectores que no pasan la validación son reemplazados por la moda obtenida de los histogramas.

Llamadas a otras funciones

bootstrapping_dataremove.m
gridfit.m

Valor	Descripción
-1	El punto no fue procesado debido a que fue enmascarado.
0	El punto fue procesado y pasó la validación.
k	El punto fue procesado, pero falló la validación UOD en el paso k del algoritmo UOD.
100	El punto fue procesado, pero falló la validación por Thresholding.
200	El punto fue procesado, pero falló la validación por Bootstrapping.

Tabla 4 - Significado de los distintos valores en la matriz Eval.

bootstrapping_dataremove.m

<p>Entradas</p> <p><i>DSIZE</i>: arreglo de dos elementos que contiene las dimensiones de la grilla a ser muestreada.</p> <p><i>ENUM</i>: valor que indica el porcentaje de la grilla a ser muestreado.</p> <p><i>MASK</i>: matriz que enmascara la grilla. Los puntos de la grilla donde Mask sea igual a -1 no serán muestreados.</p>
<p>Salidas</p> <p><i>MI</i>: matriz con dimensiones dadas por <i>DSIZE</i>, que contiene 1 en los puntos de la grilla que serán muestreados y 0 en los restantes.</p>
<p>Descripción</p> <p>Función que genera una matriz de muestreo aleatoria. Genera una matriz aleatoria con las dimensiones dadas por <i>DSIZE</i>, y que contiene 1 para los puntos de la grilla a ser muestreados y 0 para los restantes valores.</p>
<p>Llamadas a otras funciones</p> <p>-</p>

buildDefaultPranaJob.m

<p>Entradas</p> <p>-</p>
<p>Salidas</p> <p><i>defaultdata</i>: variable tipo struct con los campos dados por la Tabla 2.</p>
<p>Descripción</p> <p>Esta función crea un conjunto de parámetros por default y los carga en la variable tipo struct denominada <i>defaultdata</i>. Estos son los parámetros que se cargan por defecto en la interfaz de usuario al crear un nuevo trabajo.</p>
<p>Llamadas a otras funciones</p> <p>-</p>

energyfilt.m

Entradas

Nx: extensión en píxeles de la ventana de correlación en la dimensión *x*.

Ny: extensión en píxeles de la ventana de correlación en la dimensión *y*.

d: arreglo de dos elementos que especifican las dimensiones en píxeles de partícula en *x* y en *y*.

q: factor de peso (opcional).

Salidas

W: matriz de dimensiones *Nx* y *Ny* que constituye el filtro espectral.

Descripción

Función que calcula un filtro espectral teniendo en cuenta el diámetro de las partículas, las dimensiones del plano de correlación y un factor de peso *q*. Este filtro es aplicado en los métodos de correlación “Robust Phase Correlation” y “Dynamic Robust Phase Correlation”.

Llamadas a otras funciones

-

findwidth.m

Entradas

r: cociente entre la resolución de la ventana (tamaño de ventana efectiva) y el tamaño de la ventana de correlación.

Salidas

p: factor de escala para la ventana gaussiana.

Descripción

Función que recibe el cociente entre el tamaño de ventana efectiva y el tamaño de ventana de correlación, y devuelve el factor de escala a utilizar por la función *windowmask.m* para la generación de la función de ventaneo gaussiana. La función calcula el factor de forma como $p = (1.224/r)^{1.0172}$.

Llamadas a otras funciones

-

gridfit.m

Entradas

x: arreglo columna que contiene la componente horizontal de puntos dispersos.

y: arreglo columna que contiene la componente vertical de puntos dispersos.

z: arreglo columna que contiene los valores de una cantidad evaluada en los puntos dados por las coordenadas *x* e *y*.

xnodes: arreglo columna que contiene la componente horizontal de una grilla regular.

ynodes: arreglo columna que contiene la componente vertical de una grilla regular.

Salidas

zgrid: matriz que contiene los valores de la superficie ajustada a los puntos (*x*, *y*, *z*), evaluados en los puntos de grilla dados por *xgrid* y *ygrid*.

xgrid: matriz que contiene las coordenadas horizontales de la grilla definida por *xnodes* e *ynodes*.

ygrid: matriz que contiene las coordenadas verticales de la grilla definida por *xnodes* e *ynodes*.

Descripción

Función que toma un conjunto de puntos dispersos, y genera una superficie suave en una grilla regular, que aproxima al valor de los puntos dispersos.

Llamadas a otras funciones

-

IMgrid.m

Entradas

L: debe ser un arreglo de dos elementos que contenga las dimensiones en píxeles de las imágenes a ser procesadas (Primer elemento para la dimensión vertical y segundo elemento para la dimensión horizontal).

S: debe ser un arreglo de dos elementos que contenga la resolución en píxeles de la grilla a emplear para calcular el campo de velocidades (Primer elemento para la dimensión vertical y segundo elemento para la dimensión horizontal).

