

**Medición de Campos de Velocidades utilizando Algoritmos
de Velocimetría por Imágenes de Partículas**

POR

Francisco Javier Martínez Farías

Facultad de Ciencias UNAM (2006)

TESIS

Sometido a la Universidad Nacional Autónoma de México
en cumplimiento parcial
de los requerimientos

Para obtener el grado

en

Física

Mayo 2006



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.



UNIVERSIDAD NACIONAL
AUTÓNOMA DE
MÉXICO

ACT. MAURICIO AGUILAR GONZÁLEZ
Jefe de la División de Estudios Profesionales de la
Facultad de Ciencias
Presente

Comunicamos a usted que hemos revisado el trabajo escrito:

"Medición de Campos de Velocidades utilizando Algoritmos de Velocimetría
por Imágenes de Partículas"

realizado por Martínez Farías Francisco Javier

con número de cuenta 09306828-8 , quien cubrió los créditos de la carrera de: Física.

Dicho trabajo cuenta con nuestro voto aprobatorio.

Atentamente

Director de Tesis

Propietario Dra. Catalina Elizabeth Stern Forgach *Catalina E Stern*

Propietario Dr. Arturo Olvera Chávez *Arturo Olvera*

Propietario Dr. Steven Peter Reed Czitrom Baus *Steven Czitrom*

Suplente Dr. Carlos Malaga Iguñiz *Carlos Malaga*

Suplente Dr. Gerardo Ruiz Chavarría *Gerardo Ruiz*

Consejo Departamental de Física



FACULTAD DE CIENCIAS
DEPARTAMENTO DE FÍSICA

Alicia Zarzosa Pérez
M. EN C. ALICIA ZARZOSA PEREZ

Dedico esta tesis a todas las personas que directa o indirectamente me han apoyado a lo largo de estos últimos años, en particular a mis padres y hermanos, a mi directora de tesis la Dra. Catalina E. Stern, a todos los miembros del laboratorio de Acústica, a mis sinodales, y por supuesto a todos mis amigos. Mi trabajo es también su trabajo.

Agradecimientos

Agradezco a la UNAM por haberme brindado la oportunidad de cursar mis estudios como físico en la Facultad de Ciencias. De manera muy especial hago un agradecimiento a cada uno de los integrantes del laboratorio de Acústica por el apoyo en equipo, recursos pero sobre todo por el apoyo moral y profesional que siempre recibí. A mi directora de tesis por su tiempo, paciencia y ayuda en todo momento. Agradezco también a mis sinodales por sus sugerencias y correcciones para este trabajo que me fueron de gran importancia al mostrarme aspectos que no había considerado. Al proyecto SEMARNAT-CONACYT *Ventilación del Puerto de Encenada mediante un Sistema de Bombeo por Energía de Oleaje (SIBEO)* por la beca otorgada para realizar esta tesis. También a mi familia por su apoyo incondicional siempre que lo solicité.

Índice General

.....	III
Agradecimientos	IV
Resumen	X
1. Introducción - Motivación	1
2. Velocimetría Por Imágenes de Partículas	4
2.1. ¿Qué es PIV ?	6
2.2. Aplicación del PIV	12
2.3. Determinación de campos de velocidades	17
2.4. PIV con SCILAB	20
2.5. Cálculo del campo de vorticidades a partir del campo de velocidades	25
3. Interpolación y visualización de campos	29
3.1. Interpolación mediante <i>splines</i>	29
3.2. Visualización de campos mediante OpenDX	34
4. Resultados	39
4.1. PIV aplicado a la salida del tubo	39
4.2. Aplicación al difusor con terminación circular	44
4.3. Amortiguamiento de la oscilación de una columna de agua	47
5. Conclusiones y perspectivas	54
5.1. Conclusiones	54
5.2. Trabajo futuro	56

A. Código fuente del programa de PIV y de funciones asociadas	61
A.1. Programa principal	61
A.2. Función dpiv.sce	63
A.3. Función medianvalid.sce	65
A.4. Función subpixel.sce	67
A.5. Función interpola.sce	68
A.6. Función vorticidad.sce	68
A.7. Función magnitud.sce	69
A.8. Función imprime.sce	70
A.9. Función imprime2d.sce	71
B. Características del dispositivo experimental	73
B.1. Partes principales	73
B.2. El generador de movimiento	73
B.3. Tipo de Oscilación	75
B.4. Control del sistema	77

Índice de cuadros

3.1. Módulos utilizados para visualizar los datos de PIV	37
4.1. Ajustes exponenciales para los decaimientos	49
4.2. Parte exponencial para los decaimientos	51
4.3. Ajuste de segundo grado para los decaimientos	52

Índice de figuras

1-1. Esquema general del SIBEO	2
2-1. Imagen de la celda vista como función	8
2-2. Imagen de la celda bajo la transformada de Fourier	9
2-3. Transformada de Fourier para dos celdas de una misma región	10
2-4. Función de correlación para dos celdas	11
2-5. Salida del difusor	13
2-6. Esquema del dispositivo de PIV	14
2-7. Operaciones matriciales con imágenes	15
2-8. Representación matricial de una celda en una imagen	16
2-9. Comparación de una medida física con una virtual	18
2-10. Esquema del funcionamiento del código de PIV	23
2-11. Campo de velocidades para un flujo cortante	27
2-12. Gráfica de la vorticidad para un flujo cortante	27
3-1. Puntos unidos mediante un polinomio de cuarto grado	30
3-2. Puntos unidos mediante un spline de cuarto orden	31
3-3. Gráfica Para distintos niveles de interpolación	34
3-4. Campo de velocidades de flujo cortante interpolado	35
3-5. Interface de OpenDX , para importar datos	36
3-6. Interface de programación modular de OpenDX	37
3-7. Campo de velocidades con OpenDX	38
4-1. PIV de la salida del difusor cuando la altura es máxima	40
4-2. Vorticidad a la salida del tubo	41

4-3. Velocidad total vs tiempo	42
4-4. Campo de velocidades, donde la masa de agua en movimiento es máxima .	43
4-5. Gráfica de la vorticidad, donde la masa en movimiento es máxima	43
4-6. Difusor con terminación circular	45
4-7. Gráfica para la función V_{total} circular	45
4-8. PIV para el difusor con terminación circular	46
4-9. Gráfica de vorticidad para el difusor circular	46
4-10. Esquema del dispositivo para la libre oscilación	48
4-11. Posición de la columna de agua como función del tiempo	48
4-12. Ajuste exponencial al decaimiento de la columna de agua	50
4-13. Decaimiento de la posición h en un espacio semilogarítmico	52
4-14. Ajustes exponencial y de segundo grado para el decaimiento de la columna de agua	53
4-15. Ajustes exponencial y de segundo grado para el decaimiento de la columna de agua con $h_0 = 23,81 \text{ cm}$	53
B-1. Imagen del dispositivo experimental	74
B-2. Vista esquemática del dispositivo experimental	74
B-3. Vista esquemática del dispositivo que varía la amplitud	74
B-4. Detalle de la unión entre la biela y el pistón	75
B-5. Parámetros de oscilación	76
B-6. Parámetros de oscilación	77
B-7. Diagrama de conexión para los motores del dispositivo	78

RESUMEN

Medición de Campos de Velocidades utilizando Algoritmos de Velocimetría por Imágenes de Partículas

Por

Francisco Javier Martínez Farías
Universidad Nacional Autónoma de México, Mayo, 2006

El objetivo del presente trabajo es desarrollar un algoritmo de velocimetría por imágenes de partículas (**PIV**) para medir el campo de velocidades en función del tiempo a la entrada de un **SIBEO** (*Sistema de Bombeo por Energía de Oleaje*). Para simular el oleaje se produjo en el laboratorio un flujo oscilante con un pistón dentro de un tubo parcialmente sumergido en un tanque de agua. El agua fue teñida con pintura metálica. Se grabó en video el flujo a la salida del tubo. A partir de la digitalización del video se obtuvo el campo de velocidades en función del tiempo y de ahí el campo de vorticidad. Se utilizaron 2 geometrías diferentes en la entrada del tubo. La tesis describe el algoritmo de **PIV** y el montaje experimental. Además de estudiar el flujo oscilante en función de la amplitud y de la frecuencia de oscilación, se estudió el amortiguamiento de una columna de agua sin forzamiento. En todos los casos se analizan resultados y se presentan conclusiones y propuestas para trabajo futuro.

ABSTRACT

The goal of this work is the development of an algorithm for Particle Image of Velocimetry (**PIV**) to measure the velocity field as a function of time at the entrance of a **SIBEO** (*Sistema de Bombeo por Energía de Oleaje, (Wave Energy Sea Water Pump)*). To simulate the wave motion, an oscillating flow was produced by a piston inside a tube partially submerged in a water tank. The water was dyed with metallic paint and filmed with a video camera. The velocity field as a function of time was obtained from the digital video, and the vorticity field

was calculated from the velocity field. Two different geometries were used at the entrance of the tube. This work describes in detail the algorithm and the experimental setup. Besides the study of the oscillating flows as a function of amplitude and frequency, the damping of a free oscillating column was also measured. The results of all these experiments are analyzed and conclusions and proposals for future work are presented.

Capítulo 1

Introducción - Motivación

Este trabajo está motivado en el proyecto llamado **SIBEO**, por sus siglas *Sistema de Bombeo por Energía de Oleaje*, el cual busca, como su nombre lo indica, aprovechar la energía del oleaje.

El **SIBEO** sirve para bombear agua limpia de mar de zonas con oleaje a zonas aisladas o con poca circulación. Se puede usar para ventilar estos cuerpos de agua o como ayuda en la pesquería. Un pequeño **SIBEO** estuvo montado en la costa de Oaxaca con el fin de introducir larva de camarón a la laguna **EL LAGARTERO** cuya conexión con el mar estaba bloqueada por sedimentos. Actualmente se espera ventilar la bahía de Ensenada con varios *SIBEOS*. El presente trabajo es el desarrollo de una herramienta que permite estudiar a los flujos alrededor del **SIBEO** y así colaborar en la optimización de su funcionamiento.

Antecedentes

El **SIBEO** está compuesto de un tubo resonante, una cámara de compresión y un tubo de desagüe como se muestra en la figura 1-1.

El tubo resonante está sumergido en el área de oleaje. Al resonar pasa agua a la cámara de compresión de donde desciende por gravedad a través del tubo de desagüe al cuerpo de agua con poco o sin oleaje. En experimentos realizados con un modelo de **SIBEO** en un tanque de olas, se mostró que al poner un difusor en la entrada del tubo resonante (*toma de*

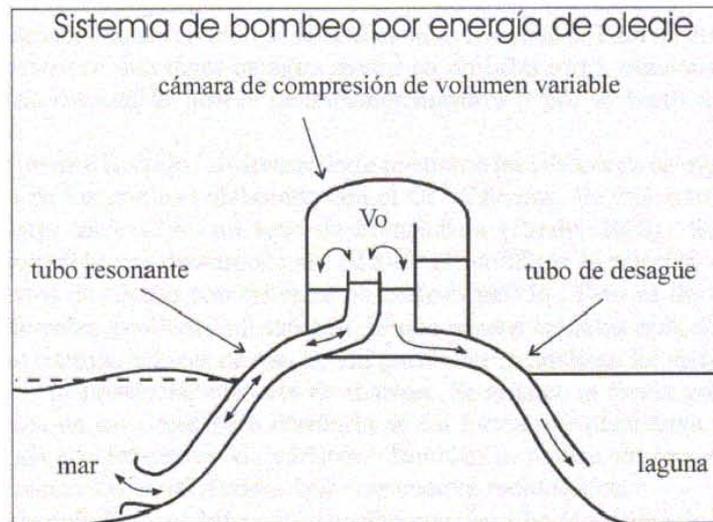


Figura 1-1: Esquema general del **SIBEO**

agua) se incrementó el flujo volumétrico hasta en un 15%. Además, en las ecuaciones que modelan al **SIBEO** [1] aparecen términos de pérdidas por formación de vórtices que tienen un efecto importante en el flujo volumétrico. Es aparente que la geometría de la toma en el tubo resonante desempeña un papel relevante en el flujo volumétrico.

Para determinar la geometría más adecuada se diseñó un sistema de laboratorio que simula el paso del oleaje a través de la boca del tubo resonante. El sistema consiste en un tubo sumergido parcialmente en un tanque. Dentro del tubo hay un pistón que realiza un movimiento periódico produciendo un flujo oscilante. Se diseñaron difusores con geometrías diferentes que pueden ser añadidos a la parte sumergida del tubo.

Las curvas de los difusores, fueron diseñadas de manera que hubiera una razón constante entre las fuerzas centrípeta e inercial a lo largo de su perfil. Con este diseño se logró evitar la formación de vórtices dentro del tubo. En cambio, en el borde del difusor se forma un vórtice cuando el agua sale y otro cuando el agua entra. Se buscan las condiciones experimentales bajo las que no se generan vórtices [2].

El flujo producido alrededor del difusor se visualizó principalmente de dos maneras: inyectando pintura localmente en la salida del difusor o introduciendo pintura metálica en el agua para observar su movimiento. Con estas visualizaciones se pudo ver que el flujo producido alrededor del difusor depende no solo de su geometría, sino también de la amplitud y frecuencia de la oscilación.

El objetivo de este trabajo es encontrar un método para cuantificar el efecto de las geometría y de la oscilación, es decir, determinar el campo de velocidades alrededor del difusor y de ser posible dentro del tubo.

En el capítulo 2 se describe el algoritmo de *Velocimetría por imágenes de partículas* o **PIV** por sus siglas en inglés; (*Particle Image Velocimetry*). En el capítulo 3 se describe en detalle el dispositivo experimental. En el capítulo 4 se reportan y se analizan los resultados. Finalmente en el capítulo 5 se presentan las conclusiones y perspectivas del trabajo.

Capítulo 2

Velocimetría Por Imágenes de Partículas

Desde hace algunos años se están desarrollando métodos que permiten medir el campo de velocidades de un fluido utilizando el procesamiento digital de imágenes. De estos métodos destaca el de velocimetría por imágenes de partículas **PIV**. Este es un método efectivo instantáneo y no intrusivo, que ha sido perfeccionado por diversos investigadores. El método consiste en iluminar un plano de luz e introducir partículas reflejantes con una densidad similar a la del fluido. Esta luz incide sobre el visor de la cámara formando una imagen. Se toman fotografías separadas entre sí por un tiempo pequeño conocido y se analizan las imágenes. Los métodos **PIV** se basan en registrar imágenes consecutivas del flujo y determinar las distancias recorridas por las partículas entre las dos imágenes. Por ser de la misma densidad que el fluido se supone que el desplazamiento de una partícula sólida corresponde al de una partícula fluida. Al conocer el desplazamiento y el tiempo se calcula la velocidad instantánea de cada partícula.

Con la revolución tecnológica desatada por la aparición de las computadoras y el desarrollo en estado sólido que ha sido capaz de crear los denominados **CCD**,¹ el método **PIV** se ha visto enormemente beneficiado. Existen principalmente dos tipos de **PIV** que dependen de la forma en que se analizan las imágenes. La autocorrelación [19], este método consiste en

¹Un CCD (Coupled Charge Device) es un dispositivo formado por millones de pequeñas unidades fotosensibles que producen un flujo de electrones al ser expuestas a la luz

exponer la región que se desea analizar dos veces sobre la misma superficie, ya sea un *CCD* o una película fotosensible. Se calculan los desplazamientos, pero el signo de los vectores queda indeterminado debido a la pérdida de información temporal por haber grabado sobre el mismo dispositivo. El otro método es el llamado de correlación cruzada . Este es el más utilizado ya que resuelve el problema de la ambigüedad en la dirección de los vectores de velocidad. En este trabajo se digitalizaron videos de manera que pudieron obtenerse campos de velocidades en función del tiempo.

Otras técnicas de medición de Velocidades vs PIV

La técnica de **PIV** es muy útil para calcular campos de velocidades con unos segundos de exposición de la cámara. Sin embargo, la mayoría de las técnicas tiene limitaciones implícitas a su metodología, por ejemplo, cuando tenemos un fluido opaco, o cuando se desea una definición espacial mayor a la resolución que ofrece una cámara convencional. Existen otras técnicas que hacen mediciones puntuales con gran resolución. La más conocida es la llamada Anemometría láser por efecto Doppler (LDA) ². Esta técnica consiste en hacer pasar partículas a través del volumen de intersección de dos haces que forman en el fluido un patrón de interferencia. La luz dispersada por la partícula al pasar por el patrón de interferencia se capta en un foto detector en forma de pulsaciones en la intensidad de luz. La frecuencia de la luz dispersada tiene un corrimiento en frecuencia debido al movimiento de las partículas. Este corrimiento debido al efecto *Doppler*, está relacionado con la magnitud de la velocidad de la partícula. Para evitar la ambigüedad en el sentido de la velocidad es necesario cruzar dos haces de diferente frecuencia. Aunque esta técnica es muy exacta en términos de resolución, solamente se puede medir un punto a la vez. Se necesitan varias mediciones para determinar el campo de velocidades en toda una región. Esta técnica se aplica fácilmente solo a flujos estacionarios ó periodicos. En la siguiente expresión se muestra la relación matemática que describe este fenómeno.

$$f_D = f_0 \left(1 - \frac{e_{B1} \cdot V}{c} \right).$$

²por sus siglas en ingles **Laser Doppler Anemometry**

Donde f_0 es la frecuencia del rayo laser que se utiliza, c es la velocidad de la luz, e_{B1} es un vector unitario en la dirección del rayo, V es la velocidad de la partícula, y f_D es la frecuencia de la luz dispersada por el flujo en movimiento. [3].

Otra técnica para medir la velocidad en un flujo es la **Velocimetría por Hilo Caliente CTA**³, que está basada en medir cambios en el voltaje debido a cambios en la temperatura de un alambre suspendido sobre el flujo. El cambio en la temperatura está ligado al cambio en la velocidad. La medición se realiza punto a punto. La distancia entre puntos es la resolución espacial de la medición.

2.1. ¿Qué es PIV?

El método de velocimetría por imágenes de partículas consta de dos partes. La primera consiste en preparar el prototipo para poder observar el fluido y obtener imágenes. Es importante seleccionar correctamente las partículas en suspensión en cuanto a su reflectividad y densidad para obtener imágenes nítidas. La segunda parte consiste en el procesamiento de las imágenes obtenidas.

Cada imagen obtenida tiene una representación matemática en forma de matriz, donde cada uno de sus elementos representa la intensidad de brillo en un pixel.

En nuestro caso, el método utilizado para realizar las mediciones **PIV** es el de correlación cruzada [5], ya que como mencionamos anteriormente, resuelve el problema de la ambigüedad en la dirección de los vectores de velocidad. El método de correlación cruzada se basa en la hipótesis de que dentro de una región dada, un pequeño desplazamiento del fluido se realiza sin deformarse ni rotar [4]. Por lo tanto, si realizamos la correlación entre ambas imágenes el valor del desplazamiento para el cual hay un máximo en la función de correlación indicará el

³por sus siglas en ingles, *Constant Temperature Anemometry*

desplazamiento del fluido respecto a la posición inicial. Con este fin se dividen las imágenes en pequeñas ventanas de observación. Para las ventanas ubicadas en la misma posición en ambas fotos se calcula la correlación entre ambas imágenes.

El método operacional para calcular la correlación cruzada consiste en obtener primero la transformada de Fourier de cada una de las funciones asociadas a través de la escala de grises a las imágenes, obtener la conjugada de la segunda, y multiplicar punto a punto ambas matrices[24].

Para calcular la probabilidad de encontrar una cierta partícula en alguna de las ventanas de interrogación o celdas, es necesario conocer el tamaño de la ventana, que se sugiere tenga el tamaño;

$$d = v_{max} \Delta t. \quad (2.1)$$

donde v_{max} es la velocidad máxima que puede alcanzar la partícula en el fluido y Δt es la diferencia temporal entre las imágenes. El número de partículas N encerradas en cada celda, está dado por;

$$N = C \Delta Z_0 d_i^2. \quad (2.2)$$

El parametro ΔZ_0 es el espesor de la hoja de luz, y C es una constante de proporcionalidad, que esta ligada a la concentracion de particulas brillantes en el agua.

Como se mencionó anteriormente, primero se calcula la transformada espacial de Fourier F de cada una de las imágenes f . Puesto que están digitalizadas las imágenes pueden ser analizadas como dominios discretos [8]. Las figuras 2-1 a) y b) corresponden a una celda de interrogación sobre el fluido en movimiento. La parte correspondiente a 2-1 a), es la imagen, la parte b) de la misma figura es la gráfica de intensidad luminosa y posición. En este tipo de dominios para calcular la Transformada de Fourier, se utiliza la versión para dominios discretos, de esta forma, definimos a (x, y) , en el dominio de la función y a (u, v) en el contradominio de modo que $x, u = 1, 2, \dots, M$ $y, v = 1, 2, \dots, M$., donde M es el número

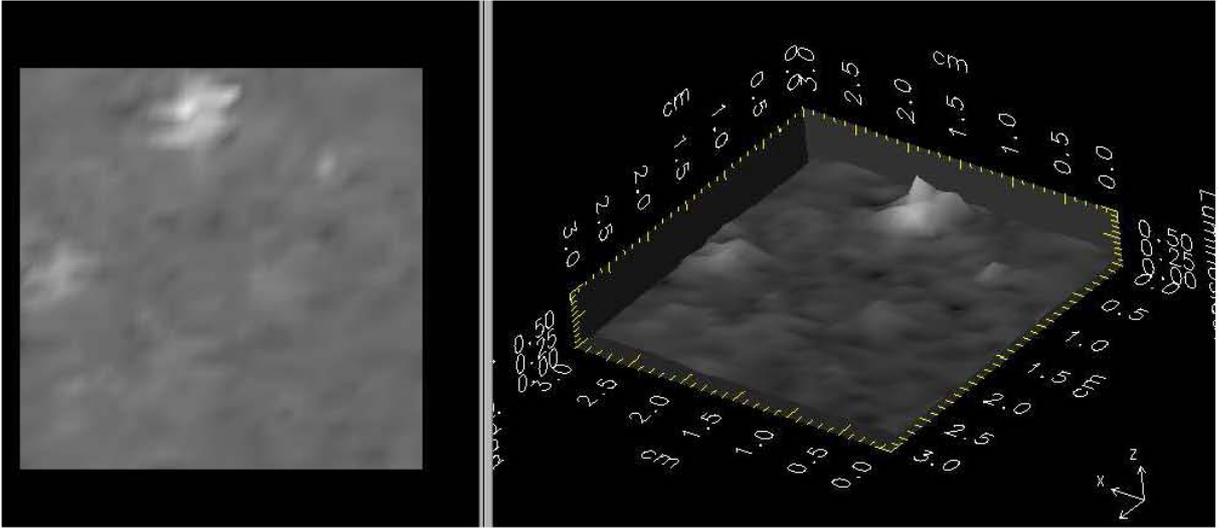


Figura 2-1: a) Imagen de la celda b) Gráfica de luminosidad contra posición de cada pixel de pixeles en cada lado de la celda, suponiendo que es cuadrada.

La figura 2-2 muestra la transformada de Fourier de la imagen 2-1. El valor que toma la función F en cada punto está representado por la intensidad de color, siendo el color negro el número cero, y el color blanco el número 255. Todos los demás colores en una escala de grises son valores intermedios.

$$F(u, v) = \frac{1}{M} \sum_{x=0}^{M-1} \sum_{y=0}^{M-1} f(x, y) e^{-2\pi i \left(\frac{ux+vy}{M} \right)}. \quad (2.3)$$

Si graficamos $F(u, v)$, para cada una de las imágenes, obtenemos las figuras 2-3 a) y b), donde se observa que los valores máximos aparecen en posición distinta que en $f(x, y)$. Sin embargo, se ven envueltos en una serie de picos que originalmente no estaban, y que son consecuencia de la forma en que opera la transformación. De cualquier manera, la información relevante se encuentra en los máximos principales de la función, y el resto de los picos nos indica la distribución de puntos luminosos sobre la celda.

Si graficamos $F(u, v)$, para dos celdas sucesivas temporalmente, con una diferencia $\Delta t = 0,033$ segundos, obtenemos las figuras 2-3 a) y b), donde se observa que los valores

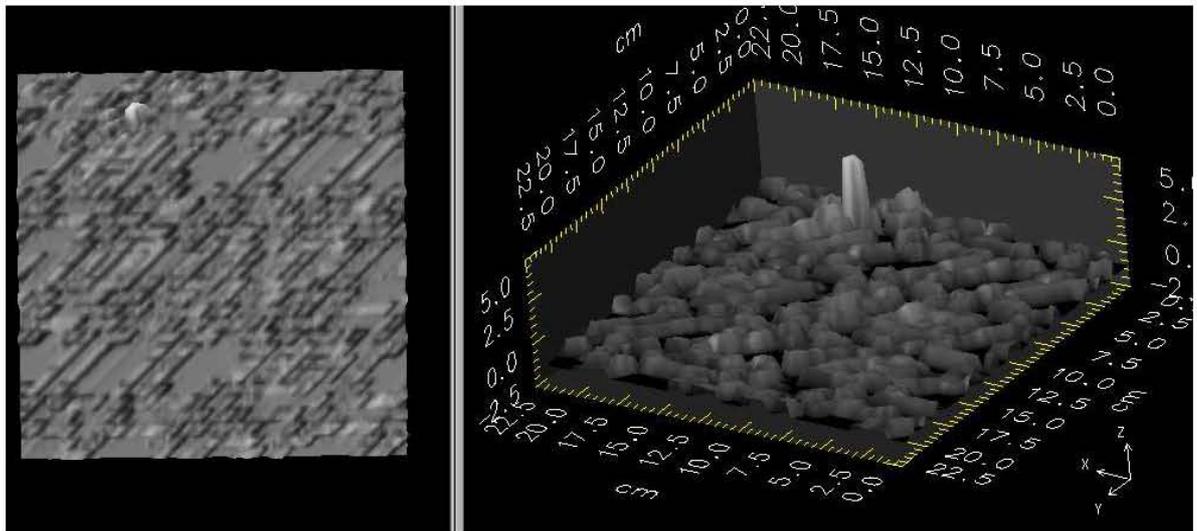


Figura 2-2: a) Imagen Transformada b) Gráfica de la transformada de Fourier

máximos aparecen en posición distinta que en $f(x,y)$. Sin embargo la información se encuentra en todos los puntos de ambas funciones. La forma de relacionar una celda con otra temporalmente es a través de una función de correlación como veremos más adelante.

Lo que sigue es calcular la función de correlación [6]. La función de correlación de 2 funciones continuas $h(x)$, y $g(x)$ se representa como $h(x) \circ g(x)$ y se define por la relación

$$h(x) \circ g(x) = \int_{\text{dominio}} h^*(x)g(x + \alpha)d\alpha. \quad (2.4)$$

Donde $*$ es el complejo conjugado. La función de correlación representa una medida de la similitud o interdependencia entre las funciones $h(x)$ y $g(x)$ en función del parámetro α . En el caso en que $h(x)$ y $g(x)$ sean iguales, entonces a la función de correlación se le denomina función de autocorrelación.

La correlación entre dos funciones discretas se define de acuerdo a la ecuación 2.5.

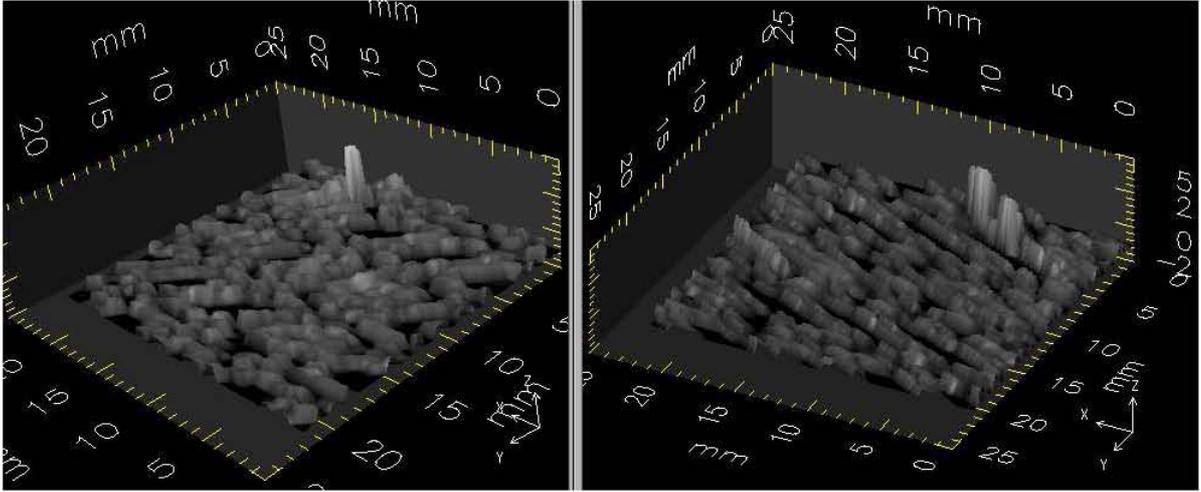


Figura 2-3: Transformada de Fourier de dos celdas de una misma region que distan temporalmente $\Delta t = 0,033$ segundos

Para $x = 0, 1, 2, 3..M - 1$:

$$h(x) \circ g(x) = \sum_{m=0}^{M-1} h(M)^* g(x + m), \quad (2.5)$$

En el caso bidimensional de funciones continuas las expresiones se muestran en la ecuación 2.6.

$$h(x, y) \circ g(x, y) = \int_{\text{dominio}} h^*(x, y) g(x + \alpha, y + \beta) d\alpha d\beta, \quad (2.6)$$

$$h(x, y) \circ g(x, y) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} h(M, N)^* g(x + m, y + n). \quad (2.7)$$

Para $u = 0, 1, 2, 3..M - 1$ y $v = 0, 1, 2, 3..N - 1$. En el espacio u, v $F_1(u, v)$, y $F_2(u, v)$ dos celdas sucesivas cualesquiera. La correlación es espacial, es decir el tiempo aunque esta involucrado no aparece explícitamente en la expresión. $F_{i=1,2}(u, v)$ corresponde con la intensidad de la imagen ya transformada con Fourier. Se tiene que:

$$F_1(u, v) \circ F_2(u, v) = \sum_{m=0}^{M-1} \sum_{n=0}^{M-1} F_1(M, M)^* F_2(u + m, v + m) \quad (2.8)$$

En la figura 2-4 se muestra la gráfica que corresponde a la correlación entre las dos matrices graficadas en 2-3. El valor más alto en la figura 2-4 indica el máximo de la

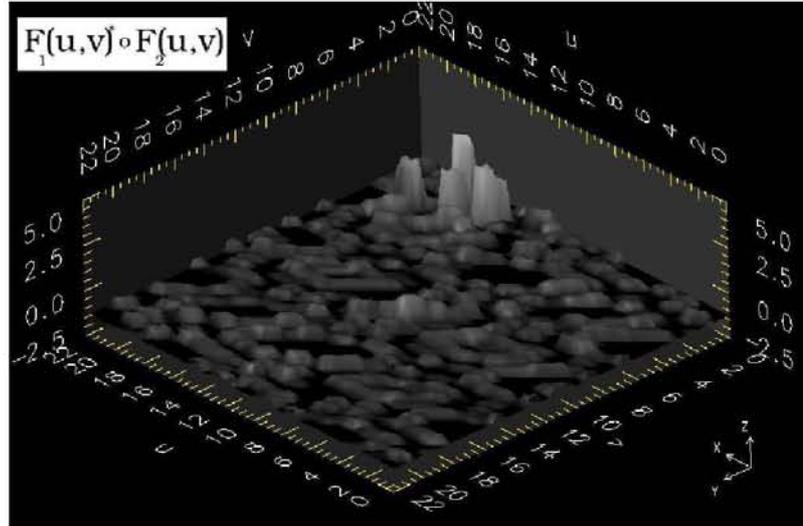


Figura 2-4: Función de correlación. de las celdas correspondientes a la figura 2-3

correlación. También aparecen dos valores secundarios o picos en la misma gráfica, que están relacionados con cada uno de los valores máximos originales antes de la correlación. La distancia que hay entre la posición del máximo R y la posición del primer máximo secundario, con respecto a la esquina superior izquierda se denomina R_{-D} . Análogamente la distancia con el segundo máximo secundario se llama R_{+D} . El desplazamiento de la partícula entre las dos celdas se calcula con las ecuaciones 2.9 y 2.10. El signo en cada una indica la dirección del desplazamiento.

$$\mu_{+D} = R_{+D} - R \quad (2.9)$$

$$\mu_{-D} = R_{-D} - R \quad (2.10)$$

Donde μ_{-D} es el desplazamiento en sentido negativo, y μ_{+D} en sentido positivo. Ambos cumplen con la siguiente propiedad [10].

$$\mu_{-D} = -\mu_{+D} \quad (2.11)$$

Para calcular la velocidad es necesario conocer el intervalo de tiempo Δt en el que se llevó a cabo dicho desplazamiento. Este es el intervalo temporal entre las imágenes, por lo

tanto, el vector velocidad queda definido como:

$$\vec{v}_{+D} = \frac{\mu+D}{\Delta t} \quad (2.12)$$

$$\vec{v}_{-D} = \frac{\mu-D}{\Delta t} \quad (2.13)$$

Ambos vectores tienen la misma magnitud pero sentido opuesto. Lo único que resta saber es la distancia física a la que corresponde cada pixel. Esto se mide haciendo marcas o etiquetas en las fotografías originales. Estas marcas son en general objetos del mismo tamaño, como por ejemplo las líneas de un flexómetro o regla de medir. De este modo, si se conoce la distancia real o física entre las marcas, se puede conocer la distancia virtual, es decir, la distancia que hay en términos de píxeles entre ambas etiquetas. El cociente de ambas magnitudes indica la distancia real a la que corresponde cada pixel.

$$dp = \frac{\Delta d}{\Delta p} \quad (2.14)$$

En la ecuación 2.14 dp es la distancia por pixel, Δd es la distancia física que hay entre las etiquetas, y Δp la distancia virtual entre las etiquetas. En resumen, se puede decir que este proceso de calibración es el que permite conocer la magnitud real de cada vector en el campo de velocidades. Para los experimentos de este caso, cada pixel tiene una longitud de 0,02 cm, en la sección 2.2 se detallará como se obtuvo este valor. Por lo tanto, si la imagen total tiene un tamaño de 720×480 píxeles, y está dividida en 20×20 celdas, entonces cada celda tiene un tamaño de 18×12 píxeles, que es equivalente a $0,36 \times 0,24$ cm.

2.2. Aplicación del PIV

El equipo de **PIV** consta de una cámara de video digital *Panasonic PV-DV401D* en formato mini-DV que tiene una velocidad de captura de 29.97 cuadros por segundo, con una resolución de 720 por 480 píxeles. Se empleó como trazador pintura metálica soluble en agua. Para producir una hoja de luz se utilizó un proyector de 300 *watts*, al cual se le inserta una laminilla con una rejilla de 0,8 *mm* de ancho. La figura 2-5 muestra al difusor

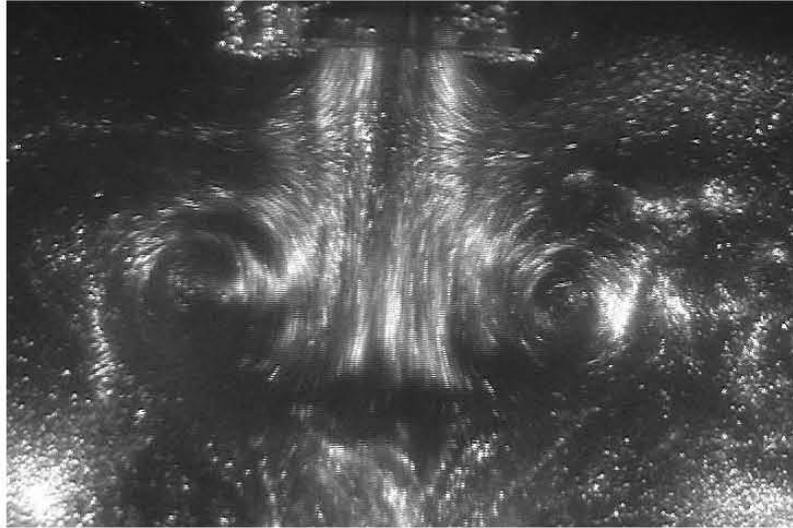


Figura 2-5: La imagen corresponde con la salida del difusor. Los puntos brillantes son partículas de pintura metálica suspendida en el agua. La iluminación es producida con una hoja de luz.

inmerso en el agua con partículas de pintura metálica en suspensión.

La hoja de luz ilumina la toma del difusor haciendo brillar las partículas de aluminio suspendidas en el agua y permitiendo seguir su trayectoria mientras no salgan de la región iluminada. La hoja tiene un espesor aproximado de 1.5 cm y como es de esperar, muchas partículas entran y salen a cada instante. Se podría pensar que el intercambio de partículas entre la zona iluminada y la no iluminada vuelve confuso o inestable el mecanismo para la medición de campos de velocidades. Sin embargo, esto no es problema, ya que como se explicó en la sección 2.1, se calcula la transformada de Fourier para obtener un centroide de luminosidad, lo cual se podría entender también como un promedio. La cámara de video se coloca perpendicularmente a la hoja de luz, recibiendo la luz dispersada por las partículas inmersas en el fluido en esa dirección, como se muestra en la imagen 2-6

Las imágenes son grabadas por la cámara en formato digital, y se envían al disco duro de

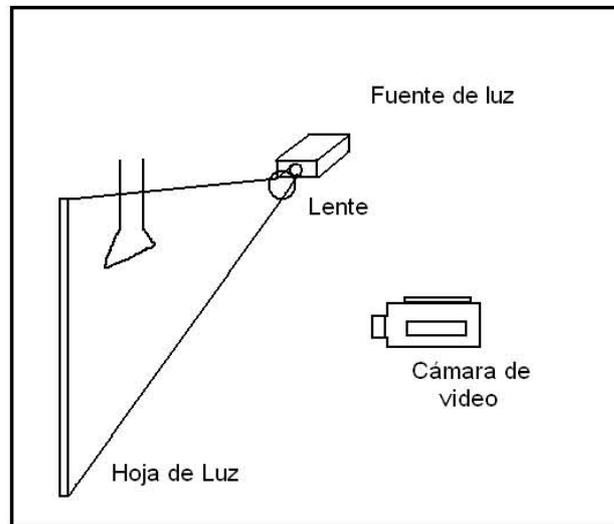


Figura 2-6: Esquema del dispositivo experimental para mediciones de **PIV**, La Cámara permanece perpendicular a la hoja de luz

una computadora personal via IEEE-1394⁴. El análisis **PIV** no ocurre en tiempo real debido a su alto costo computacional. La información enviada por la cámara a la computadora es convertida en secuencias de imágenes tiff, las cuales deben ser convertidas a escala de grises, ya que es la forma en que operan los algoritmos **PIV**. Esto tiene que ver con el vector de color *RGB*. Las imágenes a color son imágenes vectoriales, donde cada pixel es la combinación de rojo (*Red*), verde (*Green*), azul (*Blue*) es decir, un vector de tres entradas (R, G, B). R, G, B , son variables que pueden tomar valores entre 0 y 255 lo que es igual a 2^8 valores distintos. Si cada color tiene 2^8 posibilidades, entonces la combinación de los tres ofrece 2^{24} posibilidades de color. A esto también se le conoce como 24 bits de color.

Al convertir las imágenes *RGB* a escala de grises, lo que se hace es tomar la norma del vector de color para cada pixel. Este valor es asignado a las entradas de otra matriz de igual tamaño. Así se pasa de una matriz de entradas vectoriales a una de entradas escalares, lo que visualmente corresponde a una transformación de una imagen de 24 bits de color a una

⁴IEEE 1394 es un estándar definido como un bus serial de alta velocidad.

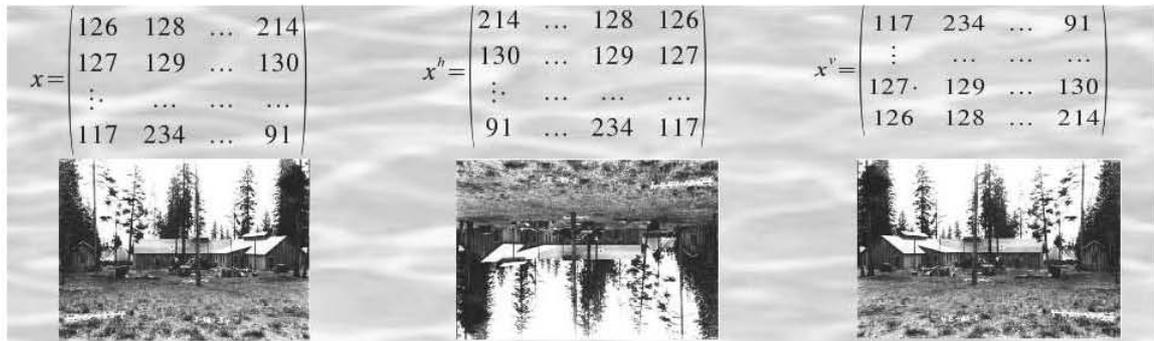


Figura 2-7: a)Normal b)Rotación vertical c)Rotación horizontal, Un ejemplo de imagen bajo distintas transformaciones matriciales, cada pixel de la imagen corresponde con valor entre 0 y 255 en la matriz

en escala de grises. Hoy en día existen imágenes de 32 y 48 bits. Sin embargo, el principio es exactamente el mismo al *RGB*, aunque ahora el vector correspondiente tiene un número mayor de entradas.