G: debe ser un arreglo de cuatro elementos que indique el comienzo y final de la grilla, en píxeles (opcional). El primer y tercer elementos representan el comienzo y final de la grilla en la dirección vertical, respectivamente, mientras que el segundo y cuarto elementos representan el comienzo y final de la grilla en la dirección horizontal.

Salidas

X: arreglo de 2 dimensiones que constituye la componente *x* de la grilla, en píxeles, donde serán calculados los vectores velocidad.

Y: arreglo de 2 dimensiones que constituye la componente *y* de la grilla, en píxeles, donde serán calculados los vectores velocidad.

Descripción

Esta función calcula, en píxeles, los puntos de la grilla *X* e *Y* donde serán evaluados los vectores velocidad, a partir de la dimensión de las imágenes, la resolución de la grilla y el buffer de grilla (indica las coordenadas en píxeles donde se comienza la grilla, a partir del borde de la región de interés). Si el buffer de grilla no se especifica o se especifica como [0,0], se usa el espaciamento de grilla como buffer.

Llamadas a otras funciones

-

jobfile_validator.m

Entradas

Data: variable tipo struct con los campos dados en la Tabla 2.

Salidas

Data: variable tipo struct con los campos dados en la Tabla 2.

Descripción

Esta función toma los datos ingresados por el usuario, almacenados en la estructura *Data*, y los valida, garantizando que estos sean aptos para la ejecución de todo el programa. Los datos no válidos son reemplazados por valores estándar.

Llamadas a otras funciones

-

leastsquares2D.m

Entradas

x: arreglo de 6 elementos que contiene los valores de los parámetros del modelo de ajuste gaussiano (valor máximo, factor de escala en la dirección *x*, factor de escala en la dirección *y*, centroide en *x*, centroide en *y*, ángulo de orientación).

mapint_i: arreglo que contiene los valores de las mediciones a ajustar, realizadas en los puntos dados por *locxy_i*.

locxy_i: matriz de dos columnas, con tantas filas como elementos contiene *mapint_i*.

Cada fila representa las componentes *x* e *y* de un punto de medición.

method: valor que especifica el método de ajuste. 3 – Standard Gaussian; 4 – Continuous Gaussian.

Salidas

F: arreglo con la misma cantidad de elementos que *mapint_i*, que contiene las diferencias entre el valor de las mediciones y el valor obtenido mediante el ajuste gaussiano.

Descripción

Función que toma los parámetros de un modelo de ajuste Gaussiano, las coordenadas de puntos de medición y los valores medidos para la señal en dichos puntos, y devuelve un vector con las diferencias entre los valores medidos y los predichos por el modelo gaussiano sujeto a los parámetros dados de entrada.

Llamadas a otras funciones

-

matrixform.m

Entradas

x: arreglo columna que contiene la componente *x* de los puntos de grilla a ser procesados, pero ordenados en forma de lista.

y: arreglo columna que contiene la componente *y* de los puntos de grilla a ser procesados, pero ordenados en forma de lista.

u: matriz con la misma cantidad de filas que *x* e *y*, que contiene los desplazamientos en la componente *x* en píxeles, con precisión subpíxel, y calculados por el método PIV, para las posiciones dadas por los arreglos de *x* e *y*, y ordenados en forma de lista. Si se determinaron los desplazamientos para los tres picos máximos de correlación, estos se agrupan en tres columnas, de lo contrario *u* posee una sola columna.

v: idéntico a *u*, pero para la componente de desplazamiento en *y*.

eval: matriz de las mismas dimensiones que *u*. Los valores para los elementos de esta matriz vienen dados por los de la Tabla 4.

c: matriz con las mismas dimensiones que *u*, que contiene la magnitud del pico máximo de correlación, o bien de los tres picos máximos de correlación dispuestos en tres columnas.

d: matriz con las mismas dimensiones que *u*, que contiene el ancho estimado del pico máximo de correlación, o bien el ancho estimado de los tres picos máximos de correlación dispuestos en tres columnas.

Salidas

X: matriz que contiene los elementos de *x* dispuestos en forma de grilla.

<p><i>Y</i>: matriz que contiene los elementos de <i>y</i> dispuestos en forma de grilla.</p> <p><i>U</i>: matriz que contiene los elementos de <i>u</i> dispuestos en forma de grilla. Si <i>u</i> tiene tres columnas, la grilla formada por cada columna se dispone en una matriz <i>U</i> tridimensional.</p> <p><i>V</i>: matriz que contiene los elementos de <i>v</i> dispuestos en forma de grilla. Si <i>v</i> tiene tres columnas, la grilla formada por cada columna se dispone en una matriz <i>V</i> tridimensional.</p> <p><i>Eval</i>: matriz que contiene los elementos de <i>eval</i> dispuestos en forma de grilla.</p> <p><i>C</i>: matriz que contiene los elementos de <i>c</i> dispuestos en forma de grilla. Si <i>c</i> tiene tres columnas, la grilla formada por cada columna se dispone en una matriz <i>C</i> tridimensional.</p> <p><i>D</i>: matriz que contiene los elementos de <i>d</i> dispuestos en forma de grilla. Si <i>d</i> tiene tres columnas, la grilla formada por cada columna se dispone en una matriz <i>D</i> tridimensional.</p>
<p>Descripción</p> <p>Función que toma los argumentos de entrada dados en forma de lista, y los lleva a forma de grilla.</p>
<p>Llamadas a otras funciones</p> <p>-</p>