Una vez que las imágenes o fotogramas están en memoria y convertidas a escala de grises, es posible hacer la correlación. Para ello se utilizaron un conjunto de algoritmos cuyo objetivo es automatizar, en la medida de lo posible, las tareas necesarias para el cálculo de campos de velocidades. Primero, se toma un par de imágenes sucesivas en el tiempo que de aquí en adelante serán referidas como matrices. Cada una tiene un tamaño de 720 columnas y 480 renglones. El valor numérico que toma cada entrada en la matriz es la intensidad de color, como se muestra en el ejemplo de la figura 2-7

De este modo al tener un conjunto de matrices, cada una de las cuales representa un estado del experimento, se puede utilizar algebra matricial. El algoritmo de correlación se puede entender como una sucesión de operaciones entre ambas matrices. Se supone que el conjunto de matrices que describe al experimento ya fue almacenado, y se puede acceder a él sin problemas.

$$A(i) = \begin{pmatrix} \overbrace{c_{1,1} \ c_{1,2} \ \dots \ c_{1,y}}^{a_{1,1}^i} & \dots & c_{1,h} \\ c_{2,1} & c_{2,2} & \dots & \dots & \dots \\ \vdots & \dots & \dots & \dots & \dots \\ c_{o,1} & c_{o,2} & \dots & \dots & \dots \\ \vdots & \dots & \dots & \dots & \dots \\ c_{k,1} & c_{k,2} & \dots & c_{k,y} & \dots & c_{k,h} \end{pmatrix}$$

$$A(i+1) = \begin{pmatrix} \overbrace{d_{1,1} \ d_{1,2} \ \dots \ d_{1,y}}^{a_{1,1}^{i+1}} & \dots & d_{1,h} \\ d_{2,1} & d_{2,2} & \dots & \dots & \dots \\ \vdots & \dots & \dots & \dots & \dots \\ d_{o,1} & d_{o,2} & \dots & \dots & \dots \\ \vdots & \dots & \dots & \dots & \dots \\ d_{k,1} & d_{k,2} & \dots & d_{k,y} & \dots & d_{k,h} \end{pmatrix}$$

Figura 2-8: a) Representación matricial de una celda en una imagen b) La misma celda un tiempo $\Delta t = 0,033$ segundos despues

Sea $A(1)$ la primera fotografía a un tiempo t_0 . A un tiempo $t_0 + \Delta t$ se toma la siguiente imagen $A(2)$, y así hasta $t_0 + i \cdot \Delta t$ para la matriz $A(i)$. De esta manera se forma el conjunto de matrices o fotografías digitales que describe al experimento. La correlación entre las matrices $A(1)$ y $A(2)$ es representada con el simbolo \circ , es decir $A(1) \circ A(2)$, pudiendolo hacer para el caso más general $A(i) \circ A(i+1)$. El primer paso es subdividir la matriz $A(i)$ en arreglos más pequeños $f(n, m)$ donde n y m indican la posición de la submatriz f dentro de $A(i)$. Estas submatrices son las celdas de interrogación. El tamaño de la celda está relacionado con la resolución que se obtendrá del campo de velocidades. Se calcula la transformada de Fourier (ecuación 2.4) para la primera submatriz de $A(i)$, tal que $T_F[f_{1,1}] = F_{1,1}$. es decir $F_{1,1}$ es la matriz transformada. Se hace un procedimiento análogo para $A(i+1)$, es decir, se divide en arreglos del mismo tamaño que $A(i)$, y se obtiene $f_{1,1}$ (ver figura 2-8), al aplicar la transformada de Fourier, tenemos $F_{1,1}$.

Correlacionar imágenes, permite identificar patrones de una en otra. Estos patrones están dados por la distribución de las partículas en el fluido. Dividir las imágenes en celdas más pequeñas permite tener una mejor resolución de la distribución, y por lo tanto de buscar estructuras en la imagen. El tamaño de una estructura en el interior de la imagen debe ser por lo menos dos veces más grande que una celda para que pueda ser observada.

Al calcular la transformada de Fourier, lo que se hace es obtener un promedio pesado por la distribución espacial de la intensidad luminosa para cada celda. El costo de este método es la pérdida de las trayectorias de partículas individuales. Existen otros métodos para estos casos. Lo que el método busca son tendencias de un conjunto de puntos brillantes para cada celda de interrogación a lo largo del grupo de matrices A .

El siguiente paso es establecer una relación espacial entre las celdas, para lo cual se utiliza la función de correlación en su forma discreta.

$$F_t(u, v) \circ F_{t+\delta t}(u, v) = \sum_{m=0}^{M-1} \sum_{m=0}^{M-1} F_t(M, M)^* F_{t+\delta t}(u+m, v+m) \quad (2.15)$$

donde $F_t(u, v)$ representa la celda de interrogación al tiempo t y $F_{t+\delta t}(u, v)$ un δt después. El proceso es repetido para correlacionar cada celda de interrogación con su análoga en posición un tiempo Δt después. De este modo, se hace la correlación total de cada par de imágenes correlacionando todas las celdas que las componen. Sin embargo, este método no es equivalente a correlacionar cada par de imágenes directamente, es decir, con celdas de interrogación del tamaño de la imagen. La diferencia radica en la resolución del campo de velocidades. Así, si las celdas de interrogación $F_t(u, v)$ son demasiado grandes comparadas con la imagen A , entonces se obtiene un campo de baja resolución espacial en la medición de PIV. Por el contrario, si las celdas $F_t(u, v)$ son muy pequeñas respecto de A , entonces la resolución espacial aumenta. También el número de operaciones aumenta de forma cuadrática. Además el tamaño mínimo de las celdas está limitado por la densidad de partículas en el medio, y por la resolución en píxeles de la imagen A .

2.3. Determinación de campos de velocidades

Hasta este momento, se ha determinado el campo de desplazamientos, que no es propiamente el de velocidades aunque tiene la misma forma. Las unidades del campo de desplazamientos están en píxeles, si multiplicamos por el factor 0,02 cm/píxel, convertimos el campo de desplazamiento a centímetros. Aunque el campo obtenido ofrece información matemática

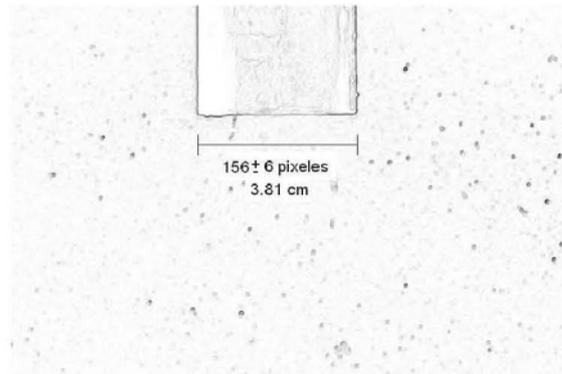


Figura 2-9: Comparacion de una medida virtual con una física.

en términos de forma y tendencia, no ofrece información física en el entendido de obtener los vectores de velocidad con sus unidades correspondientes. La relación entre el campo de desplazamientos y el de velocidades es una constante k con unidades de $\frac{distancia}{(pixel)tiempo}$. El problema de determinar k se reduce a saber a qué distancia física corresponde cada pixel. Si suponemos que la lente de la cámara no distorsiona la imagen ó que si lo hace es de forma constante ⁵, entonces, es posible fijar el valor de k sobre toda la imagen. De lo contrario sería una función de la posición $k(x, y)$. Para facilitar los cálculos se considera que es constante sobre toda la imagen, lo cual es una buena aproximación debido a que para las distintas comparaciones de la medida de referencias físicas y referencias virtuales coincide siempre con una constante de 0,02 cm/pixel

$$k = \frac{3,81cm}{156 \pm 6pixeles} = 0,02 \pm 0,003 \frac{cm}{pixel}$$

Para observar con mayor definición las fronteras del objeto físico, o referencia, se usó la transformada de **SOBEL**. [16]

El método que se utilizó; para medir la magnitud de la referencia en pixeles consiste en buscar los contornos de la imagen a través de la transformada de **SOBEL** (ver ecuación 2.16). Se toma una línea horizontal a lo largo de la imagen resultante, y que pase por

⁵La amplificación es un ejemplo

encima de los contornos de la referencia. Esta línea es un vector renglón R_i , donde i es la posición. Si se busca los dos primeros máximos o máximos principales de este vector, entonces se habrá localizado los contornos de dicha referencia. La distancia entre ambos contornos indica el tamaño en pixeles de la referencia. Se puede repetir este método para otros renglones paralelos a R_i , y se puede tomar un promedio entre los máximos.

$$\langle H_z, A \rangle_{m,n} \equiv \sum_{i=1}^3 \sum_{j=1}^3 H_z(i, j) A(i + m, j + m) \quad (2.16)$$

donde H_z es llamado el núcleo (*kernel*) de la transformación, $z = x, y$ H_x , y H_y , representan un barrido horizontal y otro vertical respectivamente. La transformación de **SOBEL** consiste en aplicar primero la transformación horizontal $\langle H_x, A \rangle$, y luego la vertical a la matriz resultante $\langle H_y, \langle H_x, A \rangle \rangle$.

$$\mathbf{H}_x = \begin{pmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{pmatrix}$$

$$\mathbf{H}_y = \begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & -1 \end{pmatrix}$$

Los núcleos o *kernel* de la transformada de **SOBEL** se utilizan para determinar los contornos de una imagen. Comparan valores punto a punto sobre cada pixel con los 8 vecinos inmediatos. Esto da un valor que es almacenado en la entrada de una nueva matriz. Si una imagen tiene un color homogéneo o un gradiente de color que cambia suavemente, la matriz de contornos estará llena de ceros y será una imagen negra completamente. Es decir, no se detectan contornos. En cambio si se aplica la transformada a una imagen que tenga cambios muy rápidos de color, como por ejemplo; si la imagen esta compuesta de dos colores, en la frontera entre ellos la matriz de contornos tomará valores distintos de cero. La matriz de contorno puede interpretarse como una imagen. Los valores diferentes de cero corresponden a puntos del contorno.

2.4. PIV con SCILAB

El lenguaje de trabajo se escoge sobre todo por las funciones que lo enriquecen. En este caso en particular, el lenguaje utilizado es **SCILAB**, ya que cuenta con un gran número de funciones matemáticas que ahorran trabajo de programación. Algunas de estas funciones están ya muy especializadas como es el caso de las funciones para **PIV** hechas por Enrico Segre <http://www.weizmann.ac.il/home/fesegre>. Tal vez más adelante, como proyecto futuro se pueda reescribir el código del programa en *C++*. **SCILAB** es un proyecto desarrollado desde hace ya varios años por el instituto francés **INRIA** *Institut National de Recherche en Informatique et en Automatique*. Es un lenguaje de cálculo numérico, parecido a **OCTAVE** ó **MATLAB**. La diferencia con este último radica en la licencia de propiedad, ya que **SCILAB** forma parte del proyecto **GNU** ⁶. Es decir, el código fuente se puede obtener de forma gratuita en <http://www.inria.org>, además de versiones compiladas para distintos sistemas operativos. Para habilitar el manejo de imágenes bajo **SCILAB**, es necesario instalar los paquetes **SIP**, por sus siglas en inglés (**Scilab Image Process**), el cual se puede obtener en <http://www.imagemagic.com>, y está basado en las librerías **Gandalf** para el manejo de imágenes e incluso de video. El editor de texto que se utilizó para escribir los códigos fue **kwrite**, debido a su fácil manejo, su interface visual y su cómodo manejo de abreviaturas rápidas.

A continuación se hará una breve descripción del código. La primera parte consiste en solicitar, limpiar memoria, y activar las funciones especiales denominadas **SIP** que serán requeridas más tarde, además de verificar la existencia de funciones hechas especialmente para este código.

```
clear;
exec("/home/acustica/sip/loader.sce");
exec("/home/acustica/PIVtool/macros/vorticidad.sci");
if ~exists('medianvalid') then
getf('/home/acustica/PIVtool/macros/medianvalid.sci'); end
```

⁶Avalado por la **Free software Foundation**

```

if ~exists('dpiv') then getf('/home/acustica/PIVtool/macros/dpiv.sci'); end
if ~exists('interpola') then getf('/home/acustica/PIVtool/macros/vorticidad.sci'); end
stacksize(20000000)

```

El programa **loader.sce** que se encuentra en la ruta indicada, se encarga de habilitar las librerías de **SIP** para el manejo de imágenes. En el siguiente bloque se definen parámetros tales como el número de celdas de interrogación *catx*, y *caty* y la resolución de las imágenes *resx resy*.

```

catx=20; caty=20;
mx=catx*5; my=caty*5;
resx=720; resy=480;

```

La siguiente parte del programa es lo que se podría llamar el cuerpo del mismo, ya que está inmerso en un ciclo. Es decir, se repite tantas veces como lo indica la variable número. Primero deben ser cargadas en memoria dos imágenes sucesivas, que están indicadas por las variables *j*, y *j + 1*.

```

numero=51;
for j=1:numero
  archiv='' + string(j) + '.tif';
  archiv2='' + string(j+1) + '.tif';
  ima1=imread('/home/acustica/pinturap/' + archiv);
  ima2=imread('/home/acustica/pinturap/' + archiv2);
  i1=zeros(resx,resy);
  i2=zeros(resx,resy);
  i1=ima1';
  i2=ima2';
  iws=[150,100,64,45,30];
  presh='\%t';

```

Se define la variable *mtr* como el mínimo o referencia de intensidad luminosa. Además se definen los vectores *x*, *y* que servirán para determinar la escala del campo de desplazamientos.

```

if ~exists('mtr') then mtr=2*min(i1); end
[nix,niy]=size(i1);
nsx=mx * iws($)/iws(1); nsy=my * iws($)/iws(1);
x=linspace(nix/(2*nsx),nix-nix/(2*nsx),nsx);

```

```

y=linspace(niy/(2*nsy),niy-niy/(2*nsy),nsy);
svx1=-0*ones(nsx,nsy); svy1=-0*ones(nsx,nsy);
pek=1;
nsx=int(mx * iws($)/iws(pek)); nsy=int(my * iws($)/iws(pek));
x=linspace(nix/(2*nsx),nix-nix/(2*nsx),nsx);
y=linspace(niy/(2*nsy),niy-niy/(2*nsy),nsy);
masc=bool2s(double(i1(x,y))>mtr);
masc=masc(1:size(svx1,1),1:size(svx1,2));

```

Se llama a la función *dpiv* que calcula los campos de velocidades en las direcciones horizontal *sx*, y vertical *sy*, y que se explica más adelante con detalle.

```

[sx,sy]=dpiv(i1,i2,nsx,nsy,iws(pek),iws(pek),svx1,svy1,masc);
xbasc();
champ(x,-y,sx,-sy,strf="041");

```

Se calculan los campos de vorticidad.

```

[vortex]=vorticidad(sx,sy);
imshow(vortex')

```

Se almacenan las imágenes en disco duro.

```

salida=string(j);
xs2gif (0,'/home/acustica/temporal/'+string(salida),0);
ima1=imread('/home/acustica/temporal/'+string(salida));
salida=string(j)+'.bmp';
imwrite(zet*257,'/home/acustica/resultados/'+string(salida));
imshow(resu');
printf('Porcentaje  '), (j/numero)*100;
end

```

El funcionamiento esquemático del programa se muestra en la figura 2-10 El ciclo se repite tomando las imágenes en pares y haciendo el cálculo para todas ellas.

La función *dpiv.sci* se muestra a continuación. *dpiv* contiene el procedimiento para la correlación de las celdas.



Figura 2-10: Esquema del funcionamiento del código de para el calculo de **PIV**.

```

function [sx,sy]=dpiv(i1,i2,nx,ny,iwx,iwy,presx,presy,mask)
[lhs,rhs]=argn(0);
[ix,iy]=size(i1)
if rhs<6 then
if rhs==5 then iwy=iwx; end
if rhs<5 then iwx=min(32,ix);iwy=min(32,iy); end
end
if rhs<3 then
if rhs==3 then ny=nx; end
if rhs==2 then nx=int(ix/32); ny=int(iy/32); end
end
if rhs<8 then
presx=zeros(nx,ny); presy=zeros(nx,ny);
end
if rhs<2 then disp(" I need two images!"); return; end
sx=zeros(nx,ny); sy=zeros(nx,ny);
if ~exists("mask","local") then
mask=ones(nx,ny)==1
else
if $size(mask,1)<nx | size(mask,2)<ny then
write(\%io(2), " mask is too small! "+string(nx)+"x"+string(ny)+..
" needed, "+strcat(string(size(mask)),'x')+"found")

```

```

return
end
end
cx=(0:(nx-1))*(ix-iwx)/(nx-1)
cy=(0:(ny-1))*(iy-iwy)/(ny-1)
I1=zeros(2*iwx,2*iwy); I2=I1;
for i=1:nx
for j=1:ny
if mask(i,j) then
I1=double(i1(cx(i)+(1:iwx),cy(j)+(1:iwy)))
isx=min(max(0,cx(i)+presx(i,j)),ix-iwx)
isy=min(max(0,cy(j)+presy(i,j)),iy-iwy)
I2=double(i2(isx+(1:iwx),isy+(1:iwy)))

```

En esta sección se realiza la correlación, tomando la parte real del producto de las transformadas de Fourier para $I1$, $I2$.

```
C=real(fft(fft(I2,-1).*conj(fft(I1,-1)),1))
```

Cabe la posibilidad de que el máximo de la función de correlación no esté contemplado en el cálculo, por lo que es necesario interpolar la función.

```

exec("/home/acustica/PIVtool/macros/subpixel.sci");
[m1,m2]=subpixel(fftshift(C))
sx(i,j)=m1-floor(iwx/2)-1; sy(i,j)=m2-floor(iwy/2)-1;
end
end
end

```

En la matrices sx , y sy , se guarda la información sobre el campo de desplazamientos.

```
sx=sx+presx; sy=sy+presy;
```

La función de correlación calculada es discreta. El máximo no necesariamente coincide con el verdadero máximo de la función continua. Un modo de aproximar este máximo es aumentando la resolución de la celda de interpolación. Para realizar esta tarea se utiliza la

función *subpixel*⁷, que esencialmente sirve para determinar, a través de un algoritmo de interpolación basado en mínimos cuadrados, el máximo de la función de correlación ó un valor más cercano. El código de esta función se muestra en el apéndice A.

2.5. Cálculo del campo de vorticidades a partir del campo de velocidades

Existen diversos parámetros en nuestro dispositivo experimental. Los que consideramos más importantes son la amplitud y la frecuencia de oscilación del flujo y la geometría del difusor. Las condiciones de frontera que impone el tanque son importantes pero para este estudio estaban determinadas de antemano. A partir de los campos de velocidad medidos calcularemos los campos de vorticidad.

Los campos de velocidad V que genera el programa son una proyección bidimensional de vectores que apuntan en las direcciones vertical V_y , y horizontal V_x . El campo total obtenido es la suma de cada uno de estos vectores.

La vorticidad se define como $\vec{\zeta} = 2\vec{\omega}$ [20] donde $\vec{\omega}$ es el rotacional del vector de velocidad \vec{V} , como se muestra en la ecuación 2.17. En este caso \vec{V} es un vector bidimensional, por lo tanto $\vec{\omega}$ se puede expresar como se muestra en la ecuacion 2.18.

$$\vec{\omega} = \nabla \times \vec{V}. \quad (2.17)$$

$$\vec{\omega} = \left(\frac{\delta V_y}{\delta x} - \frac{\delta V_x}{\delta y} \right) \vec{k}. \quad (2.18)$$

Donde \vec{k} es un vector unitario perpendicular al plano. Para calcular las derivadas parciales numéricamente utilizamos la definición de derivada numérica con dos puntos [21]

⁷esta función fue desarrollada por Alex Liberzon <http://alexl.wordpress.com> & Roi Gurka <http://www.eng.uwo.ca/people/rgurka>

como se muestra en la ecuación 2.19. En las ecuaciones 2.20 y 2.21 aparecen las derivadas cruzadas de V_y y V_x respectivamente.

$$y' = \frac{y(x_i + \Delta x) - y(x_i - \Delta x)}{2\Delta x}, \quad (2.19)$$

$$\frac{\delta V_{y_i}}{\delta x_i} = \frac{V_{y_{i+1}} - V_{y_{i-1}}}{2\Delta x}, \quad (2.20)$$

$$\frac{\delta V_{x_i}}{\delta y_i} = \frac{V_{x_{i+1}} - V_{x_{i-1}}}{2\Delta y}. \quad (2.21)$$

Por lo tanto $\vec{\omega}_{x_i, y_i}$ queda definida de acuerdo a la ecuación 2.22, y $\vec{\zeta}_{x_i, y_i}$ a la ecuación 2.23.

$$\vec{\omega}_{x_i, y_i} = \left(\frac{V_{y_{i+1}} - V_{y_{i-1}}}{2\Delta x} - \frac{V_{x_{i+1}} - V_{x_{i-1}}}{2\Delta y} \right) \vec{k}, \quad (2.22)$$

$$\vec{\zeta}_{x_i, y_i} = \left(\frac{V_{y_{i+1}} - V_{y_{i-1}}}{\Delta x} - \frac{V_{x_{i+1}} - V_{x_{i-1}}}{\Delta y} \right) \vec{k}. \quad (2.23)$$

Se probará el método con un problema. Se considera un flujo cortante en el que el campo de velocidades cambia linealmente, en la dirección \hat{y} , pero permanece constante en la dirección \hat{x} . Entonces \vec{V} es de la forma;

$$V_x = cy \quad (2.24)$$

Entonces.

$$\vec{\omega} = -c\hat{k}, \quad (2.25)$$

$$\vec{\zeta} = -2c\hat{k}. \quad (2.26)$$

Si en particular $c = 1$, entonces, la vorticidad $\vec{\zeta} = -2\hat{k} \text{ Hz}$. Para comprobar esto con el algoritmo de vorticidad descrito anteriormente, se construye el campo de velocidades de flujo cortante como se muestra en la figura 2-11.

El resultado de aplicar este algoritmo se muestra en la gráfica siguiente 2-12. El código del algoritmo para la vorticidad esta impreso en el apéndice A.

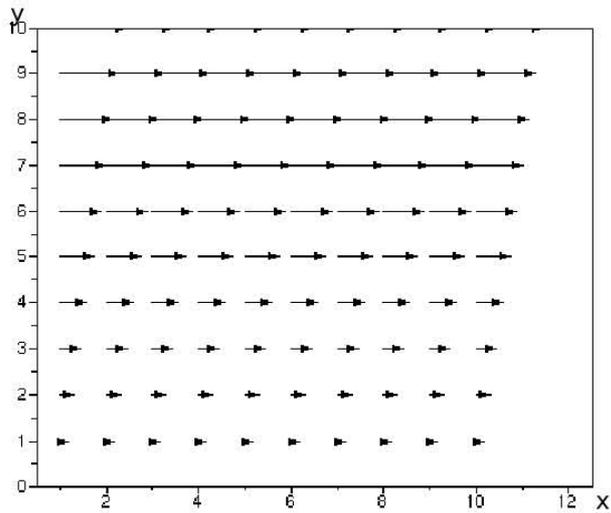


Figura 2-11: Campo de velocidades para flujo cortante con $c = 1$, Ambos ejes tienen una escala entre 1 y 10 cm , la magnitud de la velocidad esta entre 1 y 10 cm/s .