PIVphasecorr.m

<p>Entradas</p> <p><i>im1</i>: la primera imagen del par a ser procesada, en forma de matriz. Se debe ingresar con la componente <i>i</i> (dimensión vertical) invertida.</p> <p><i>im2</i>: la segunda imagen del par a ser procesada, en forma de matriz. Se debe ingresar con la componente <i>i</i> (dimensión vertical) invertida.</p> <p><i>window</i>: arreglo de dos elementos que indica las dimensiones en píxeles de la ventana total de interrogación (Actual Window Size). El primer elemento indica la dimensión horizontal, mientras que el segundo elemento indica la dimensión vertical.</p> <p><i>res</i>: matriz de 2x2 que indica la resolución en píxeles de las ventanas en la primera y segunda imagen. La primera fila indica las dimensiones horizontal y vertical de la primera ventana, mientras que la segunda fila indica las dimensiones horizontal y vertical de la segunda ventana (effective window resolution).</p> <p><i>zpad</i>: indica si se aplica un “zero-padding”, haciendo que las ventanas correlacionadas tengan el doble de sus dimensiones originales, rellendo estas con ceros. El “zero-padding” se aplica si <i>zpad</i> es distinto de 0. Si bien la función ofrece esta opción, a través de la interfaz de Prana no es posible establecer el valor de <i>zpad</i>, y el programa siempre usa <i>zpad</i> = 0.</p> <p><i>D</i>: arreglo de dos elementos que contiene las dimensiones en píxeles de la partícula.</p> <p><i>Zeromean</i>: valor 0 o 1, que indica si se aplica una normalización a las ventanas restándole su media.</p> <p><i>Peakswitch</i>: Valor 0 o 1. Indica si se guarda información de picos adicionales.</p> <p><i>X</i>: Matriz de dos dimensiones con las componentes en <i>x</i> de la grilla, dadas en píxeles, sobre las que se calcularán los desplazamientos. También se puede ingresar como arreglo columna con los elementos dispuestos en forma de lista.</p> <p><i>Y</i>: Matriz de dos dimensiones con las componentes en <i>y</i> de la grilla, dadas en píxeles, sobre las que se calcularán los desplazamientos. También se puede ingresar como arreglo columna con los elementos dispuestos en forma de lista.</p>

Uin: arreglo columna que especifica la componente x de desplazamiento, en píxeles, en los puntos de la grilla a ser evaluados. Se utiliza para aplicar desplazamiento de ventana discreto.

Vin: arreglo columna que especifica la componente y de desplazamiento, en píxeles, en los puntos de la grilla a ser evaluados. Se utiliza para aplicar desplazamiento de ventana discreto.

dt: intervalo de tiempo entre pulsos láser, dado en microsegundos (opcional). Si no se especifica se adopta el valor 1.

Salidas

X: arreglo columna que contiene los elementos de la matriz X de entrada, pero reagrupados en forma de lista.

Y: arreglo columna que contiene los elementos de la matriz Y de entrada, pero reagrupados en forma de lista.

U: arreglo columna que contiene los desplazamientos en la componente x en píxeles, con precisión subpíxel, y calculados por el método PIV, para las posiciones dadas por los arreglos de salida X e Y .

V: idéntico a U , pero para la componente de desplazamiento en y .

C: arreglo columna que contiene las magnitudes de los picos máximos de correlación. Si *Peakswitch* vale 1, devuelve en su lugar una matriz en el que se disponen en sus columnas las amplitudes de los tres picos máximos de correlación.

t_opt: arreglo columna que indica el canal de color óptimo para la estimación del desplazamiento en cada punto.

Descripción

Función que toma un par de imágenes, el conjunto de puntos de la grilla en los que se evaluarán los desplazamientos, y las resoluciones de ventana, para estimar los desplazamientos aplicando el método de correlación denominado "Spectral Phase Correlation", el cual se basa en una estimación del desplazamiento sobre el dominio espectral, por medio de descomposición de valores singulares. Opcionalmente, se pueden incorporar como entrada el campo de velocidades supuesto o estimado en un paso anterior, para que la función aplique un desplazamiento discreto sobre las ventanas a correlacionar, y mejorar la relación señal-ruido.

Llamadas a otras funciones

energyilt.m

windowmask.m

wlsq.m

PIVwindowed.m

Entradas

im1: la primera imagen del par a ser procesada, en forma de matriz. La imagen se debe ingresar con la componente i (dimensión vertical) invertida.

im2: la segunda imagen del par a ser procesada, en forma de matriz. La imagen se debe ingresar con la componente i (dimensión vertical) invertida.

tcorr: tipo de esquema de correlación ('SCC', 'RPC', 'DRPC', 'GCC', 'FWC', 'SPC', 'DCC').

window: arreglo de dos elementos que indica las dimensiones en píxeles de la ventana de interrogación.

res: matriz de 2x2 que indica la resolución en píxeles de las ventanas en la primera y segunda imagen. La primera fila indica las dimensiones horizontal y vertical de la primera ventana, mientras que la segunda fila indica las dimensiones horizontal y vertical de la segunda ventana.

zpad: indica si se aplica un zero-padding, haciendo que las ventanas correlacionadas tengan el doble de sus dimensiones originales, relleno con ceros. El zero-padding se aplica si *zpad* es distinto de 0. Si bien la función ofrece esta opción, a través de la interfaz de Prana no es posible establecer el valor de *zpad*, y el programa siempre usa *zpad* = 0.

D: arreglo de dos elementos que contiene las dimensiones en píxeles de la partícula.

Zeromean: valor 0 o 1, que indica si se aplica una normalización a las ventanas restándole su media.

Peaklocator: valor (1, 2 o 3), que indica el método a utilizar para el cálculo del desplazamiento con precisión subpíxel. 1 – Estimación gaussiana de tres puntos; 2 – Estimación gaussiana de cuatro puntos; 3 – Ajuste gaussiano de mínimos cuadrados.

Peakswitch: Valor 0 o 1. Indica si se guarda información de picos adicionales.

Fracval: Valor fraccional para el método de “Fraction-Filter Weighted Correlation”.

Saveplane: Valor 0 o 1. Indica si se guardan los planos de correlación.

X: matriz de dos dimensiones con las componentes en *x* de la grilla, dadas en píxeles, sobre las que se calcularán los desplazamientos. También se puede ingresar como arreglo de una dimensión con los elementos dispuestos en columna.

Y: matriz de dos dimensiones con las componentes en *y* de la grilla, dadas en píxeles, sobre las que se calcularán los desplazamientos. También se puede ingresar como arreglo de una dimensión con los elementos dispuestos en columna.

Uin: arreglo columna que especifica la componente *x* de desplazamiento, en píxeles, en los puntos de la grilla a ser evaluados. Se utiliza para determinar el centro de las ventanas con una aproximación de segundo orden. (Opcional).

Vin: arreglo columna que especifica la componente *y* de desplazamiento, en píxeles, en los puntos de la grilla a ser evaluados. Se utiliza para determinar el centro de las ventanas con una aproximación de segundo orden. (Opcional).

Salidas

X: arreglo columna que contiene los elementos de la matriz *X* de entrada, pero reagrupados en forma vectorial.

Y: arreglo columna que contiene los elementos de la matriz *Y* de entrada, pero reagrupados en forma vectorial.

U: arreglo columna que contiene los desplazamientos en la componente *x* en píxeles, con precisión subpíxel, y calculados por el método PIV, para las posiciones dadas por *X* e *Y*. Si la variable *Peakswitch* vale 1, devuelve en su lugar una matriz de tres columnas, en donde en las columnas se disponen los desplazamientos calculados para los tres picos máximos del plano de correlación, con precisión subpíxel, para cada punto de la grilla.

V: idéntico a *U*, pero para la componente de desplazamiento en *y*.

C: arreglo columna que contiene las magnitudes de los picos máximos de correlación. Si *Peakswitch* vale 1, devuelve en su lugar una matriz de tres columnas en el que se disponen en sus columnas las amplitudes de los tres picos máximos de correlación para cada punto de grilla. Si tanto *Peakswitch* como *saveplane* son 0, devuelve un arreglo vacío.

Dia: arreglo columna que contiene los anchos estimados, en píxeles, de los picos máximos de correlación. Si *Peakswitch* vale 1, devuelve en su lugar una matriz de dos dimensiones en el que se disponen en sus columnas los anchos estimados de los tres

picos máximos de correlación. Si tanto *Peakswitch* como *saveplane* son 0, devuelve un arreglo vacío.

Corrplanes: Si *saveplane* vale 1, devuelve un arreglo de tres dimensiones en el que se almacenan todos los planos de correlación. De lo contrario, devuelve un 0.

Descripción

Función que toma un par de imágenes, el conjunto de puntos de la grilla en los que se evaluarán los desplazamientos, y las resoluciones de ventana, para estimar los desplazamientos aplicando el método de correlación dado por *tcrr*. Opcionalmente, se pueden incorporar como entrada el campo de velocidades supuesto o estimado en un paso anterior, para que la función aplique un desplazamiento discreto sobre las ventanas a correlacionar.

Llamadas a otras funciones

windowmask.m
energyfilt.m
subpixel.m

pranaPIVcode.m

Entradas

Data: variable tipo struct con los campos dados en Tabla 2.