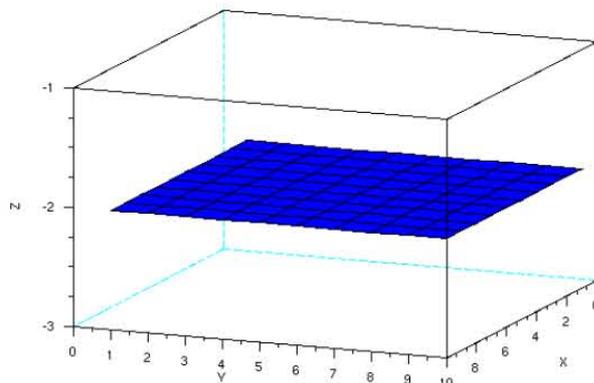


Figura 2-12: Gráfica de la magnitud del campo de vorticidad para un flujo cortante con $c = 1$, utilizando el algoritmo de vorticidad, la escala en Z esta graduada en Hz , los ejes X y Y en cm .

El resultado obtenido coincide satisfactoriamente con el que se habia calculado, pues la gráfica para la vorticidad de este ejemplo (ver gráfica 2-12), mantiene un valor constante igual a 2.0 Hz a lo largo de la trayectoria. A continuación se aplicará este método a los campos de velocidades obtenidos con las imágenes a través del **PIV**.

Capítulo 3

Interpolación y visualización de campos

Existen diversas técnicas de interpolación de superficies. Entre las más conocidas están la de mínimos cuadrados y las polinomiales. Dentro de estas últimas destacan los llamados *splines*, que se construyen con un cierto número de puntos sobre el dominio, y con otro cierto número de condiciones a la frontera, para garantizar suavidad en las derivadas.

La visualización de los campos también es muy importante, ya que muchas veces una gráfica permite observar rápidamente algún comportamiento, que sería imposible de visualizar directamente sobre las tablas de datos. En este capítulo se incluye la sección de visualización mediante **OpenDX**

3.1. Interpolación mediante *splines*

La interpolación mediante splines [17], es una variedad de la interpolación de una función mediante funciones que son polinomios por partes. Lo que distingue a las splines del resto de las interpolaciones con polinomios por partes, es que estas tienen una cantidad determinada (dependiendo del grado de la spline) de derivadas continuas. Dicho en forma poco rigurosa, se le impone a la función interpoladora una cierta suavidad en la unión de los polinomios que la componen. Por ejemplo, en el caso de la interpolación cúbica mediante splines, se imponen a cada polinomio dos puntos para la interpolación y los otros dos grados de libertad

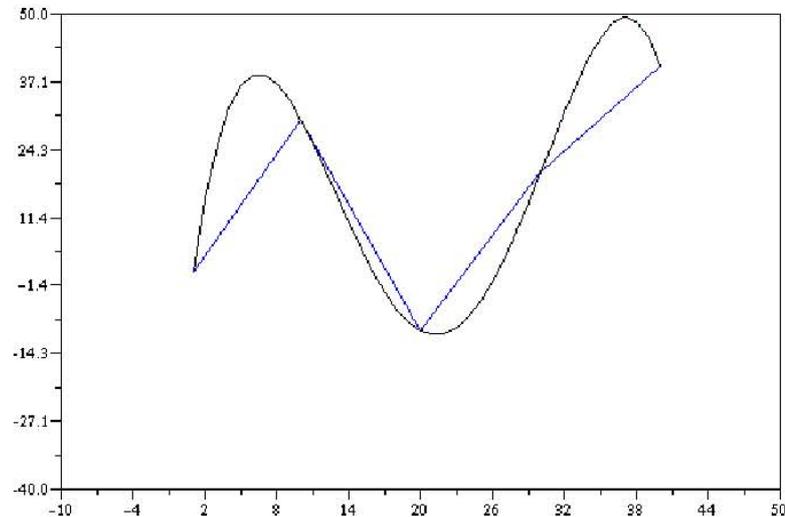


Figura 3-1: Colección de cinco puntos unidos mediante líneas rectas y mediante un polinomio de cuarto grado

se utilizan para lograr que la función sea continua en los extremos donde se empalma con otro polinomio.

Supongamos que se tiene una colección de cinco puntos, $[1,1]$, $[10,30]$, $[20,-10]$, $[30,20]$, $[40,40]$. Para ajustar una curva, hay varias opciones. Primero están los mínimos cuadrados. Estos garantizan gran suavidad a la información interpolada, sin embargo este método pierde los puntos originales, lo que lo pone en desventaja con respecto a otros métodos. El siguiente es ajustar un polinomio a los puntos originales como se muestra en la figura 3-1.

El inconveniente de este método radica en lo siguiente. Si se tiene un número grande de puntos, digamos n , hay que ajustar un polinomio de grado $n - 1$. Si decimos que $m = n - 1$, esto implica resolver determinantes para matrices de dimensión $m \times m$. Si n crece linealmente, el número de operaciones crece geométricamente. Lo que se desea es que el costo computacional no sea demasiado alto. Es aquí donde se utilizan los **splines**, que en general son polinomios de un grado mucho menor que el número de puntos. Una buena aproximación es utilizar polinomios de 4^{to} orden, los cuales se pueden construir con tres

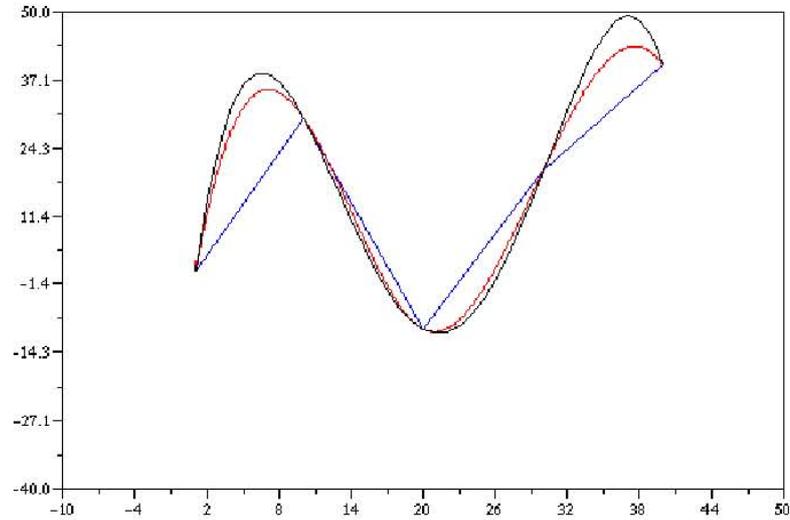


Figura 3-2: Colección de cinco puntos unidos mediante líneas rectas, mediante un polinomio de cuarto grado en negro y mediante **splines** en rojo.

puntos, y dos condiciones a la frontera. Esto en inglés se conoce como *not-a-knot*[18]. La condición a la frontera es de suavidad, es decir, si $P_i(n)$ es el polinomio i -ésimo sobre el punto n debe cumplir con $P'_{i-1}(n+) = P'_i(n-)$, donde $+$ significa aproximarse por la izquierda, y $-$ por la derecha. En la figura 3-4 se muestra la aproximación en rojo mediante este método.

En este trabajo se utilizó la técnica de *splines* para interpolar los campos de velocidades. Primero se hace la interpolación en la dirección horizontal, renglón a renglón hasta completar toda la matriz. De esta forma se obtiene una nueva matriz como se muestra en la ecuación 3.1 Si v es la matriz que guarda la información del campo de velocidades, entonces, a través de la transformación T_r se pasa a una nueva matriz V_r , donde r indica únicamente que la transformación se aplicó sobre los renglones. Lo relevante es que el número de renglones de V_r es mayor que el de v , de esta forma.

$$V_r = T_r(v) \quad (3.1)$$

Por ejemplo si v es una matriz de 4×4 la transformamos en una matriz V_r de 4×8 . Para facilitar el ejemplo solo tomaremos la parte entera de la interpolación.

$$\begin{pmatrix} 2 & 4 & 3 & 2 \\ 3 & 5 & 4 & 3 \\ 2 & 4 & 2 & 2 \\ 1 & 2 & 3 & 3 \end{pmatrix} \rightarrow \begin{pmatrix} 2 & 2 & 4 & 3 & 3 & 1 & 2 & 1 \\ 3 & 4 & 5 & 2 & 4 & 3 & 3 & 2 \\ 2 & 3 & 4 & 3 & 2 & 1 & 2 & 3 \\ 1 & 1 & 2 & 1 & 3 & 4 & 3 & 2 \end{pmatrix}.$$

Ahora se aplica una transformación análoga pero columna a columna sobre la matriz V_r

$$V = T_c(V_r) \quad (3.2)$$

Con la transformación sobre las columnas, se tiene;

$$\begin{pmatrix} 2 & 2 & 4 & 3 & 3 & 1 & 2 & 1 \\ 3 & 4 & 5 & 2 & 4 & 3 & 3 & 2 \\ 2 & 3 & 4 & 3 & 2 & 1 & 2 & 3 \\ 1 & 1 & 2 & 1 & 3 & 4 & 3 & 2 \end{pmatrix} \rightarrow \begin{pmatrix} 2 & 2 & 4 & 3 & 3 & 1 & 2 & 1 \\ 2 & 3 & 6 & 1 & 4 & 2 & 3 & 2 \\ 3 & 4 & 5 & 2 & 4 & 3 & 3 & 2 \\ 1 & 3 & 3 & 3 & 3 & 2 & 1 & 1 \\ 2 & 3 & 4 & 3 & 2 & 1 & 2 & 3 \\ 1 & 2 & 3 & 2 & 4 & 2 & 4 & 4 \\ 1 & 1 & 2 & 1 & 3 & 4 & 3 & 2 \\ 2 & 1 & 3 & 2 & 2 & 3 & 2 & 1 \end{pmatrix}.$$

Donde V es la matriz final, y T_c es la transformación que interpola las columnas de V_r , c indica que la transformación se aplica únicamente sobre las columnas. Se denota la composición de T_r y T_c , por el operador \cdot se tiene que;

$$V = T_c(T_r(v)) = T_c \cdot T_r(v),$$

Y si se define $T = T_c \cdot T_r$, entonces;

$$V = T(v). \quad (3.3)$$

Aplicando este mecanismo de interpolación al caso particular del flujo cortante, se debe tener en cuenta que la magnitud de cada vector representa el área de cada celda. Al

interpolador, se refina la maya que divide la imagen total. Ahora las celdas son mas pequeñas, y por lo tanto la magnitud de los vectores representa un área menor. El factor de escalamiento C_e esta dado por:

$$C_e = \frac{e_{x,i} e_{y,i} e_{x,i}}{e_{x,f} e_{y,f} e_{x,f}}. \quad (3.4)$$

Donde $e_{x,i}$ representa el número inicial de renglones del campo, $e_{x,f}$ representa el número de renglones después de la interpolación, análogamente $e_{y,i}$, y $e_{y,f}$, para las columnas. Además se cumple que $C_e \leq 1$, de tal manera que se debe conservar la cantidad de la ecuación 3.5, que representa la suma de las normas de todos los vectores de velocidad:

$$G_e = \sum_{sx=1}^{e_{x,i}} \sum_{sy=1}^{e_{y,i}} \frac{\|a(sx, sy)\|}{A_r} = \sum_{sx=1}^{e_{x,f}} \sum_{sy=1}^{e_{y,f}} \frac{\|b(sx, sy)\|}{C_e^{-1} A_r}, \quad (3.5)$$

Donde las barras indican la norma, $a(sx, sy)$ es el vector en la posición sx , y sy para el campo no interpolado, $b(sx, sy)$ es el vector en el campo bajo la interpolación, y A_r es el área del campo visualizado. En la figura 3-4 aparece el campo de flujo cortante, pero ahora se hizo un refinamiento en las celdas por un factor $I = 2$. Y se repite el cálculo para la vorticidad y se obtiene nuevamente una constante $\vec{\zeta} = -2\vec{k}$. Tambien se calculó el factor G_e , para distintos niveles de interpolación como se muestra en la figura 3-3. Al factor de interpolación $I = 1$, le corresponde un valor de $G_e = 5,50 \text{ 1/(cm} \cdot \text{s)}$ donde $A_r = 100\text{cm}^2$. Cuando $I \rightarrow \infty$ entonces $G_e = 5,00 \text{ 1/(cm} \cdot \text{s)}$. Comparando esto con un campo continuo de velocidades de flujo cortante con las mismas condiciones del ejemplo anterior, se puede calcular G_e de acuerdo a;

$$\frac{1}{A_r} \int_{\Gamma} \|v(x, y)\| \delta x \delta y,$$

Γ es la curva que encierra el campo de velocidades $v(x, y) = cy$ con $c = 1$. Simplificando la integral.

$$\frac{1}{A_r} \int_0^{10} y \cdot \delta y \int_0^{10} \delta x = \frac{1}{100} \frac{y^2}{2} \Big|_0^{10} x \Big|_0^{10} = 5,0.$$

El propósito de este ejemplo es demostrar que hay una cantidad G_e que converge al caso continuo cuando el factor de interpolación $I \rightarrow \infty$. La idea detrás es relacionar a G_e con la cantidad de masa en movimiento para un campo de velocidades dado, y demostrar que

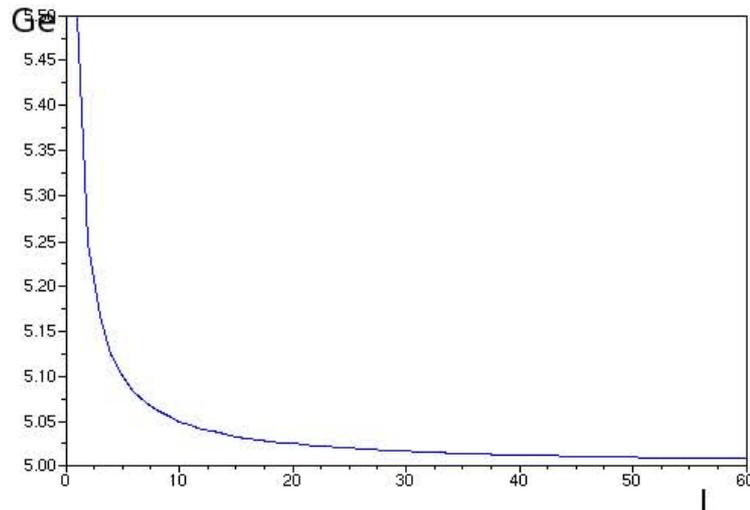


Figura 3-3: En el eje vertical el parámetro G_e , en el horizontal, I , que representa los distintos niveles de interpolación.

no depende sustancialmente de la resolución del campo, ya que en este caso la diferencia máxima entre los distintos valores de G_e es de 9%.

3.2. Visualización de campos mediante OpenDX

Muchas veces tener una gráfica de los datos, ya sean simulados o medidos, puede ofrecer un panorama general del fenómeno en cuestión. Existen diversos paquetes y librerías para ello, desde una simple gráfica en dos dimensiones hasta gráficas mucho más sofisticadas, como sería el caso de visualización de líneas de corriente en un túnel de viento o la contaminación de un flujo con otro como función del tiempo. Este caso es más sencillo, debido a que son campos bidimensionales. Aun así es muy útil tener una visualización en uno de estos paquetes. El que se eligió por su disponibilidad, y por su fácil manejo fue **OpenDX**.

OpenDX es una aplicación y un paquete de desarrollo de software para visualizar datos, especialmente los adquiridos a partir de observaciones o simulaciones. Usa una Interfaz

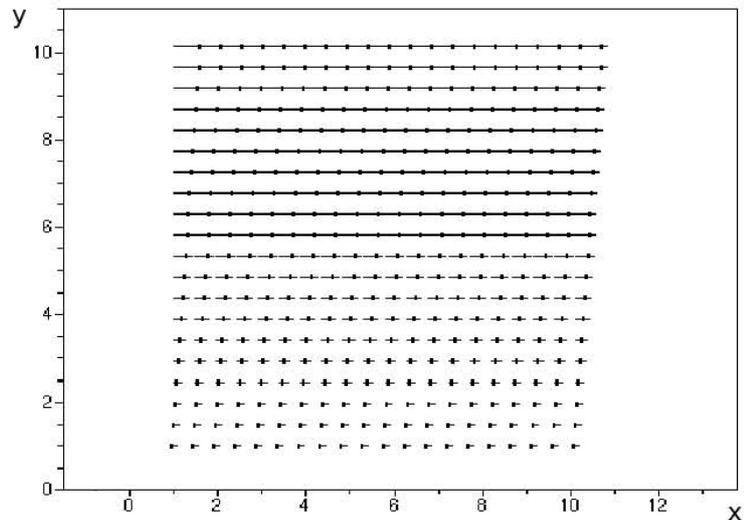


Figura 3-4: Campo de velocidades con el doble de puntos obtenidos por interpolación. Ambos ejes están graduados en cm.

Gráfica que provee de un conjunto completo de herramientas para desarrollar y observar los datos. En 1991 fue creada para la comunidad de visualización IBM Data Explorer. Desde entonces, ha sido ampliamente usada en la industria, academia, gobierno e investigación. El formato de datos es lo suficientemente flexible para trabajar en una amplia variedad de aplicaciones, incluyendo química, diseño de dispositivos, ciencias ambientales y medicina, entre otras. El 18 de Mayo de 1999, IBM publicó el código fuente y nació **OpenDX**, que es ahora de libre distribución, <http://www.opendx.org>. El paquete en su versión completa está acompañado por un tutorial y alrededor de 50 ejemplos, que para esta aplicación resultó ser suficiente. Se describe cómo se introduce los datos, ya que hay muchas páginas web que describen de manera más general este programa. Una de ellas, <http://www.facyt.uc.edu.ve/opendx>

Todos los cálculos de **PIV**, y de vorticidad, están hechos en **Scilab**, así que es necesario preparar los datos para su importación posterior con **OpenDX**, Primero, se guarda en un archivo de texto plano *.txt*, En el caso del campo de velocidad, por ser un campo vectorial, se utiliza el formato que se presenta a continuación, donde cada pareja de datos $datos_x(i, j)$ $datos_y(i, j)$, tiene la información de un vector de velocidad.

```
dato_x(1,1) dato_y(1,1) dato_x(1,2) dato_y(1,2) ...dato_x(1,m) dato_y(1,m)
```

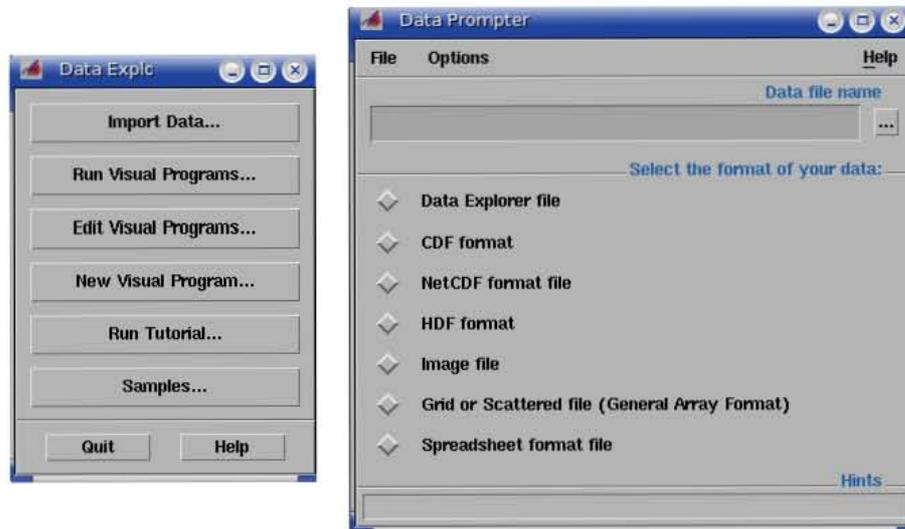


Figura 3-5: Interface de **OpenDX** para importar datos

```

dato_x(2,1) dato_y(2,1) dato_x(2,2) dato_y(2,2) ...dato_x(2,m) dato_y(2,m)
.
.
dato_x(n,1) dato_y(n,1) dato_x(n,2) dato_y(n,2) ...dato_x(n,m) dato_y(n,m)

```

Una vez guardado en un archivo, por ejemplo *flujocortante.txt*, se importa con *OpenDx-Data Promt*, como se muestra en la figura 3-5.