Salidas

-

Descripción

Función que prepara al sistema para el procesamiento PIV. Valida los datos cargados en *Data* por medio de la función *jobfile_validator*. Genera un archivo de texto con los parámetros del experimento a través de la función *write_expsummary.m*. Configura el uso de programación en paralelo, y finalmente le pasa la estructura *Data* a *pranaprocessing.m* para comenzar con el procesamiento de las imágenes. Prana solo llama a la función *pranaPIVcode.m* si está activada la opción "Run Particle Image Velocimetry".

Llamadas a otras funciones

jobfile_validator.m
write_expsummary.m
pranaprocessing.m

pranaprocessing.m

Entradas

Data: variable tipo struct con los campos dados en Tabla 2.

I1: arreglo fila que contiene los índices de la primera imagen de cada par a ser procesado (opcional). Si no se especifica se extrae de *Data*.

I2: arreglo fila que contiene los índices de la segunda imagen de cada par a ser procesado (opcional). Si no se especifica se extrae de *Data*.

maskname: arreglo fila que contiene los índices de las máscaras a ser aplicadas para el método de máscara dinámica (opcional). Esta entrada se encuentra obsoleta debido a que, dentro de la función, *maskname* siempre se redefine a partir de *Data*.

Salidas

Archivo .m con los resultados del procesamiento.

Descripción

Esta función es la que ejecuta el procesamiento PIV de las imágenes según las especificaciones ingresadas por el usuario en la estructura de entrada *Data*. La función lee *Data* y carga los parámetros que contiene en distintas variables. Luego llama a las distintas subrutinas para realizar el procesamiento PIV.

Llamadas a otras funciones

IMgrid.m
PIVwindowed.m
PIVphasecorr.m
VAL.m
Matrixform.m
VFinterp.m
sincBlackmanInterp2.m
PIVensemble.m
VELfilt.m
Write_dat_val_C.m

sincBlackmanInterp2.m

Entradas

Z: matriz a ser deformada.
XI: matriz que contiene los componentes *x* de la grilla deformada.
YI: matriz que contiene los componentes *y* de la grilla deformada.
KERNELRADIUS: Valor que indica el radio del Kernel.
Method: Cadena de caracteres que indica el método a emplear ('Sinc' o 'Blackman').

Salidas

ZI: matriz deformada.

Descripción

Función que toma una señal contenida en la matriz *Z*, y la interpola a una grilla deformada.

Llamadas a otras funciones

-

subpixel.m

Entradas

G: matriz que constituye el plano de correlación sobre el cual se quiere estimar el desplazamiento.
ccsizex: tamaño del plano de correlación en la dimensión *x*.
ccsizey: tamaño del plano de correlación en la dimensión *y*.
W: matriz que tiene el mismo tamaño que *G* y que se aplica como matriz de pesos sobre *G*. Desde Prana esta matriz siempre es asignada con unos.
Method: valor (1, 2 o 3), que indica el método a utilizar para el cálculo del desplazamiento con precisión subpíxel. 1 – Estimación gaussiana de tres puntos; 2 – Estimación gaussiana de cuatro puntos; 3 – Ajuste gaussiano de mínimos cuadrados.
Peakswitch: Valor 0 o 1. Indica si se guarda información de picos adicionales.

d: arreglo de dos elementos que especifican el tamaño de la partícula.

Salidas

u: desplazamiento estimado sobre la componente *x*. Devuelve el desplazamiento en píxeles con respecto a la componente *x*, con precisión subpíxel, calculado a partir del mayor pico de correlación del plano de correlación *G*, mediante el método dado por *Method*. Si *Peakswitch* vale 1, devuelve en su lugar un arreglo fila de tres elementos, que contienen los desplazamientos con precisión subpíxel para los tres máximos picos de *G*.
v: idéntico a *u*, pero para la componente de desplazamiento en *y*.

M: amplitud del pico máximo de correlación. Si *Peakswitch* vale 1, devuelve en su lugar un arreglo fila de tres elementos que contiene la amplitud de los tres mayores picos de correlación.

D: Devuelve el ancho estimado del pico de correlación. Si *Peakswitch* vale 1, devuelve en su lugar un arreglo fila de tres elementos que contiene el ancho estimado de los tres mayores picos de correlación.

Descripción

Función que toma un plano de correlación obtenido por medio de algún esquema de correlación, para calcular el desplazamiento en *x* y en *y*, en píxeles, a partir del máximo pico de correlación, con precisión subpíxel. El método empleado para la estimación puede ser ajuste gaussiano de tres puntos, ajuste gaussiano de cuatro puntos, o ajuste gaussiano 2D a partir de mínimos cuadrados no lineales. En el último caso se aplica el algoritmo de Levenberg-Marquardt.

Llamadas a otras funciones

leastsquares2D.m

Thresh.m

Entradas

U: Matriz que contiene los valores de desplazamiento estimados en la componente *x*, dispuestos en forma de grilla.

V: Matriz que contiene los valores de desplazamiento estimados en la componente *y*, dispuestos en forma de grilla.

uthreshold: arreglo de dos elementos que contiene el valor mínimo y máximo de umbral de desplazamiento en la dirección *x*, para el método de validación por “threshold”.

vthreshold: arreglo de dos elementos que contiene el valor mínimo y máximo de umbral de desplazamiento en la dirección *y*, para el método de validación por “threshold”.

Eval: matriz de iguales dimensiones que *U*, cuyos elementos vienen determinados a partir de la Tabla 4.

Salidas

Uval: matriz de las mismas dimensiones de *U*, que contiene los desplazamientos en la dirección *x* ya validados.

Vval: matriz de las mismas dimensiones que *V*, que contiene los desplazamientos en la dirección *y* ya validados.

Eval: matriz *Eval* validada. La matriz toma distintos valores según como haya pasado el punto de grilla el proceso de validación. -1 – el punto de grilla no fue procesado debido a que estaba enmascarado; 0 – el punto de grilla fue procesado y pasó la validación; 100 – el punto de grilla fue procesado, pero falló en la validación por Thresholding.

Descripción

Función que toma las matrices de velocidades estimadas *U* y *V*, y compara sus valores con los valores de umbral especificados por el usuario. Si alguno de los dos valores *U* o

V en un determinado punto de la grilla cae fuera de los umbrales, los valores de U y V en dicho punto se reemplazan por “NaN”, y el valor de $Eval$ en el punto se reemplaza por 100 para indicar que no pasó la validación por Thresholding.

Llamadas a otras funciones

-

UOD.m

Entradas

U : Matriz que contiene los valores de desplazamiento estimados en la componente x , dispuestos en forma de grilla.

V : Matriz que contiene los valores de desplazamiento estimados en la componente y , dispuestos en forma de grilla.

t : matriz de dos columnas que contiene el tamaño de ventanas para cada paso del método de Universal Outlier Detector (UOD). La primera columna se corresponde con la resolución en x , mientras que la segunda columna se corresponde con la resolución en y . Distintas filas se usan para sucesivos pasos de validación por UOD.

tol : arreglo columna que indica la tolerancia para el método UOD. Cada elemento indica la tolerancia para un paso sucesivo de validación por UOD.

$Eval$: matriz de iguales dimensiones que U , cuyos elementos vienen determinados a partir de la Tabla 4.

Salidas

$Uval$: matriz de las mismas dimensiones de U , que contiene los desplazamientos en la dirección x ya validados.

$Vval$: matriz de las mismas dimensiones que V , que contiene los desplazamientos en la dirección y ya validados.

$Eval$: matriz $Eval$ validada. La matriz toma distintos valores según como haya pasado el punto de grilla el proceso de validación. -1 – el punto de grilla no fue procesado debido a que estaba enmascarado; 0 – el punto de grilla fue procesado y pasó la validación; 1 – el punto de grilla fue procesado, pero falló en la validación UOD en el paso 1; k – el punto de grilla fue procesado, pero falló en la validación UOD en el paso k .

Descripción

Función que aplica el método UOD para la validación de los vectores desplazamiento calculados. La función efectúa un paso de validación por cada fila que contenga t . Se recorren todos los puntos de la grilla en donde $Eval = 0$ (puntos no enmascarados donde se calculó el desplazamiento), y se toma una ventana en torno a ese punto, con las dimensiones dadas por t . Se cuentan la cantidad de valores válidos dentro de esta ventana (distintos a NaN). Si la cantidad de valores válidos disponibles es menor que 8, se incrementa el tamaño de la ventana y se vuelve a comprobar, hasta que se tengan al menos 8 valores válidos. Luego, se calcula la mediana dentro de la ventana y se calcula el desvío con respecto a la mediana normalizado. Este se compara con el valor de tolerancia. Si alguno de los dos valores U o V para el punto analizado da un desvío por encima de la tolerancia, estos valores de U y V se reemplazan por NaN, y el valor de $Eval$ se reemplaza por el número de paso en el cual la validación falló. La subrutina $UOD_sub.m$ se utiliza para calcular los residuos.

Llamadas a otras funciones

$UOD_sub.m$

<p>Entradas</p> <p><i>W</i>: matriz que contiene el valor a ser validado y sus valores vecinos. <i>p</i>: número de fila en el que se encuentra el valor a ser validado dentro de la ventana <i>W</i>. <i>q</i>: número de columna en el que se encuentra el valor a ser validado dentro de la ventana <i>W</i>.</p>
<p>Salidas</p> <p><i>R</i>: Valor que representa el desvío con respecto a la mediana normalizado.</p>
<p>Descripción</p> <p>Función que toma una ventana que contiene el valor a ser validado y los valores vecinos, y calcula el desvío del valor con respecto a la mediana normalizado.</p>
<p>Llamadas a otras funciones</p> <p>-</p>