Se indica la ubicación del archivo en la estructura de directorios, en opción de *Grid or scattered file*. Se indica que son vectores de 2 dimensiones. Con esto se genera un archivo denominado de cabecera, que contiene la información de cómo leer los datos. El archivo de cabecera tiene extensión *.general*, por ejemplo, *flujocortante.general*. Lo siguiente es elaborar el programa de visualización con el lenguaje de objetos que representa propiamente **OpenDX**. como se muestra en la figura 3-7. Los rectángulos representan los módulos de programación. Están unidos por líneas, que describen el flujo de ejecución. Se cuenta con 167 módulos para distintas tareas, desde redimensionar una matriz hasta texturizarla o animar un conjunto de cuadros. También existe la posibilidad de editar nuevos módulos a partir de lenguaje *script*, propio del programa. Para visualizar el campo de velocidades se

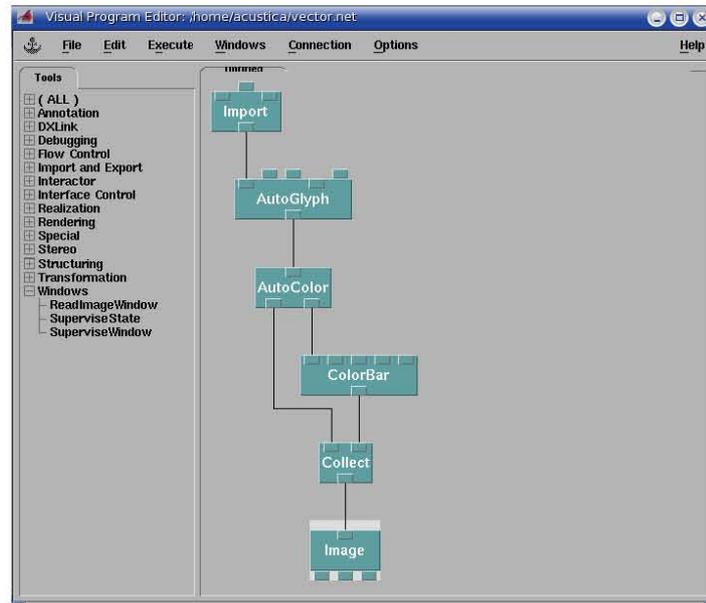


Figura 3-6: Interface de Programación modular de **OpenDX**

utilizaron los módulos que se muestran en el cuadro 3.1.

Módulo	Descripción
Import	Carga en memoria el archivo de datos a visualizar
AutoGlyph	Asigna vectores a los datos de acuerdo a como estan organizados
AutoColor	Asigna color a los vectores provenientes del módulo
ColorBar	Dibuja una barra indicadora de magnitud dependiendo de los datos
Collect	Recibe los datos provenientes de distintos módulos
Image	Despliega en pantalla los resultados, hace las gráficas

Cuadro 3.1: Módulos para visualizar el campo de velocidades

Al ejecutar el programa, aparece automáticamente la ventana con las gráficas como se indicó en el editor de módulos. Cada módulo, tiene un menú de configuración, donde es posible cambiar parámetros como la escala de datos, amplificación, tipo de color y algunas

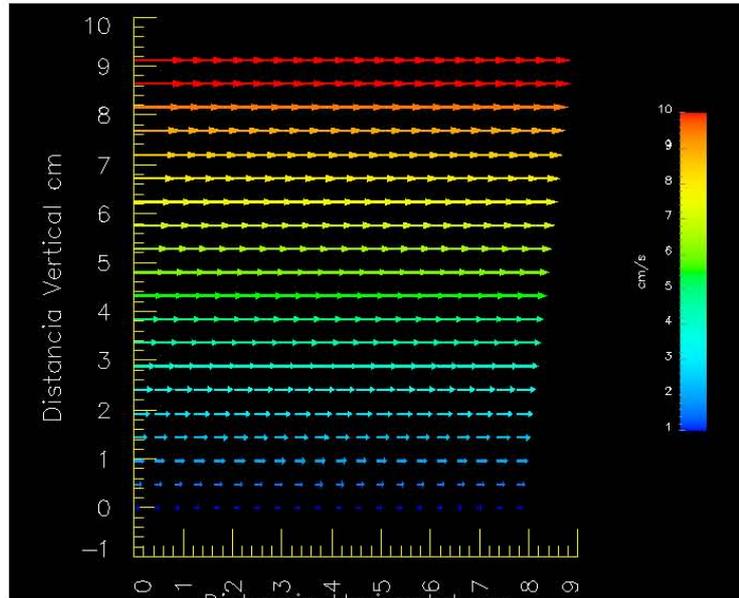


Figura 3-7: Campo de Velocidades para el flujo cortante, visualizado con **OpenDX**, Los ejes estan graduados en *cm*, la magnitud de los vectores esta indicada por la barra de color.

otras características. Las librerías de **OpenDX** son compatibles con lenguajes como *C++*, o *Fortran*, además de contar con su propio editor de *script*. En la figura 3-7, se observa el campo de velocidades de flujo cortante.

Capítulo 4

Resultados

En este capítulo se muestran los campos de velocidad y de vorticidad obtenidos con la técnica descrita en el capítulo dos. En la primera parte se estudian los flujos producidos alrededor de un tubo y una trompeta cuando un pistón produce un flujo sinusoidal interno. El objetivo es determinar el efecto de la geometría del difusor en la producción de vórtices para diversas amplitudes y frecuencias de oscilación. En la segunda parte se trata de determinar el amortiguamiento de una columna de agua que oscila sin forzamiento a partir de una amplitud inicial.

4.1. PIV aplicado a la salida del tubo

Se probaron los algoritmos de **PIV**, para dos difusores. En primer lugar se probó con un tubo plano de diámetro interno de 3,40 cm y un diámetro externo de 3,81 cm, que sirvió como referencia. Se montó el tubo recto en el dispositivo experimental, se sembraron finas partículas de pintura metálica en el fluido, y se hizo pasar la hoja de luz. La cámara permaneció perpendicular a la hoja de luz, y se grabó todo el ciclo justo en la salida del tubo. En particular elegimos un período de 1,66 segundos, equivalente a una frecuencia de 0,60 Hz y con una amplitud de 10,0 cm. Se grabaron varios ciclos. Esta parte, que corresponde a la toma de datos, es bastante rápida. En general bastan unos cuantos segundos para obtener las imágenes necesarias.

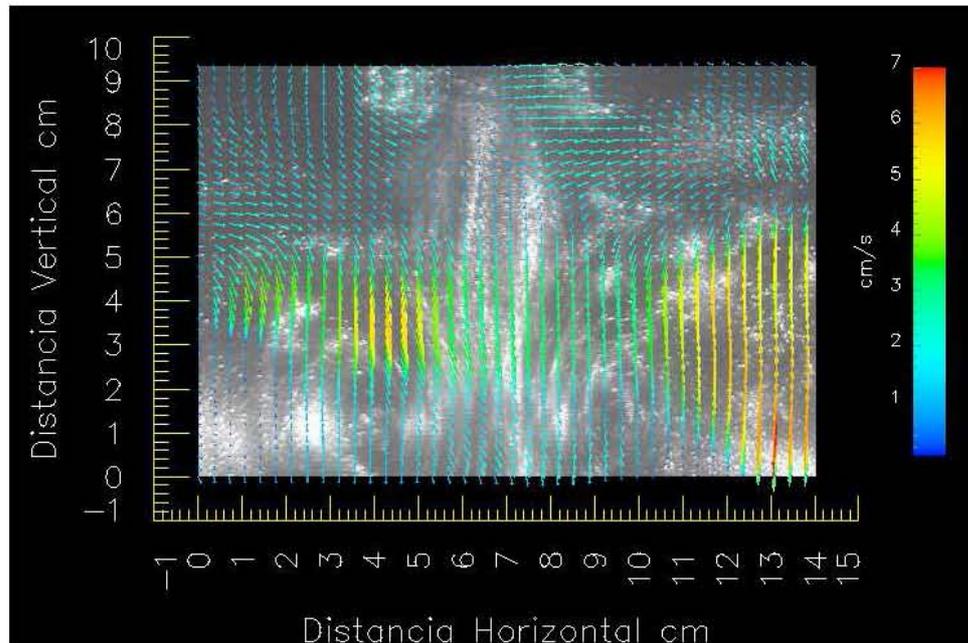


Figura 4-1: Imagen de **PIV** a la salida del difusor, cuando la altura de la columna de agua en el interior es máxima

El siguiente paso fue conectar la cámara a la computadora mediante el IEEE1394, y transferir el conjunto de imágenes en escala de grises y formato tiff al disco duro. La resolución del **PIV** en este caso fue de 20×20 celdas de interrogación, lo cual es equivalente a 400 celdas. El tiempo aproximado para este cálculo fue de 10 segundos. El resultado fue una gráfica donde se muestra el campo de velocidades (ver figura 4-1), correspondiente a la posición más alta del pistón, es decir, cuando el tubo está completamente lleno de agua. Este instante es el inicio del ciclo. El programa no se detiene, continua el análisis con el siguiente par de imágenes, lo que arroja la segunda instantánea para el campo velocidades y así sucesivamente con el resto.

El programa no se detiene hasta que ha procesado todas las imágenes que se encuentran en la carpeta, que en este ejercicio fueron 63. Por lo tanto, se pueden formar 62 pares y sus correspondientes campos de velocidades.

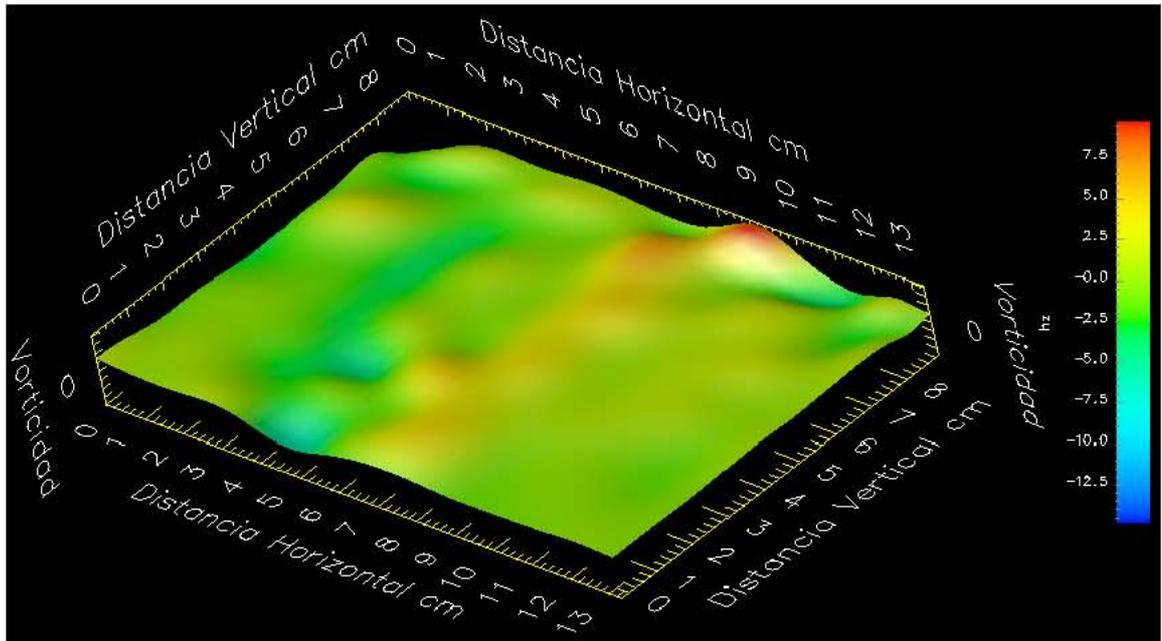


Figura 4-2: Vorticidad a la salida del tubo. La escala de magnitud esta indicada por la barra de color.

Utilizando el campo de velocidades de la figura 4-1, se calculó el campo de vorticidad que se muestra en la figura 4-2.

De acuerdo a la gráfica 4-2, los valores de la vorticidad son tanto positivos como negativos. Se espera que la distribución de la vorticidad, ya sea por forma y signo, tenga cierta simetría axial respecto a un eje que corta el difusor justo a la mitad. En esta gráfica no es muy claro, pero se debe a varios factores, uno de los mas importantes es que la iluminación no es simétrica, entrando por el lado derecho y perdiendo intensidad hacia el izquierdo.

Aplicar las herramientas de **PIV**, a la salida del tubo, permitirá cuantificar los campos de velocidad y vorticidad. Además abre la posibilidad de obtener otros resultados, en el caso de que se agreguen nuevas funciones, tales como líneas de vorticidad o de corriente. En primer lugar se estudiará el tubo plano mostrando algunas imágenes y gráficas del

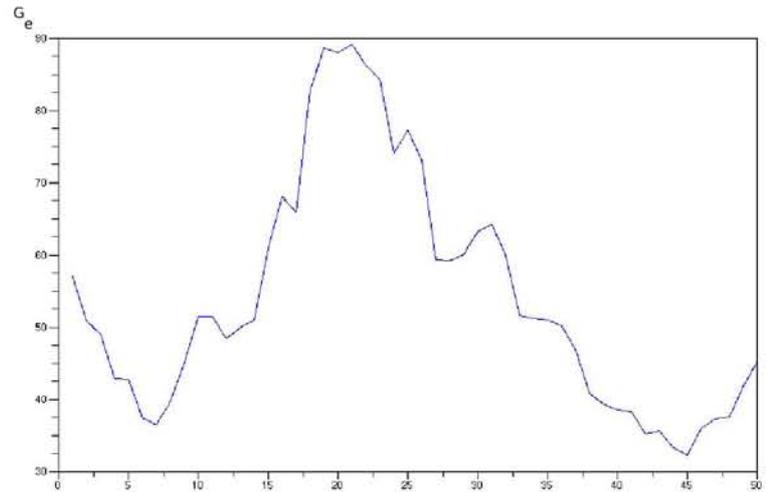


Figura 4-3: Gráfica del parámetro G_e , contra el tiempo medido en cuadros, en el eje horizontal

comportamiento que presenta con este análisis para después hacer lo mismo con el difusor.

Se tiene una colección de 62 campos de velocidades, de entre ellas elegiremos para analizar las que arrojen información de los casos extremos, ya sea para la vorticidad máxima, mínima, o aquellos en que la suma de los vectores de velocidad sea máxima o mínima. Esto último está relacionado con la cantidad de masa de agua puesta en movimiento por la oscilación G_e ¹. El resto de los casos está contenido en este análisis. Si se calcula G_e para cada uno de los cuadros como se muestra en la figura 4-3. El máximo de esta gráfica corresponde al cuadro 21, que corresponde a 0.70 segundos después de empezado la oscilación desde su máxima altura.

Los campos de velocidad y de vorticidad para el cuadro 21, se muestran en las figuras 4-4, y 4-5, respectivamente.

¹Parámetro definido en la sección 3.1

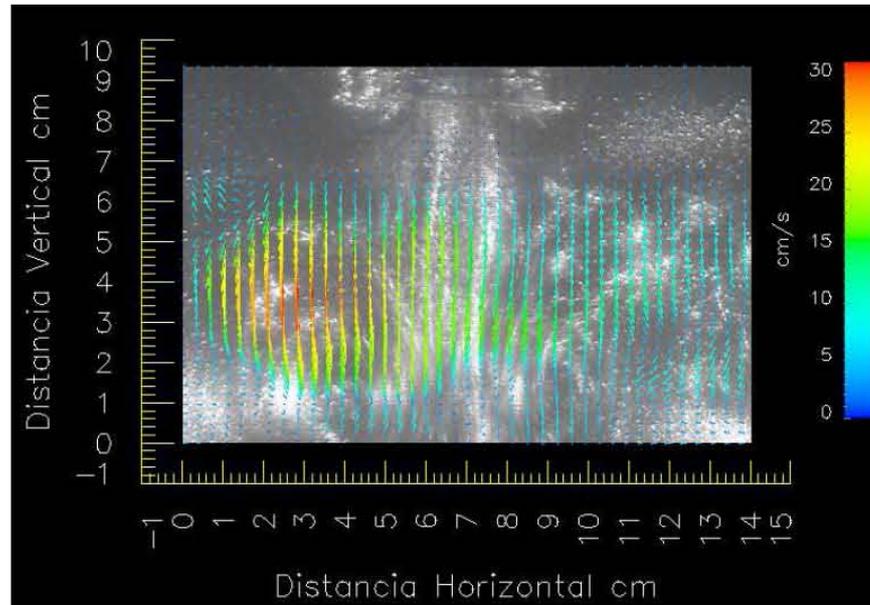


Figura 4-4: Campo de velocidades, donde la masa de agua en movimiento es máxima

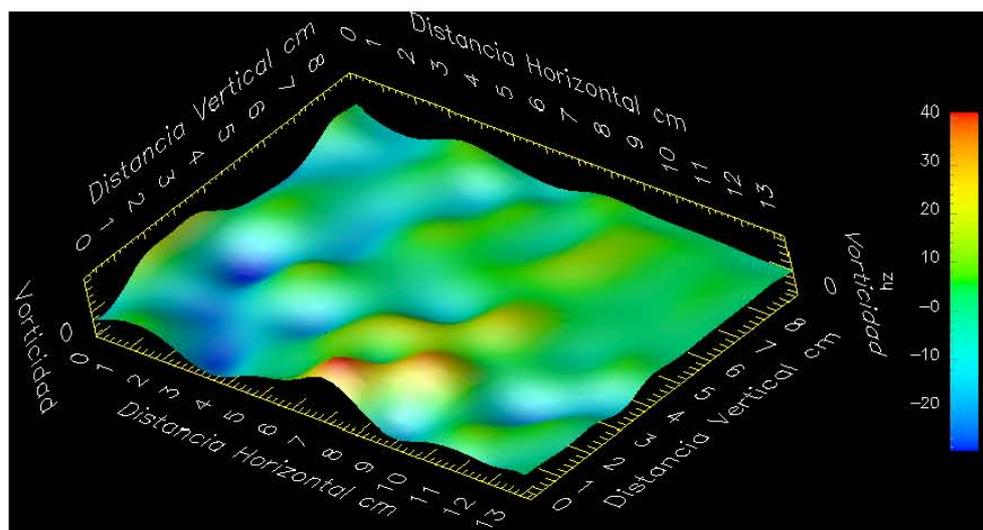


Figura 4-5: Gráfica de la vorticidad, donde la masa en movimiento es máxima

El valor que alcanza el parámetro G_e , en el cuadro 21, es de $88,7 \text{ 1}/(\text{cm} \cdot \text{s})$. La vorticidad máxima para el cuadro 21, es de 40.1 Hz , El valor mínimo es de -29.7 Hz , como se muestra en la figura 4-5. Análogamente al parámetro G_e para la velocidad, se puede calcular la función ζ_{total} , para la vorticidad, como se muestra en la ecuación 4.1.

$$\zeta_{total} = \sum_{sx=1}^{ex,f} \sum_{sy=1}^{ey,f} \frac{\|\vec{\zeta}(sx, sy)\|}{C_e^{-1} A_r}. \quad (4.1)$$

Donde $\vec{\zeta}(i, j)$ es el vector de vorticidad en el punto (i, j) , $e_{x,f}$, y $e_{y,f}$ son el tamaño del campo de vorticidad, C_e es el parámetro de interpolación (ver sección 3.1) y A_r es el área del campo. Para el caso del cuadro 21 $\zeta_{total} = 79,6 \text{ Hz}$. Sin embargo la vorticidad máxima no se encuentra en el cuadro 21, sino en el 18. que corresponde a 0,60 segundos después del inicio. Toma un valor de $89,96 \text{ Hz}$.

4.2. Aplicación al difusor con terminación circular

Se hace un proceso similar analizando con **PIV** los casos extremos tanto para la velocidad, como para la vorticidad. En este caso se utiliza un difusor circular con las siguientes características. La parte interior del difusor tiene un diámetro de $3,81 \text{ cm}$, la parte exterior del tubo alcanza un diámetro de $14,50 \text{ cm}$, el espesor de la pared, es de $0,15 \text{ cm}$. En la figura 4-6 se muestra una imagen de este difusor.

Se calculó G_e para este difusor y se graficó en función de los cuadros. La frecuencia de oscilación es de 0.80 Hz , y la amplitud es de 10 cm .

En la figura 4-7 se observa la existencia de dos máximos, y un mínimo al centro de la oscilación. Los máximos alcanzan un valor de $2,44 \text{ cm/s}$ al tiempo $0,3 \text{ s}$, y de $2,62 \text{ cm/s}$ al tiempo $0,73$ segundos respectivamente. El mínimo central tiene un valor para $V_{total} = 0,59 \text{ cm/s}$ en un tiempo de $0,53$ segundos. A diferencia del tubo en el que solo había un máximo, en este caso tenemos dos, y un mínimo entre ellos. Esto se debe a que cuando el pistón está en la parte baja la mayor parte del agua se encuentra en la zona amplia del difusor

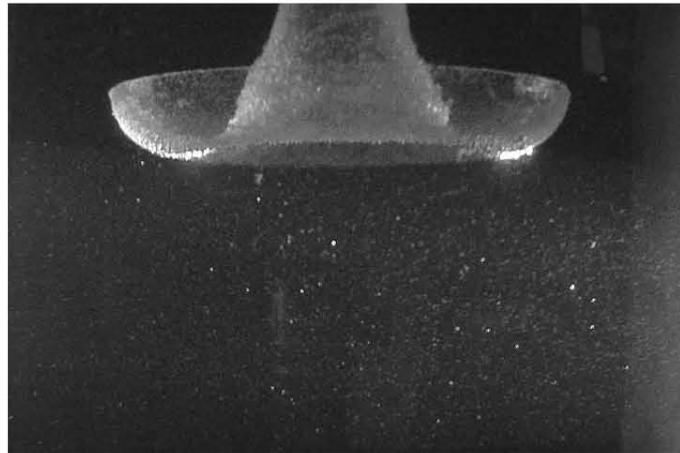


Figura 4-6: Difusor con terminación circular, La parte mas ancha de la salida alcanza 14.5 cm, la parte mas delgada mide 3.81 cm.

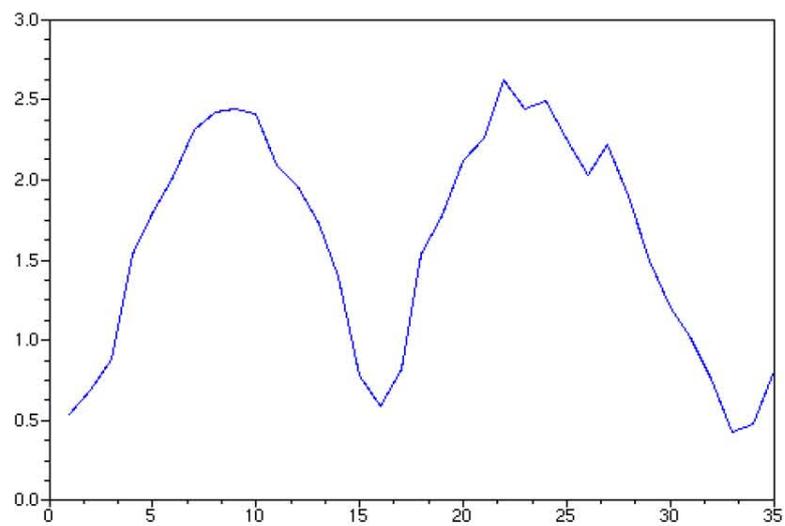


Figura 4-7: Gráfica para la función G_e . El eje horizontal, tiene unidades de tiempo, cada unidad equivale a 0,03 segundos.

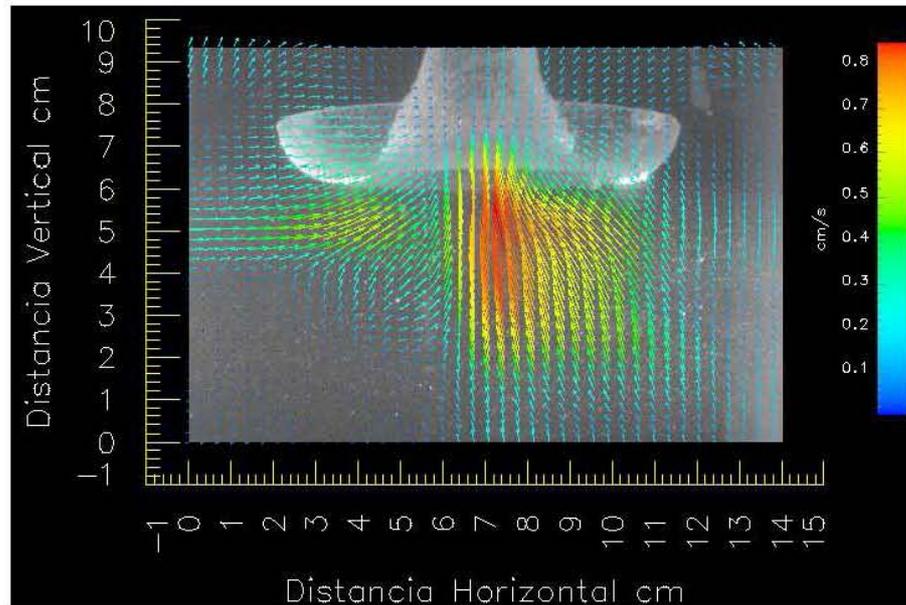


Figura 4-8: **PIV** para el difusor con terminación circular a un tiempo de 0.73 segundos de la posiciónn máxima de la columna de agua.

donde la velocidad ha disminuido por el aumento del área transversal.

4.3. Amortiguamiento de la oscilación de una columna de agua

El flujo en tuberías ha sido ampliamente estudiado debido a las multiples aplicaciones que tiene. Sin embargo se sabe poco sobre las pérdidas en tuberías cuando el flujo es oscilante. En esta sección se estudia el amortiguamiento de una columna de agua que oscila libremente, es decir sin forzamiento. Para generar esta oscilación, la columna de agua es levantada una cierta altura h_0 dentro del tubo como se muestra en la figura 4-10, después se suelta y se deja oscilar hasta que se detiene. Mientras esto ocurre se graban las imágenes dentro y fuera del tubo para su análisis posterior. Esto permite relacionar el campo de velocidades corresponde con la posición de la columna de agua en el tubo.

Las posiciones de la columna de agua se registraron por con video con una frecuencia

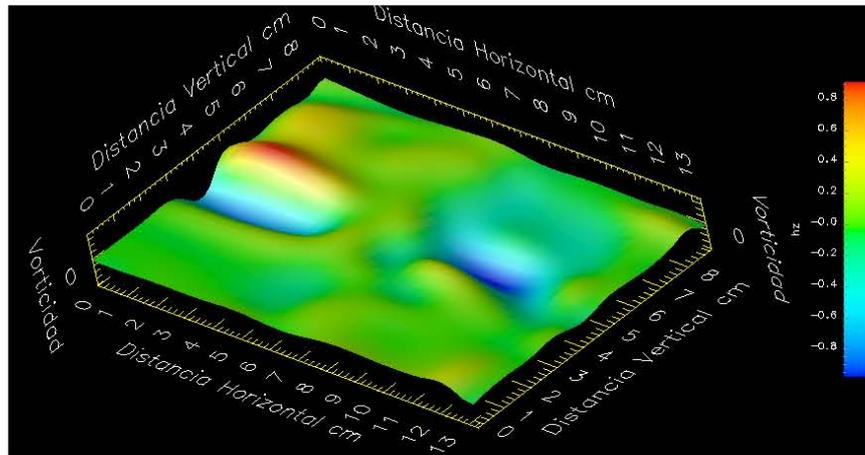


Figura 4-9: Gráfica de vorticidad para el campo de velocidades de la imagen 4-8.

de 29,97 cuadros por segundo. En la figura 4-11 se muestra la amplitud de la oscilación en función del tiempo para tres eventos. La altura inicial es $h_0 = 23,10 \pm 0,10$ cm. La incertidumbre en esta medida se debe sobre todo a las ondulaciones en la superficie libre. Se hicieron otros dos experimentos para el mismo tubo con $h_0 = 14,60 \pm 0,10$ cm y $h_0 = 23,80 \pm 0,10$ cm.

Se observa que aunque las amplitudes iniciales tienen valores distintos la frecuencia de oscilación conserva un comportamiento similar en todos los casos. Para el primer evento se capturaron 189 posiciones distintas, lo que equivale a 6.30 segundos. El período promedio τ de oscilación para los tres eventos es $0,80 \pm 0,02$ segundos, que en frecuencia ν equivale a $1,25 \pm 0,03$ hz. Con la resolución que se maneja para este experimento no se observan variaciones en el período a medida que decae la amplitud.

Todas las gráficas muestran un decaimiento en la amplitud de oscilación. Se sabe una que parte de la energía se pierde en la formación de vórtices. De ahí que también sea necesario ver la evolución del campo de velocidades en la salida a medida que oscila la columna. Otra parte se pierde por fricción con la superficie del tubo.

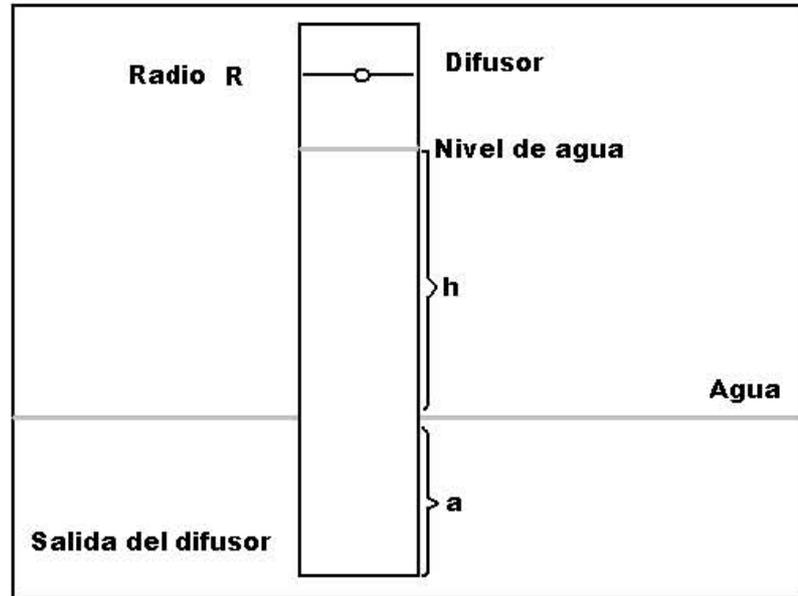


Figura 4-10: Esquema del dispositivo para la oscilación de la columna de agua.

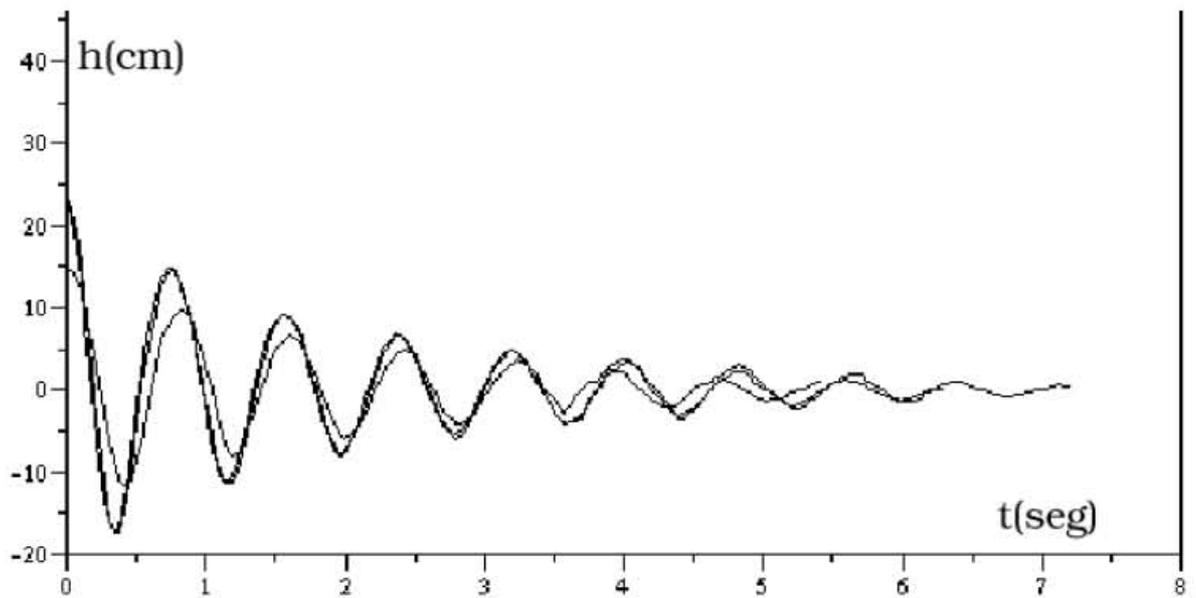


Figura 4-11: Posición de la columna de agua como función del tiempo, los valores iniciales son $h_0 = 23,10, 23,80, 14,60$ cm.

Se supuso primero que el decaimiento de la amplitud de la columna de agua obedece una regla exponencial. Se ajustaron dos curvas que envuelven las oscilaciones de la figura 4-11. Para obtener la ec. 4.2 se utilizaron los máximos y para la ec. 4.3 los mínimos. Ambas series de puntos se trazaron en un plano semilogarítmico y luego se les ajustó una recta por mínimos cuadrados. La varianza de la pendiente y de la ordenada al origen en ambos casos son $\Delta m = 0,02 \text{ hz}$, y $\Delta b = 1,00 \text{ cm}$ [7] respectivamente.

Las ecuaciones 4.4 y 4.5 fueron determinadas a partir de los datos suponiendo un decaimiento exponencial. La forma se puede determinar a partir de los máximos (ecuación 4.4) o de los mínimos (ecuación 4.5).

$$h_{max} = (19,70 \pm 1,00)e^{-(0,43 \pm 0,02)t} \quad (4.2)$$

$$h_{min} = (-19,20 \pm 1,00)e^{-(0,42 \pm 0,02)t} \quad (4.3)$$

Donde h_{max} y h_{min} son las posiciones de la columna de agua. Los valores de los exponentes se diferencian apenas un 2,32%. Para los eventos 2 y 3 se calcularon las expresiones que se muestran en el cuadro 4.1 siguiendo el mismo procedimiento.

h_0	E.S.	E.I.	ecuaciones
$21,10 \pm 0,10$	$(19,70 \pm 1,00)e^{-(0,43 \pm 0,02)t}$	$(-19,20 \pm 1,00)e^{-(0,43 \pm 0,02)t}$	4.6, 4.7
$14,60 \pm 0,10$	$(14,00 \pm 1,00)e^{-(0,43 \pm 0,02)t}$	$(-14,60 \pm 1,00)e^{-(0,46 \pm 0,02)t}$	4.8, 4.9
$23,80 \pm 0,10$	$(22,00 \pm 1,00)e^{-(0,48 \pm 0,02)t}$	$(-19,60 \pm 1,00)e^{-(0,45 \pm 0,02)t}$	4.10, 4.11

Cuadro 4.1: La primera columna corresponde con los valores iniciales de h_0 , las unidades son cm. E.S (Envolvente Superior), E.I. (Envolvente Inferior), ambas son los ajustes calculados para cada evento.

La figura 4-12 muestra la comparación gráfica entre los puntos experimentales y las curvas ajustadas para la oscilación de la columna de agua con condiciones iniciales de

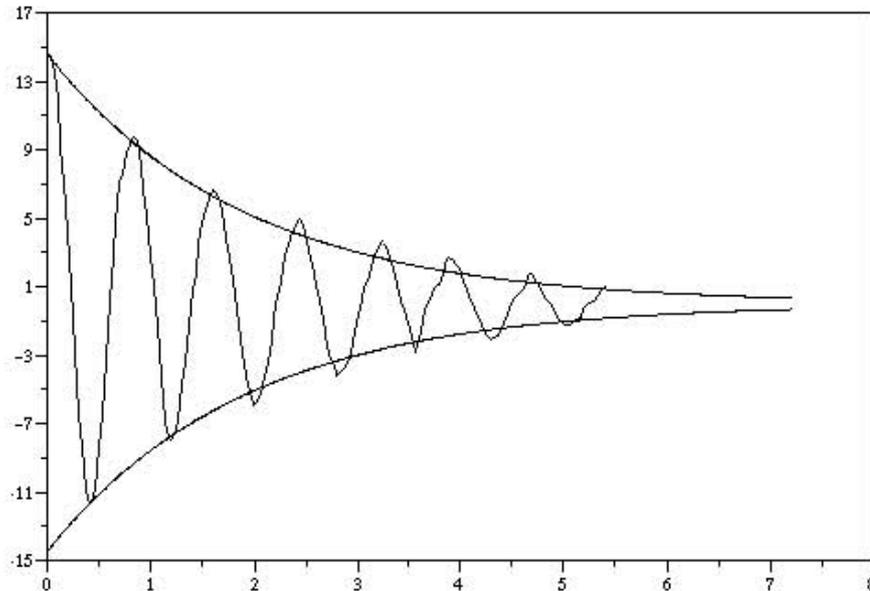


Figura 4-12: Ajuste exponencial al decaimiento de la columna de agua, para una altura inicial $h_0 = 14,60 \text{ cm}$.

$14,60 \pm 0,10 \text{ cm}$ sobre el nivel del tanque. Las curvas que forman la envolvente corresponden a las ecuaciones 4.8 en la parte superior y 4.9 en la inferior que aparecen en el cuadro 4.1.

Las curvas ajustadas difieren a lo mas en un 7% a los datos experimentales durante los primeros dos ciclos equivalentes a 1.44 segundos. Sin embargo para ciclos posteriores el error alcanza hasta un 20%.

La figura 4-13 muestra el *logaritmo* de la amplitud contra el tiempo. Se puede notar que la última parte de cada oscilación tiene una pendiente distinta a la que había al principio.

Por lo tanto para las últimas oscilaciones de cada evento se supuso que el cambio de amplitud puede describirse con un polinomio de grado 2. Se hizo entonces un ajuste exponencial para los primeros cuatro puntos de cada experimento, y los últimos puntos se ajustaron con un polinomio de grado 2. Se obtuvieron los resultados que se muestran en los

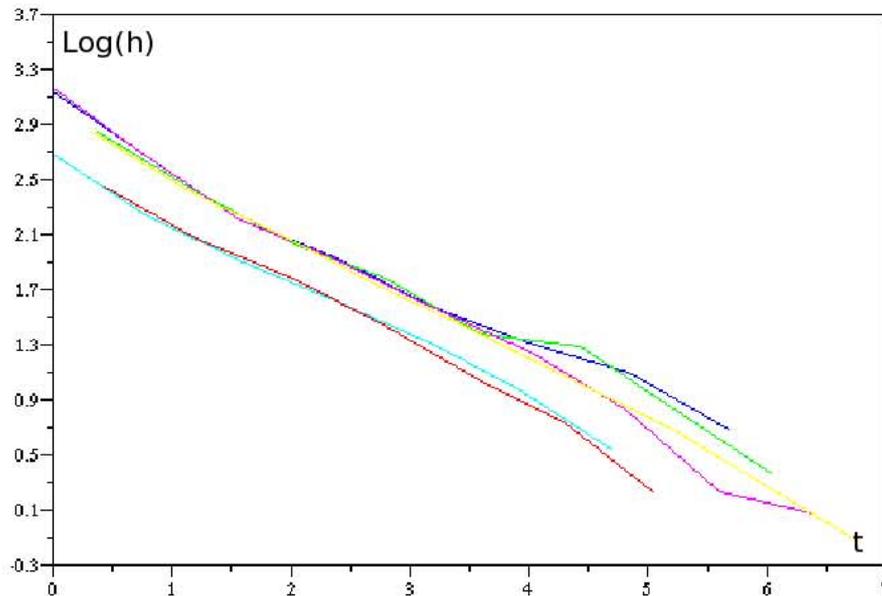


Figura 4-13: Decaimiento de la altura h para todos los eventos en un espacio semilogarítmico, en el eje horizontal el tiempo t .

cuadros 4.2, y 4.3.

En todos los ajustes es claro que la forma del decaimiento varía con la amplitud de oscilación. Para entender mejor lo que ocurre en el cambio de amplitud es necesario aplicar **PIV**. Si se comparan los campos de velocidades con los cambios en el decaimiento será posible entender los dos regímenes de decaimiento. Se sabe que el decaimiento se debe a la pérdida de energía por formación de vórtices, a la fricción con las paredes del tubo y a la masa añadida. Experimentos detallados en este sentido están en proceso.

También es interesante estudiar las frecuencias en función del tiempo durante el amortiguamiento. Las frecuencias parecen depender de la altura inicial. Sin embargo, contrariamente a lo que se esperaba, la frecuencia no parece variar con la amplitud para las condiciones iniciales dadas. Se espera poder realizar en el futuro experimentos más concluyentes.

$h_0 = 23,10 \pm 0,10$	A.E.	ecuaciones
E.S.	$(22,00 \pm 1,00)e^{-(0,51 \pm 0,04)t}$	4.12
E.I.	$(-19,70 \pm 1,00)e^{-(0,44 \pm 0,02)t}$	4.13

Cuadro 4.2: A.E. (Ajuste Exponencial), E.S. (Envolvente Superior), E.I. (Envolvente Inferior).

$h_0 = 23,10 \pm 0,10$	A.P.	ecuaciones
E.S.	$0,28 \pm 0,01t^2 - 3,68 \pm 0,01t + 13,92 \pm 0,23$	4.14
E.I.	$-0,13 \pm 0,01t^2 + 2,43 \pm 0,01t - 11,52 \pm 0,23$	4.15

Cuadro 4.3: A.P. (Ajuste de segundo grado), E.S. (Envolvente Superior), E.I. (Envolvente Inferior).

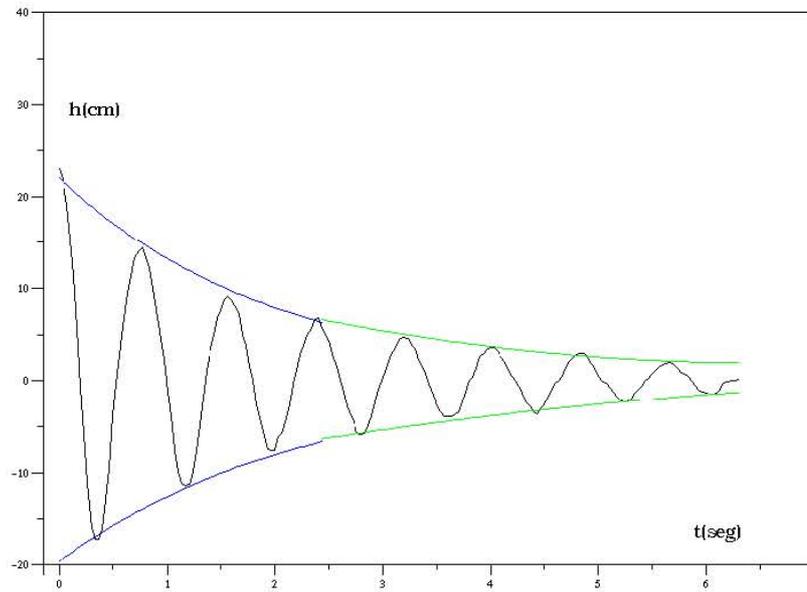


Figura 4-14: Gráfica del decaimiento de la altura $h(t)$ para una amplitud inicial $h_0 = 23,10 \pm 0,10$ cm En color azul aparece el ajuste exponencial, en verde el cuadrático.