<p>Entradas</p> <p><i>X</i>: arreglo columna con las componentes en <i>x</i> de la grilla, dadas en píxeles, sobre las que se calculará el campo de velocidades. <i>Y</i>: arreglo columna con las componentes en <i>y</i> de la grilla, dadas en píxeles, sobre las que se calculará el campo de velocidades. <i>U</i>: matriz con la misma cantidad de filas que <i>X</i> e <i>Y</i>, que contiene las componentes de desplazamiento en <i>x</i> estimadas. Contiene tres columnas si se estimó el desplazamiento a partir de picos adicionales (opciones “try additional peaks” y “save additional peak information”). <i>V</i>: matriz con las mismas dimensiones que <i>U</i>, que contiene las componentes de desplazamiento en <i>y</i> estimadas. <i>Eval</i>: arreglo columna, del mismo tamaño que <i>X</i>, que contiene un 0 en los puntos en los que se evaluó el vector desplazamiento, y -1 en los puntos enmascarados que no fueron evaluados. <i>C</i>: matriz con las mismas dimensiones de <i>U</i>, que contiene las magnitudes de los picos máximos de correlación obtenidos en cada punto de la grilla (0 en los puntos enmascarados). <i>D</i>: matriz con las mismas dimensiones de <i>U</i>, en cuyas columnas se disponen los anchos estimados de los picos de máxima correlación obtenidos en cada punto de la grilla (0 en los puntos enmascarados). <i>Threshswitch</i>: valor 0 o 1, que indica si se validan los desplazamientos por el método de “Threshold”. <i>UODswitch</i>: valor 0 o 1, que indica si se validan los desplazamientos por el método de “Universal Outlier Detection” (UOD). <i>Bootswitch</i>: valor 0 o 1, que indica si se validan los desplazamientos por el método de “Bootstrapping”. <i>extrapeaks</i>: valor 0 o 1, que indica si se guarda información de picos adicionales. <i>Uthresh</i>: arreglo de dos elementos que contiene el valor mínimo y máximo de umbral de desplazamiento en la dirección <i>x</i>, para el método de “threshold”. <i>Vthresh</i>: arreglo de dos elementos que contiene el valor mínimo y máximo de umbral de desplazamiento en la dirección <i>y</i>, para el método de “threshold”.</p>

UODwinsize: arreglo de $1 \times 2 \times 2$ que contiene el tamaño de ventanas para el método de UOD.

UODthresh: arreglo de dos elementos que contiene los valores de umbral para el método UOD.

Bootper: porcentaje de muestreo para el método de “Bootstrapping”.

Bootiter: cantidad de iteraciones por cuadro para el método de “Bootstrapping”.

Bootkmax: Número de pasadas para el método de “Bootstrapping”.

Salidas

Uval: arreglo con la misma longitud de X , que contiene los desplazamientos en la dirección x ya validados.

Vval: arreglo con la misma longitud de X , que contiene los desplazamientos en la dirección x ya validados.

Evalval: matriz *Eval* validada. La matriz toma distintos valores según como haya pasado el punto de grilla el proceso de validación. -1 – el punto de grilla no fue procesado debido a que estaba enmascarado; 0 – el punto de grilla fue procesado y pasó la validación; 1 – el punto de grilla fue procesado, pero falló en la validación UOD en el paso 1; k – el punto de grilla fue procesado, pero falló en la validación UOD en el paso k ; 100 – el punto de grilla fue procesado, pero falló en la validación por Thresholding; 200 – el punto de grilla fue procesado, pero fallo en la validación de Bootstrapping.

Cval: matriz C validada.

Dval: matriz D validada.

Descripción

Función que toma los valores de desplazamiento calculados y los valida, aplicando el método de Thresholding y/o el método UOD. Los valores detectados como outliers son reemplazados por una media ponderada de los valores vecinos al outlier, utilizando como factor de peso la distancia del valor vecino con respecto al valor a ser reemplazado. Finalmente, si *Bootswitch* vale 1, se efectúa una validación de Bootstrapping.

Llamadas a otras funciones

matrixform.m

Thresh.m

UOD.m

bootstrapping.m

vectorform.m

vectorform.m

Entradas

x : matriz que contiene las coordenadas x en píxeles de los puntos de la grilla, dado en forma de grilla.

y : matriz que contiene las coordenadas y en píxeles de los puntos de la grilla, dado en forma de grilla.

U : matriz que contiene los desplazamientos en la componente x en píxeles, con precisión subpíxel, y calculados por el método PIV, dispuestos en forma de grilla.

V : matriz que contiene los desplazamientos en la componente y en píxeles, con precisión subpíxel, y calculados por el método PIV, dispuestos en forma de grilla.

Eval: matriz con las mismas dimensiones que u .

C : matriz que contiene en sus columnas la magnitud de los tres picos máximos de correlación, dispuestos en forma de grilla.

D: matriz que contiene los elementos de *d* dispuestos en forma de grilla.

Salidas

u: arreglo columna con los elementos de *U* en forma de lista.

v: arreglo con los elementos de *V* en forma de lista.

eval: arreglo columna con los elementos de *Eval* en forma de lista.

c: arreglo columna con los elementos de *C* en forma de lista.

d: arreglo columna con los elementos de *D* en forma de lista.

Descripción

Función que toma los argumentos de entrada dados en forma de grilla, y los lleva a forma de lista.

Llamadas a otras funciones

-

VELfilt.m

Entradas

U: matriz que contiene los componentes *x* de velocidad del campo de velocidad estimado por PIV.

V: matriz que contiene los componentes *y* de velocidad del campo de velocidad estimado por PIV.

H: dimensiones de la ventana gaussiana.

C: factor de escala la ventana gaussiana.

Salidas

Uf: matriz *U* suavizada.

Vf: matriz *V* suavizada.

Descripción

Función que aplica una ventana de suavizado gaussiana de dimensiones dadas por *H* y desvío estándar *C* sobre las matrices *U* y *V*. Prana aplica como dimensiones *H* las mismas que se usan para el método UOD.

Llamadas a otras funciones

-

VFinterp.m

Entradas

X: matriz que contiene los componentes *x* de la grilla original.

Y: matriz que contiene los componentes *y* de la grilla original.

Z: matriz que contiene los valores de una cantidad evaluada en los puntos de la grilla original.

XI: matriz que contiene los componentes *x* de la grilla final.

YI: matriz que contiene los componentes *y* de la grilla final.

M: Valor que indica el método de interpolación a emplear. 1 – Nearest Neighbor; 2 – linear; 3 – Spline.

Salidas

ZI: matriz que contiene los valores de la matriz *Z* interpolados a la grilla definida por las matrices *XI* y *YI*.

Descripción

Función que interpola los valores de una matriz Z sobre una grilla dada por X , Y , y los lleva a una nueva grilla definida por XI y YI .

Llamadas a otras funciones

-

windowmask.m

Entradas

N : arreglo de dos elementos que especifican el tamaño (en píxeles) de la ventana de correlación. El primer elemento indica el tamaño en la dimensión horizontal, mientras que el segundo especifica el tamaño en la dimensión vertical.

R : arreglo de dos elementos que especifican el tamaño de ventana efectivo (en píxeles). El primer elemento indica el tamaño en la dimensión horizontal, mientras que el segundo especifica el tamaño en la dimensión vertical.

Salidas

W : matriz que representa una función de ventaneo gaussiana.

Descripción

Función que genera una función de ventaneo gaussiana de tamaño total dado por N , y con desvío parámetro de escala determinado por N y R a partir de la función *findwidth*.

Llamadas a otras funciones

findwidth.m

wlsq.m

Entradas

y : arreglo de una dimensión que contiene los valores a ser ajustados.

H : arreglo de una dimensión que contiene las coordenadas.

W : arreglo de pesos.

Salidas

Xh : pendiente del ajuste lineal.

Descripción

Función que realiza un ajuste por mínimos cuadrados para determinar el desplazamiento con precisión subpíxel para la correlación de fase. La función realiza el ajuste por mínimos cuadrados de los puntos H vs y , a la función $y = Xh H$, utilizando los pesos en W , y devuelve el valor Xh .

Llamadas a otras funciones

-

write_dat_val_C.m

Entradas

$fname$: variable de tipo char que incluye la ruta y nombre del archivo de salida.

X : matriz que contiene la coordenada horizontal de los puntos de grilla en donde se evaluó el campo de velocidades.

Y : matriz que contiene la coordenada vertical de los puntos de grilla en donde se evaluó el campo de velocidades.

U: matriz que contiene la componente de velocidad horizontal de los puntos de grilla en donde se evaluó el campo de velocidades.

V: matriz que contiene la componente de velocidad vertical de los puntos de grilla en donde se evaluó el campo de velocidades.

Eval: matriz de evaluación.

C: matriz que contiene el valor de los picos de máxima correlación evaluados en los puntos de grilla.

D: matriz que contiene el valor de los diámetros estimados para los picos de máxima correlación, evaluados en los puntos de grilla.

Strand: número de paso de método PIV.

T: tiempo asociado al par de imágenes.

Frametitle: Cadena de caracteres que representa el nombre del archivo de salida.

t_opt: No se usa. No se indica utilidad.

Salidas

Archivo con formato .dat que contiene los resultados del procesamiento.

Descripción

Función que toma las variables de entrada y las guarda en un archivo .dat.

Llamadas a otras funciones

-

write_expsummary.m

Entradas

Data: variable tipo struct con los campos dados en Tabla 2.

Salidas

Archivo con formato .txt que contiene el sumario de todos los parámetros empleados para el procesamiento.

Descripción

Esta función toma los datos introducidos por el usuario y validados, y los utiliza para crear un archivo de texto el cual contiene un sumario de todos los parámetros empleados en la ejecución del trabajo.

Llamadas a otras funciones

-