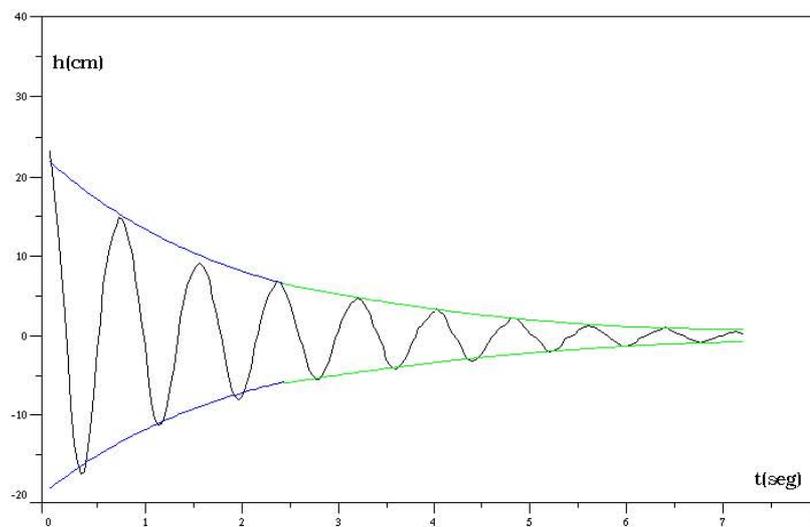


Figura 4-15: Gráfica del decaimiento de la altura $h(t)$ para una amplitud inicial $h_0 = 23,81 \pm 0,10$ cm En color azul aparece el ajuste exponencial, en verde el cuadrático.

Capítulo 5

Conclusiones y perspectivas

En este capítulo se presentan las conclusiones y perspectivas sobre el efecto de la geometría en la salida de los flujos oscilantes, ya sea para el caso plano o para el de un difusor curvo. Se lograron cuantificar el campo de velocidad y el de vorticidad para ambas geometrías, Se podrían cuantificar otras propiedades como las líneas de corriente o de vorticidad, pero esto corresponde a un trabajo futuro. El comportamiento mostrado por los difusores es radicalmente distinto entre sí. El tubo plano genera altos valores de la vorticidad respecto al difusor curvo lo cual ya había sido observado cualitativamente.

El objetivo principal de este trabajo se cumplió ya que se hizo funcionar un programa global de **PIV** que incluye el trabajo de diversos autores y distintos programas incluyendo funciones en **SCILAB**. Existen diversos sistemas de **PIV** en el mercado, y casi todos cuentan con el equipo electrónico y los programas apropiados. Sin embargo debido a su alto costo no son asequibles por cualquier laboratorio. Nuestra propuesta consiste en formar parte del desarrollo de código abierto que en este caso se utiliza para hacer las mediciones del campo de velocidades, y poner en la medida de lo posible estas herramientas al alcance de personas interesadas.

5.1. Conclusiones

La mayor parte de este trabajo fue orientada al diseño y programación de funciones de **PIV** en ambiente *SCILAB*. Se hizo un análisis minucioso de los algoritmos de *velocimetría por imágenes de partículas*. El método utilizado es un algoritmo estándar al que se le

agregaron algunas funciones extra como es el caso del algoritmo interpolador que se muestra en el apéndice y del que se habla brevemente en la sección 4.5. También se incorporó el algoritmo para generar campos de vorticidad a partir de campos de velocidades. En general se adjuntaron varias funciones más para el manejo de imágenes directamente con *SCILAB*, pero la verdadera esencia del programa permaneció intacta de acuerdo a los algoritmos estándar que se manejan en el código abierto sobre **PIV**. Tal vez más adelante estos estándares alcancen la correlación de segundo orden que se comienza a manejar ya por algunas empresas.

Los resultados obtenidos con el programa son satisfactorios, aunque se debe hacer mejoras en varios aspectos. Por ejemplo, se debe ajustar las escalas durante el análisis a una secuencia de fotogramas ya sea para medir campos de velocidad o vorticidad, pues no se sabe previamente cual de todos los campos calculados tendrá la magnitud mayor. Otro de los problemas que hay que corregir es la interface de usuario, ya que hasta el momento para trabajar en algun caso específico hay que editar directamente líneas de código. En realidad solo hay que dar las rutas para acceder a los fotogramas y eso es suficiente. Desde luego se pueden modificar otros parámetros y llegar a personalizar el código de manera que resulte más eficiente para el caso con el que se trabaje. Para solucionar esto se propondrá como trabajo futuro desarrollar una sencilla pero útil interfaz gráfica.

De acuerdo a la sección 4.3 donde se muestra que la disipación de energía por vórtices está en función de la geometría de la entrada, es posible afirmar que la formación de vórtices es mucho menor para la salida curva que para el tubo. En el cálculo mostrado en la misma sección se hace la comparación de campos de velocidad para ambas salidas. De esta forma es posible concluir que un difusor con la curvatura adecuada produce un campo de vorticidad mucho menor ¹ que un simple tubo recto. La relación de cómo cambia la forma de los campos de velocidad y vorticidad respecto a la geometría del difusor requiere un trabajo más detallado en este aspecto específico y se propone para trabajos posteriores. Para el

¹Hablando en terminos de magnitud promedio de dicho campo

SIBEO esta información es relevante.

5.2. Trabajo futuro

Existe una serie de propuestas para desarrollar en trabajos posteriores, algunas han sido mencionadas a lo largo de la tesis. Es posible dividir en tres distintos tipos de trabajo, El primero se puede catalogar en computación que equivale a hacer una serie de mejoras a los actuales algoritmos y a la interacción del usuario con el programa. El segundo está relacionado con el dispositivo experimental, al que se le pueden hacer algunas mejoras, ya sea fabricando piezas mecánicas nuevas ó incorporando un control electrónico de mayor precisión. La tercera y última se puede catalogar en la experimentación, es decir, comparar el comportamiento que exhibe el campo de velocidades o el de vorticidad, al combinar distintos valores de amplitud y frecuencia o curvatura de la entrada sobre el flujo oscilante. Algunas de las propuestas aunque parecen independientes de otras, en realidad no lo son, por ejemplo; para el caso de ampliar el experimento analizando el comportamiento de alguna trompeta o salida para ciertas condiciones de amplitud y frecuencia puede ser necesario modificar el dispositivo experimental.

En la parte de computación las propuestas están orientadas en el sentido de ampliar las capacidades de análisis del o de los programas para obtener más información a partir de los campos de velocidades, por ejemplo calcular las líneas de corriente [28]. También dentro de las propuestas de cómputo, está diseñar un programa interactivo que permita el fácil manejo del programa sin que el usuario tenga que editar líneas de código.

Para la parte correspondiente al dispositivo experimental se propone diseñar un sistema de control electrónico que permita automatizar en la medida de lo posible tareas repetitivas, como es el caso de ir cambiando lentamente la amplitud y frecuencia de oscilación del pistón, y observar por ejemplo, que forma tienen los campos de velocidades y vorticidad mediante el programa de **PIV** para cada configuración de amplitud y frecuencia. Si el

dispositivo experimental garantiza estabilidad para hacer las mediciones, y la resolución para ir cambiando los parámetros es suficiente, se puede construir por ejemplo un mapeo sobre el espacio de parámetros amplitud frecuencia contra la vorticidad máxima en esa configuración. Un control electrónico que consideramos es posible implementar de manera sencilla a través del puerto paralelo de una computadora personal, utilizando el convertidor analógico digital *ADC0809-CCN*, que es de muy fácil operación ya sea en el envío o en la recepción de señales digitales. Mediante un pequeño programa en lenguaje *C* es posible controlar los distintos motores y demás partes eléctricas con los relevadores de potencia apropiados mediante el *ADC0809-CCN*.

Por último en lo que corresponde al experimento se propone catalogar distintos difusores mediante mapeos de parámetros, con esto tal vez sea posible hacer una conjetura empírica para describir qué relación guardan la vorticidad y la geometría en condiciones de oscilación específicas. Es posible incluir en esta parte el caso de la columna de agua que oscila libremente. En la sección 4.4 se muestra que el comportamiento de la amplitud no coincide del todo con un comportamiento exponencial. Se hace entonces una aproximación polinomial para la última parte de la oscilación. La propuesta para este caso está orientada a conocer que papel desempeñan la masa añadida, las pérdidas de energía por vorticidad, y la fricción con las paredes del difusor.

Bibliografía

- [1] S. Czitrom, R Godoy, E. Prado A. Olvera, C Stern, 2000, *Hydrodynamics of an oscilation water column seawater pump*. Part II: Tuning to Monochromatic waves Ocean Engineering 27, pp 1199-1219.
- [2] Stern Catalina, Azpeitia Carlos, Czitrom Steven, Prado Esteban, 2000, *New Observations on Production and Suppression of Vortices on Oscillating Flow through a Diffuser*, American Physical Society, 53rd Annual Meeting of the Division of Fluid Dynamics.
- [3] Drain, L.E. 1980, *The Laser Doppler Technique*, John Wiley and Sons Ltd.
- [4] Ferndez B., 2002, *Introducción a la mecánica de fluidos*, Alición Chile.
- [5] Edward Thomas Jr. 1999, *Direct measuraments of two-dimensional velocity profiles in the direct current glow discharge dusty plasmas*, Phisics of Plasmas, Volume 6 Number 7.
- [6] BAXES, Gregory, 1994, *Digital Image Processing : Principles and Applications*. Nueva York : John Wiley & Sons.
- [7] D.C. Baird. 1999, *Una introducción a la teoria de mediciones y al diseño de experimentos 2 edicion*, Prentice Hall, USA.
- [8] Jose L. Alva. 1999, *Métodos de Análisis de Imágenes*, Universidad de Vigo, Documento en linea
- [9] Low A. 1992, *Introductory computer vision and image processing*, McGraw Hill Book Co.

- [10] Y. M. Marsouk, D. P. Hart, 1998, *Asimmetric autocorrelation function to resolve directional ambiguity in PIV images*, Springer Verlag, pp 401-408.
- [11] R. Gonzalez and R. Woods, 1992, *Digital Image Processing*, Addison Wesley, pp 414 - 428.
- [12] Laurencio L. Krotapalli A. 1995, *On the accuracy of velocity and vorticity measurement with PIV*, Exp Fluids 18, 421-428.
- [13] Adrian R. J., Christensen K. T. Solof S. M., Meinhard C. D. and Liu Z. C. 1998 *Decomposition of turbulent field and visualization of vortices*, 9th Int. Simp. on Applications of Lasser Technics to fluid Mechanics, Portugal.
- [14] Daniel Hoyos, Elena Hoyos, Irene De Paul, 2003, *Avances en Energías Renovables y Medio Ambiente*, Vol. 7, ISSN 0329-5184.
- [15] Gonzalez R Woods R., 1992, *Digital Image Processing*, Addison Wesley.
- [16] Robert M. Haralick y Linda G. Shapiro, 1992, *Computer and Robot Vision*, Addison Wesley Publishing Co. ISBN 0-201-10877-1.
- [17] Späth, H., 1995, *One Dimensional Spline Interpolation Algorithms*. Wellesley, MA: A. K. Peters.
- [18] K, Atkinson, 1989, *An introduction to numerical Analysis*, John Wiley, New York.
- [19] Adrian 1991, *Particle-imaging techniques for experimental fluid mechanics*, Fluid Mech., Vol 23 pp 261-304.
- [20] Robert W., Fox, Alan McDonald, 1999, *Introducción a la mecánica de Fluidos*, 4 edición.
- [21] Burden, R.; Faires, D., 1998, *Análisis Numérico*. Ed. Thomson, 6a. ed.

- [22] M. Raffel, C Willert, J. Kompenhans, 1998, *Particle Image Velocimetry*, Springer Verlag, pp 106-120.
- [23] Goldstein R.J., 1983, *Fluid Mechanics Measurements*, 2nd edition, Tylor & Francis, Washington, D.C.
- [24] Carl D. Meinhart, Steve T., Santiago J., 2000, *A PIV Algorithm for Estimating Time-Average Velocity Field*, Journal of Fluids Engineering.
- [25] Westerweel J. 1993, *Eficient detection of spurious vectors in particle image velocimetry data*, Vol 16 pp 236-247.
- [26] Westerweel J. 1997, *Fundamentals of digits particle image velocimetry*, Meas.Sci. Technol Vol 8. pp 1379-1392.
- [27] Mika Piirto, Hannu Eloranta, Pentti, Saareninne., 2000, *Interactive software for turbulence Analysis from PIV vector Data*, Mechanical engineering department, Tampere University of Technology, Finland.
- [28] Robert W., Fox, Alan McDonald, 1999, *Introducción a la mecánica de Fluidos*, 4 edición.
- [29] Keane R.D., Adrian R.J. 1991, *Optimization of particle image velocimeters*, Vol 2, No. 10 Oct 1991-pp 963-974.
- [30] keane R.d., Adrian R.J. 1992, *Theory of cross-correlation analisys of PIV images*, Vol 49 pp 191-215.
- [31] Yeung A.F.K., Lee B.H.K. 1999, *Particle Image Velocimetry study of Wing-tip Vortices*, Journal of Aircraft 36, pp 207-210.
- [32] L. Lamport. 1994, *LaTeX, User's Guide & Reference Manual* Addison-Wesley Publish Company.

Appendix A

Código fuente del programa de PIV y de funciones asociadas

Para poder ejecutar los códigos fuente, es necesario tener instalado **Scilab** version **2.7.3** o **4**, el cual se puede obtener de manera gratuita en la página <http://www.scilab.org/>, ya sea la versión binaria o el código fuente. También es necesario instalar las librerías **SIP**, *Scilab Image Package* que se pueden obtener también de forma gratuita en <http://siptoolbox.sourceforge.net/>. La función *loader.sce* está incluida en **SIP**. Para más información sobre estas librerías se recomienda visitar la página.

A.1. Programa principal

El nombre para la ejecución de este programa es *velpiv.sci*, es el cuerpo principal del análisis numérico, desde aquí se coordinan las distintas rutinas y funciones.

```
clear;
exec("/home/acustica/sip/loader.sce");
if ~exists('dpiv') then getf('/home/acustica/PIVtool/macros/dpiv.sci'); end
if ~exists('medianvalid') then getf('/home/acustica/PIVtool/macros/medianvalid.sci'); end
if ~exists('interpola') then getf('/home/acustica/PIVtool/macros/interpola.sci'); end
if ~exists('vorticidad') then getf('/home/acustica/PIVtool/macros/vorticidad.sci'); end
if ~exists('magnitud') then getf('/home/acustica/PIVtool/macros/magnitud.sci'); end
if ~exists('imprime') then getf('/home/acustica/PIVtool/macros/imprime.sci'); end
if ~exists('imprime2d') then getf('/home/acustica/PIVtool/macros/imprime2d.sci'); end
stacksize(20000000)

// catx, y caty, son el nmero de celdas en las que sera dividida la imagen total
//el resto de los factores son ajustes de escala, resx, y resy, son el tamao de la
//imagen que se cargara en memoria
```

```

catx=20;caty=20;
mx=catx*5;my=caty*5;

numero=50      //aqui se indica el total de las imagenes a analizar

for j=1:numero
    archiv='' +string(j);
    archiv2='' +string(j+1);
    ima1=imread('/home/acustica/pinturap/' +archiv+'.tif');
    ima2=imread('/home/acustica/pinturap/' +archiv2+'.tif');
    [resx,resy]=size(ima1);
    ima1=ima1';
    ima2=ima2';
    iws=[150 100 64 45 30];
    presh=%t;
    if ~exists('mtr') then mtr=2*min(ima1); end

    nsx=mx * iws($)/iws(11); nsy=my * iws($)/iws(11);

    x=linspace(nix/(2*nsx),nix-nix/(2*nsx),nsx);
    y=linspace(niy/(2*nsy),niy-niy/(2*nsy),nsy);
    y=linspace(niy-niy/(2*nsy),niy/(2*nsy),nsy);

    pek=1;
    nsx=int(mx * iws($)/iws(pek)); nsy=int(my * iws($)/iws(pek));
    xo=x; yo=y;
    mask=bool2s(double(ima1(x,y))>mtr);
    mask=mask(1:size(svx1,1),1:size(svy1,2));

    [sx,sy]=dpiv(ima1,ima2,nsx,nsy,iws(pek),iws(pek),svx1,svy1,mask);
    [sx,sy]=medianvalid(sx,sy,0.1);
    factinter=2;

    [sx]=interpola(sx,facinter*size(sx));
    [sy]=interpola(sy,facinter*size(sy));

```

```

x=linspace(nix/(2*nsx),nix-nix/(2*nsx),nsx*facinter);
y=linspace(niy/(2*nsy),niy-niy/(2*nsy),nsy*facinter);

//Calculamos la magnitud del campo de velocidad con la siguiente funcin
//e imprimimos la informacin en un archivo de texto plano llamado sv(j).txt

campo='magnitudsv';
[sv]=magnitud(sx,sy);
[ok]=imprime(sv,'/home/acustica/pivtext/magnitudsv'+archiv,j,campo);

//Aqui imprimimos en un archivo la magnitud de la vorticidad
//Los parametros dfx, y dfy, indican el tamao fsico del campo de velocidades,
//en este caso cm.
dfx=18;dfy=12;
campo='vortexsv';
[vortex]=vorticidad(sx,sy,dfx,dfy);
[ok]=imprime(vortex', '/home/acustica/pivtext/vortexsv'+archiv,j,campo);

//Para los vectores de velocidad imprimir con la funcin imprime2d
[ok]=imprime2d(sx',sy', '/home/acustica/pivtext/vectorsv'+archiv,j);

//graficamos el campo de velocidades
champ(x,y,sx,sy,strf="041");

//Guardamos la imagen del campo de velocidades, opcionalmente, si no, se eliminan estas lneas
salida=string(j);
xs2gif (0,'/home/acustica/temporal/'+string(salida),0);
end

```

A.2. Función dpiv.sce

Esta función realiza el cálculo de **PIV** para cada par de celdas. fue desarrollada por Enrico Segre, *enrico.segre@weizmann.ac.il*.

```

function [sx,sy]=dpiv(i1,i2,nx,ny,iwx,iwy,presx,presy,mask)

[lhs,rhs]=argn(0);

//presx y presy seran integradas

[ix,iy]=size(i1)

if rhs<6 then
    if rhs==5 then iwy=iwx; end
    if rhs<5 then iwx=min(32,ix);iwy=min(32,iy); end
end

if rhs<3 then
    if rhs==3 then ny=nx; end
    if rhs==2 then nx=int(ix/32); ny=int(iy/32); end
end

if rhs<8 then
    presx=zeros(nx,ny); presy=zeros(nx,ny);
end

if rhs<2 then disp(" I need two images!"); return; end

sx=zeros(nx,ny); sy=zeros(nx,ny);

if ~exists("mask","local") then
    mask=ones(nx,ny)==1
else
    if size(mask,1)<nx | size(mask,2)<ny then
        write(%io(2), " mask is too small! "+string(nx)+"x"+string(ny)+..
            " needed, "+strcat(string(size(mask)),'x')+"found")
        return
    end
end

end

cx=(0:(nx-1))*(ix-iwx)/(nx-1)
cy=(0:(ny-1))*(iy-iwy)/(ny-1)

```

```

I1=zeros(2*iwx,2*iwy); I2=I1;

for i=1:nx
    for j=1:ny
        if mask(i,j) then
            I1=double(i1(cx(i)+(1:iwx),cy(j)+(1:iwy)))
            isx=min(max(0,cx(i)+presx(i,j)),ix-iwx)
            isy=min(max(0,cy(j)+presy(i,j)),iy-iwy)
            I2=double(i2(isx+(1:iwx),isy+(1:iwy)))
            C=real(fft(fft(I2,-1).*conj(fft(I1,-1)),1))
            exec("/home/acustica/PIVtool/macros/subpixel.sci");
            [m1,m2]=subpixel(fftshift(C))
            sx(i,j)=m1-floor(iwx/2)-1; sy(i,j)=m2-floor(iwy/2)-1;
        end
    end
end

end

sx=sx+presx; sy=sy+presy;

```

A.3. Función medianvalid.sce

La función mediavalid corrige algún vector que pudiera alejarse del promedio de sus ocho vecinos mas que un cierto valor s_0 definido por el usuario.

```

function [svx,svy]=medianvalid(sx,sy,s0)

[nx,ny]=size(sx);
svx=sx;svy=sy;

for i=2:nx-1
    for j=2:ny-1
        smx=mean([sx(i-1,j-1),sx(i,j-1),sx(i+1,j-1),sx(i-1,j),sx(i+1,j),sx(i-1,j+1),sx(i,j+1),sx(i+1,j+1)])
        smy=mean([sy(i-1,j-1),sy(i,j-1),sy(i+1,j-1),sy(i-1,j),sy(i+1,j),sy(i-1,j+1),sy(i,j+1),sy(i+1,j+1)])
        if (sx(i,j)-smx)^2+(sy(i,j)-smy)^2 > s0^2 then
            svx(i,j)=smx; svy(i,j)=smy
        end
    end
end

```

```

        end
    end
end

for i=2:nx-1

    smx=mean([sx(i-1,1),sx(i+1,1),sx(i-1,2),sx(i,2),sx(i+1,2)])
    smy=mean([sy(i-1,1),sy(i+1,1),sy(i-1,2),sy(i,2),sy(i+1,2)])
    if (sx(i,1)-smx)^2+(sy(i,1)-smy)^2 > s0^2 then
        svx(i,1)=smx; svy(i,1)=smy
    end

    smx=mean([sx(i-1,ny),sx(i+1,ny),sx(i-1,ny-1),sx(i,ny-1),sx(i+1,ny-1)])
    smy=mean([sy(i-1,ny),sy(i+1,ny),sy(i-1,ny-1),sy(i,ny-1),sy(i+1,ny-1)])
    if (sx(i,ny)-smx)^2+(sy(i,ny)-smy)^2 > s0^2 then
        svx(i,ny)=smx; svy(i,ny)=smy
    end
end

for j=2:ny-1

    smx=mean([sx(1,j-1),sx(1,j+1),sx(2,j-1),sx(2,j),sx(2,j+1)])
    smy=mean([sy(1,j-1),sy(1,j+1),sy(2,j-1),sy(2,j),sy(2,j+1)])
    if (sx(1,j)-smx)^2+(sy(1,j)-smy)^2 > s0^2 then
        svx(1,j)=smx; svy(1,j)=smy
    end

    smx=mean([sx(nx,j-1),sx(nx,j+1),sx(nx-1,j-1),sx(nx-1,j),sx(nx-1,j+1)])
    smy=mean([sy(nx,j-1),sy(nx,j+1),sy(nx-1,j-1),sy(nx-1,j),sy(nx-1,j+1)])
    if (sx(nx,j)-smx)^2+(sy(nx,j)-smy)^2 > s0^2 then
        svx(nx,j)=smx; svy(nx,j)=smy
    end
end
end

```

A.4. Función subpixel.sce

Función desarrollada por Alex Liberzon, y Roi Gurka, denominada originalmente *uraPIV*, Modificada por Enrico Segre. Tiene por tarea interpolar, a través de una campana Gaussiana, los valores provenientes de la función de correlación, para encontrar la posición con mayor exactitud del máximo de esta función.

```
function [peakx,peaky]=subpixel(c)
    [iwx,iwy]=size(c);
    [cm,mi]=max(c)
    pixi=mi(1); pixj=mi(2);

    c(c<=0)=min(abs(c(abs(c)>0)));

    if pixi>1 & pixi<iwx & pixj>1 & pixj<iwy
        f0 = log(c(pixi,pixj));
        f1 = log(c(pixi-1,pixj));
        f2 = log(c(pixi+1,pixj));
        D=(2*f1-4*f0+2*f2);
        if D<>0 then peakx = pixi+ (f1-f2)/D; else peakx=pixi; end
        f0 = log(c(pixi,pixj));
        f1 = log(c(pixi,pixj-1));
        f2 = log(c(pixi,pixj+1));
        D=(2*f1-4*f0+2*f2);
        if D<>0 then peaky = pixj+ (f1-f2)/D; else peaky=pixj; end
    else
        peakx = pixi; peaky=pixj;
    end

    if ~isreal(peakx) then disp("complex peakx"); end
    if ~isreal(peaky) then disp("complex peaky"); end
    if isnan(peakx) then disp("peakx NaN"); end
    if isnan(peaky) then disp("peaky NaN"); end
endfunction
```

A.5. Función `interpola.sce`

La tarea de esta función, es redimensionar las celdas que componen las distintas matrices, ya sea para un campo vectorial, o para un campo escalar. Desarrollada por Francisco J. Martínez.

```
function [resu]=interpola(tott,neot);

tx=neot(1);
ty=neot(2);

[zx,zy]=size(tott);
resx=zx/neot(1);
zzx=[1:zx];
medio=zeros(tx,zy);

for j=1:zy
    zz=[tott(1:zx,j)];
    inex=smooth([zzx;zz'],resx);
    medio(1:tx,j)=inex(2,1:tx)';
end

resu=zeros(tx,ty);
resy=zy/neot(2)
zzy=[1:zy];

for j=1:tx
    zz=[medio(j,1:zy)];
    inex=smooth([zzy;zz],resy);
    resu(j,1:ty)=inex(2,1:ty);
end
```

A.6. Función `vorticidad.sce`

Calcula la vorticidad de un campo de velocidades utilizando derivadas de dos puntos. Desarrollada por Francisco J. Martínez.

```

function [vortex]=vorticidad2(sx,sy,dfx,dfy)

[mx,my]=size(sx);
hx=dfx/mx;hy=dfy/my;
catx=mx-1;
caty=my-1;
vortexmed=zeros(mx,my);

for dery=2:(caty)
    for derx=2:(catx)
        dvyx=(sy(derx+1,dery)-sy(derx-1,dery))/(2*hx);
        dvxy=(sx(derx,dery+1)-sx(derx,dery-1))/(2*hy);
        vortexmed(derx,dery)=(dvyx-dvxy);
    end
end

vortex=2*vortexmed(2:catx,2:caty);

```

A.7. Función magnitud.sce

Calcula la norma de cada vector de campo, y almacena la información en un campo escalar. Desarrollada por Francisco J. Martínez.

```

function [mag]=magnitud(sx,sy)

[mx,my]=size(sx)

for j=1:my
    for i=1:mx
        mag(i,j)=sqrt((sx(i,j)^2)+(sy(i,j)^2));
    end
end
end

```

A.8. Función `imprime.sce`

Imprime en un archivo de texto plano *.txt*, la información contenida en la matriz introducida. Se especifica el nombre del archivo y el lugar en la estructura de directorios. También imprime un archivo de texto con formato *.general*, para que los datos puedan ser leídos por **OpenDX**.

```
function [ok]=imprime(sz,nombre,num,campo)

clear fd;
clear fg;
[mx,my]=size(sz);
mm='%5.16f\t';

for j=1: my-1
    mm=mm+'%5.16f\t';
end

mm=mm+'\n';
fd=mopen(nombre+'.txt','w');
mfprintf(fd,mm,sz);
mclose(fd);

fg=mopen(nombre+'.general','w');
pp='file = /home/acustica/pivtext/'+campo+string(num)+'.txt\n';
mfprintf(fg,pp);
mfprintf(fg,'grid = %d x %d\n',mx,my);
mfprintf(fg,'format = ascii\n');
mfprintf(fg,'interleaving = record\n');
mfprintf(fg,'majority = column\n');
mfprintf(fg,'field = field0\n');
mfprintf(fg,'structure = scalar\n');
mfprintf(fg,'type = double\n');
mfprintf(fg,'dependency = positions\n');
mfprintf(fg,'positions = regular, regular, 0, 0.1, 0, 0.1\n');
mfprintf(fg,'\n');
```

```
mfprintf(fg,'end');
ok=1;
```

A.9. Función `imprime2d.sce`

Imprime en un archivo de texto plano *.txt*, la información contenida en las dos matrices introducidas, alternando columna a columna. Se especifica el nombre del archivo y el lugar en la estructura de directorios. También imprime un archivo de texto con formato *.general*, para que los datos puedan ser leídos por **OpenDX**, en este formato combinado.

```
function [okk]=imprime2d(ssx,ssy,nombre,num)

clear fd;
clear fg;
[mx,my]=size(ssx);

kxx=zeros(mx,2*my);

for i=1:my
    kxx(1:mx,(2*i-1))=ssx(1:mx,i);
    kxx(1:mx,(2*i))=ssy(1:mx,i);
end

mm='%5.16f\t';

for j=1: 2*my-1
    mm=mm+'%5.16f\t';
end

mm=mm+'\n';
fd=mopen(nombre+'.txt','w');
mfprintf(fd,mm,kxx);
mclose(fd);

//La siguiente parte crea el formato necesario para OpenDX
```

```
nombre=nombre+'.general'  
fg=mopen(nombre,'w')  
mfprintf(fg,'file = /home/acustica/pivtext/vectorsv%d.txt\n',num)  
mfprintf(fg,'grid = %d x %d\n',mx,my);  
mfprintf(fg,'format = ascii\n');  
mfprintf(fg,'interleaving = record-vector\n');  
mfprintf(fg,'majority = column\n');  
mfprintf(fg,'header = lines 0\n');  
mfprintf(fg,'field = velocity\n');  
mfprintf(fg,'structure = 2-vector\n');  
mfprintf(fg,'type = float\n');  
mfprintf(fg,'dependency = positions\n');  
mfprintf(fg,'positions = regular, regular, 0, 18.0, 0, 12.0\n');  
mfprintf(fg,'\n');  
mfprintf(fg,'end');  
mclose(fg);  
okk=1
```

Appendix B

Características del dispositivo experimental

Para poder realizar este trabajo el autor construyó primero un dispositivo experimental que tiene la característica, de poder controlar la frecuencia de oscilación para producir un movimiento sinoidal y cambiar la amplitud de oscilación de forma continua. El objetivo es caracterizar algunos difusores de acuerdo al flujo producido en función de la amplitud y la frecuencia. Sabemos que el agua al salir y entrar del difusor forma una estela de vórtices. Nuestro estudio está orientado hacia el entendimiento de este fenómeno, ya que para caracterizar los difusores es necesario conocer el patrón de flujo.

B.1. Partes principales

El dispositivo que se construyó en el **Laboratorio de Fluidos** para caracterizar los distintos perfiles con los que trabajamos, consta de dos partes principales, la primera es lo que llamamos **Generador de Movimiento** constituido por un motor de corriente directa conectado mediante un sistema de poleas como se observa en la figura B-2. Es un mecanismo que tiene la capacidad de modificar la distancia que hay entre el eje de rotación de la barra y el punto superior de sujeción de la biela (ver figura B-3). La biela se encarga de transmitir únicamente la componente vertical del movimiento de la polea. Esta variación en la distancia tiene un rango de 0.2 a 12.0 cm. La segunda parte es la que llamamos **Trompeta**. Está compuesta por una parte recta y un difusor; la biela, con su movimiento oscilante mueve al pistón dentro de la parte recta de la *trompeta*. La trompeta está parcialmente sumergida en un pequeño tanque (ver figura B-2) ¹ de agua. Los cambios de presión generados por el pistón hacen que el agua entre y salga del difusor, con lo cual tenemos un flujo oscilante, cuya amplitud y frecuencia están relacionadas directamente con las del pistón.

B.2. El generador de movimiento

Esta sección del dispositivo tiene a su vez dos partes. La primera es, como mencionamos anteriormente, un motor eléctrico, que opera con un voltaje de 24 V y alcanza una frecuencia de rotación de entre (0.2 - 2.0) Hz. Este motor es controlado por un regulador de voltaje al que se le puede variar de forma manual la intensidad de la potencia eléctrica, que a su

¹La capacidad del tanque es de alrededor de 30 litros



Figura B-1: Dispositivo experimental completo

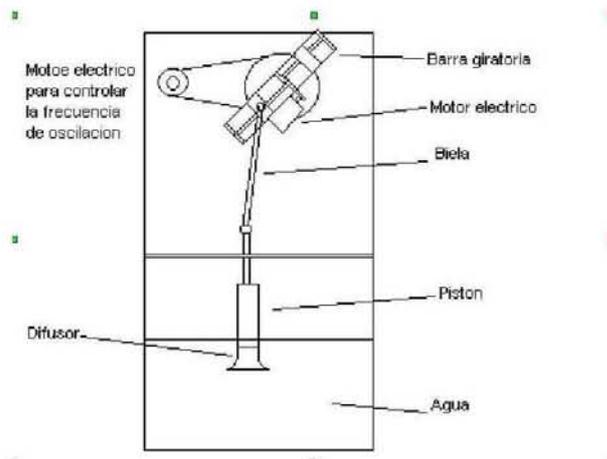


Figura B-2: Vista esquematica del dispositivo experimental

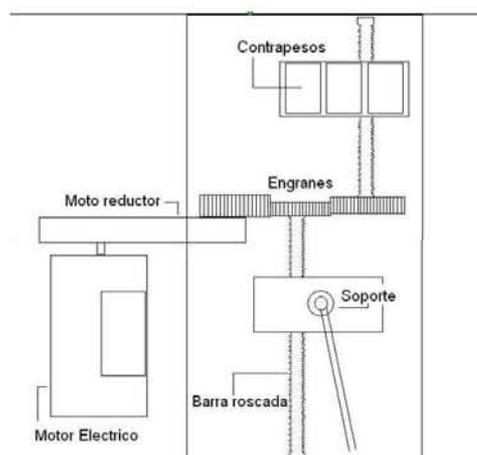


Figura B-3: Vista esquematica del dispositivo que varia la amplitud

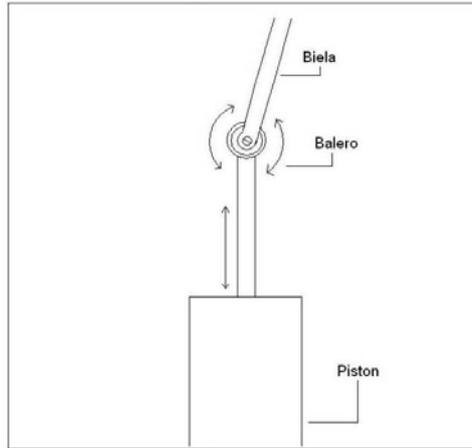


Figura B-4: Detalle de la unión entre la biela y el pistón

vez está relacionada con la frecuencia de rotación. El motor está conectado mediante una banda de hule a la segunda parte de esta sección. Este dispositivo es capaz de cambiar la distancia que hay entre el eje de rotación y un carro sobre el que está montado el soporte de la biela (ver figura B-3). El motor eléctrico que aparece en la misma figura, es también de corriente directa, pero su función es precisamente mover el carro mediante un sistema de engranes y un tornillo sin fin. Este segundo motor al encontrarse en un sistema que rota, es alimentado eléctricamente por un grupo de conmutadores o anillos y escobillas o carbones. Tiene la propiedad de ser bidireccional, lo cual nos permite acercar o alejar el carro. También aparece un segundo carro, que se mueve en sentido contrario al que sujeta la biela, transporta una serie de contrapesos para equilibrar el movimiento de todo el sistema, es decir para que la velocidad angular sea constante mientras no cambie el voltaje del motor principal. En el punto donde se sujeta la biela hay un balero que permite que rote este punto sin forzamientos. La biela a su vez, está conectada al pistón a través de un segundo balero (ver figura B-4) que permite minimizar esfuerzos y rozamientos en el pistón, además de garantizar que los únicos movimientos que se transmiten al pistón son los de tipo vertical.

B.3. Tipo de Oscilación

La oscilación que se desearía tener con un mecanismo como el descrito hasta este momento es de tipo senoidal. Sin embargo, esto no es del todo posible. Es necesario tomar en cuenta el efecto que tiene la biela entre el generador de movimiento y el pistón. Esto es debido a que la distancia entre el émbolo del pistón y el punto de sujeción de la biela no permanece constante. La razón está relacionada con el hecho de que la biela no es completamente vertical a lo largo de cada ciclo de oscilación. La única parte de la biela que contribuye al movimiento del pistón, es una proyección vertical de la misma. Como esta proyección sinusoidal cambia durante el ciclo, entonces la longitud vertical de la biela también cambia (ver figura B-6). Como el ángulo θ es distinto para cada configuración, es decir, si la longitud de la biela es h , la longitud neta x que presenta al pistón es $h \cos \theta$. Si ϕ es el ángulo de giro de la polea que sostiene la biela respecto al eje llamado D , entonces, el ángulo θ queda definido de acuerdo a el siguiente desarrollo.

$$\tan \theta = \frac{\sin \theta}{\cos \theta} = \frac{L \sin \phi}{h \cos \theta}$$

De donde

$$L \sin \phi = h \cos \theta \tan \theta = h \sin \theta$$

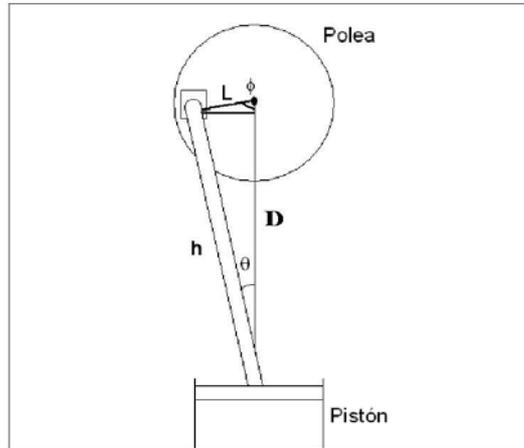


Figura B-5: Se muestra la ubicación de los distintos parámetros que definen el tipo de oscilación

$$\theta = \sin^{-1}\left(\frac{L}{h} \sin \phi\right)$$

Si $\beta = \pi - \alpha$

$$\sin \theta = \frac{L}{h} \sin \beta$$

De esta forma el parámetro que ahora se define como Δx , incorpora una oscilación adicional a la producida simplemente por el movimiento giratorio de la polea. La oscilación adicional no es de la magnitud de x , sino, la variación en x , que esta dada por:

$$\Delta x = h - h \cos \theta = h \cos(\sin^{-1}\left(\frac{L}{h} \sin \phi\right))$$

La oscilación total se expresa como $y = A + x$, donde $A = L \sin \phi$, de manera explícita:

$$y = L \sin \phi + h(1 - \cos(\sin^{-1}\left(\frac{L}{h} \sin \phi\right)))$$

Las gráficas de la función y junto con las de la función A , son diferentes (ver figura B-6). En color negro aparece la curva correspondiente a y , y en color gris la correspondiente a A . Se observa que la función y , es ligeramente mayor que A en cada punto como se esperaba y coinciden en los nodos. En realidad se puede demostrar que si tomamos el límite

$$\lim_{\frac{L}{h} \rightarrow 0} y \rightarrow A$$

Basta con demostrar que;

$$\lim_{t \rightarrow 0} \sin^{-1}(t) \rightarrow 0$$

ya que el

$$\lim_{t \rightarrow 0} \cos(t) \rightarrow 1$$

Se puede destacar que si el tamaño de la biela h es mucho mayor que la distancia L , entonces, el movimiento oscilatorio queda descrito aceptablemente por una función

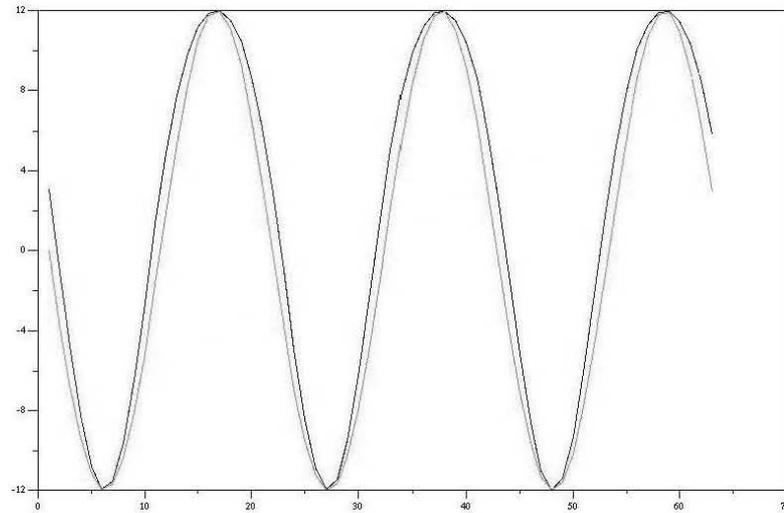


Figura B-6: La curva y en puntos de color negro, la curva A en puntos de color gris. En el eje vertical se muestra la amplitud en cm , en el horizontal aparece el tiempo en unidades de $1/30$ segundos

sinusoidal. En este caso $h = 32,0$ cm y $L \leq 12,0$ cm. Estos parámetros se utilizaron para construir la gráfica que aparece en la figura B-6. Se encontró que la diferencia máxima porcentual entre la función y , con la función A es del orden de $3,17\%$, lo que en términos prácticos no es demasiado, aunque es importante no olvidarlo.

B.4. Control del sistema

El dispositivo cuenta con dos motores, independientes. El primero encargado de cambiar la frecuencia de oscilación, como se muestra en la figura B-2, y el segundo de cambiar la distancia entre el eje de rotación y el punto de sujeción de la biela sobre la polea, como se muestra en la figura B-3. En cálculos anteriores a este parámetro se le llamó L . Sin embargo, el usuario tiene la posibilidad de combinar el funcionamiento de ambos. El control es básicamente un par de interruptores y un controlador de potencia, uno de los interruptores también funciona como inversor de polaridad. El interruptor inversor es finalmente el que se encarga de controlar la amplitud de oscilación es decir L . Tiene tres estados, uno es apagado, los otros dos son encendidos, pero con polaridades invertidas una respecto de la otra. Este interruptor está conectado a un arreglo de cuatro conmutadores mediante el mismo número de escobillas, se observa con detalle en la figura B-3. En el caso del control para variar la potencia eléctrica, que es la parte que se encarga de alimentar el motor principal, tenemos un rango de variación tal, que las frecuencias de rotación van desde los $(0.2 - 2.0)$ Hz. La conexión entre el motor y la fuente eléctrica es directa como se muestra en la figura B-7. Se cuenta entonces con un sencillo mecanismo para muestrear el espacio de parámetros, es decir, el espacio de amplitudes y frecuencias. En estos momentos la forma de medir cada uno de los parámetros es manual. Esto significa, que tanto la amplitud como la frecuencia se miden de manera externa y ayudados por reglas escaladas, en un caso, y en otro midiendo el período con un cronómetro. Desde luego esto implica serias limitaciones como por ejemplo la resolución con la que se pueden hacer los desplazamientos sobre el espacio experimental. Como trabajo futuro, se instalarán sensores electrónicos apropiados, tanto para la amplitud como para la frecuencia. La información recolectada por estos dispositivos será enviada

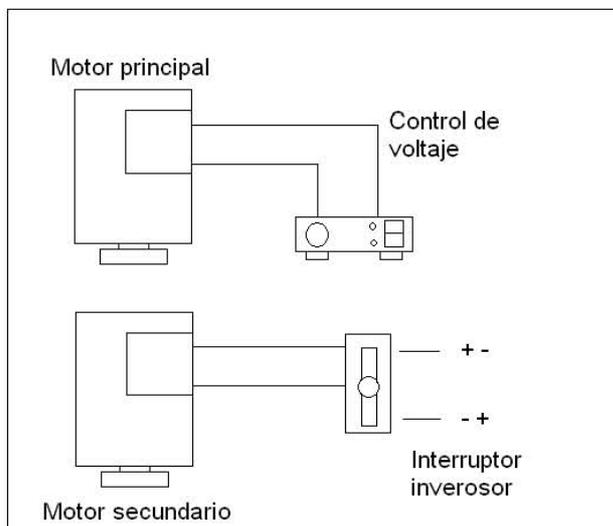


Figura B-7: Diagrama de la conexión eléctrica para los motores del dispositivo.

directamente a una computadora la cual tendrá también el control, de cada uno de los parámetros.