



ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

DIPARTIMENTO DI INGEGNERIA INDUSTRIALE (DIN)

CORSO DI LAUREA IN INGEGNERIA MECCANICA

Analisi fluidodinamica di un levitatore acustico con Particle Image Velocimetry

Tesi di laurea in Fisica Tecnica

Relatore

Prof. Massimiliano Rossi

Presentata da

Chiara Minervino

Sessione ottobre 2025
Anno Accademico 2024/2025

Abstract

Questa tesi tratta la caratterizzazione sperimentale e l'analisi fluidodinamica di TinyLev, un levitatore acustico ad asse singolo non risonante, dotato di due emettitori ultrasonici operanti a 40 kHz. La presenza di due onde acustiche viaggianti in senso opposto permette la formazione di un'onda stazionaria, nei cui nodi è possibile levitare stabilmente piccoli oggetti.

In seguito ad una prima analisi del segnale, si è determinato come ampiezza e frequenza, regolabili con l'oscilloscopio digitale *Digilent Analog Discovery 3*, impattano sul funzionamento del levitatore e sull'intensità del campo acustico, che cala al diminuire della prima. È stata eseguita una mappatura sperimentale del campo acustico in diversi punti di controllo, al variare del voltaggio, confrontando i risultati con quelli ottenuti utilizzando il segnale prodotto da una scheda Arduino. Un'analisi spettrale ha, poi, permesso di valutare la ripetibilità del segnale generato dall'oscilloscopio.

È stato, infine, realizzato un setup sperimentale per condurre un'analisi PIV (*Particle Image Velocimetry*) con l'obiettivo di caratterizzare i campi di velocità associati allo streaming acustico relativi alla levitazione di una sfera di polistirolo e di una goccia d'acqua. Le immagini acquisite sono state analizzate inizialmente tramite il software PIVlab e successivamente rielaborate con MATLAB.

I risultati ottenuti evidenziano la stretta dipendenza del flusso dalla geometria e dalle caratteristiche fisiche della superficie degli oggetti levitati. Nel caso della goccia d'acqua, infatti, il flusso ne segue la superficie sferica, curvandosi attorno ad essa, fenomeno non osservato nel caso del polistirolo, probabilmente dovuto alla differente geometria dei due oggetti, il secondo non è perfettamente sferico, e alle diverse proprietà fisiche, come la scabrezza della superficie. Questi studi forniscono, dunque, interessanti spunti per ulteriori sviluppi nell'ambito della levitazione acustica.

Indice

1	Introduzione	4
1.1	Tecnologie di levitazione	4
1.2	Levitazione acustica	4
1.2.1	Influenza della temperatura sulla lunghezza d'onda	6
1.2.2	Forza di radiazione acustica	7
2	Caratterizzazione del levitatore acustico	9
2.1	Levitatore acustico	9
2.2	Setup sperimentale	11
2.2.1	Analisi della trasmissione del segnale	11
2.2.2	Effetto della frequenza del segnale sulla levitazione	12
2.2.3	Variazione dell'ampiezza	13
2.3	Risultati	14
2.3.1	Analisi del segnale	14
2.3.2	Variazione della frequenza	15
2.3.3	Variazione dell'ampiezza	15
3	Analisi acustico-fluidodinamica	18
3.1	Metodologia	18
3.1.1	Particle Image Velocimetry	18
3.1.2	Seeding	20
3.1.3	Ottica e fenomeni di diffrazione	22
3.2	Realizzazione del setup ottico	24
3.2.1	Ottica	25

3.2.2	Realizzazione del box	26
3.2.3	Generatore di fumo e high speed camera	27
3.3	Risultati	27
3.3.1	Impostazioni PIVlab	28
3.3.2	Post-processing con MATLAB	30
4	Discussione	45
4.1	Sfera di polistirolo	45
4.2	Goccia d'acqua	46
4.3	Osservazioni	47
5	Conclusioni	49
A	Codici Matlab	52
A.1	Analisi comparativa	52
A.2	Analisi spettrale	54
A.3	Rielaborazione PIV, sfera di polistirolo	57
A.3.1	Colormaps con vettori	57
A.3.2	Colormaps senza vettori	60
A.4	Rielaborazione PIV, goccia d'acqua	63
A.4.1	Colormaps con vettori	63
A.4.2	Colormaps senza vettori	66
B	Grafici	69
B.1	Grafici analisi comparativa	69

Capitolo 1

Introduzione

1.1 Tecnologie di levitazione

Nel corso del tempo sono state sviluppate diverse tecnologie che permettono di levitare piccole particelle.

La levitazione ottica, ad esempio, sfrutta il trasferimento di quantità di moto da parte dei fotoni, che si verifica quando la particella da sospendere viene colpita da un fascio di luce laser. Questo principio porta alla levitazione di particelle molto piccole, dell'ordine di $0.01 - 10 \mu\text{m}$.

La levitazione magnetica, invece, permette di sospendere oggetti in materiale ferromagnetico e diamagnetico. Nel secondo caso, tuttavia, dato il debole diamagnetismo della maggior parte dei materiali di interesse, si rendono necessari campi magnetici molto forti.

Esistono, inoltre, la levitazione aerodinamica e quella elettrostatica. Entrambe presentano dei limiti, nella prima dovuti ad una possibile agitazione e alterazione dell'oggetto sospeso, nella seconda a causa della complessità dei sistemi di controllo necessari al suo funzionamento.

La levitazione acustica, argomento di questa tesi, sarà trattata in maniera approfondita nel paragrafo seguente.

1.2 Levitazione acustica

La levitazione acustica è un fenomeno fisico che sfrutta le onde sonore, che sono onde meccaniche e quindi trasportano quantità di moto, per sospendere piccoli oggetti in aria. Ciò avviene

grazie alla presenza di forze acustiche che agiscono in tutte le direzioni, mantenendone stabile la posizione.

Trova applicazione in diversi campi, quali la farmaceutica, il trasporto di oggetti senza la necessità di utilizzare un contenitore, lo studio dei cambiamenti di fase, dell'evaporazione di liquidi binari, della formazione di cristalli eutettici nei metalli fusi e della rapida cristallizzazione o ionizzazione dei materiali in levitazione. In generale le principali applicazioni trovano riscontro nelle ricerche su biomateriali e nella chimica.

Esistono due principali categorie di levitatori acustici ad asse singolo[2]:

- non risonanti a due emettitori

Due emettitori distinti generano delle onde acustiche in fase che viaggiano in direzione opposta, nel caso del TinyLev ad una frequenza pari a 40 kHz, e questo porta alla formazione di un'onda stazionaria, caratterizzata dalla presenza di nodi e di antinodi. In prossimità dei nodi è possibile mettere dei piccoli oggetti in sospensione.

Questo tipo di levitatori risulta essere meno efficiente rispetto a quelli risonanti, ma è anche meno sensibile alle variazioni delle condizioni ambientali.

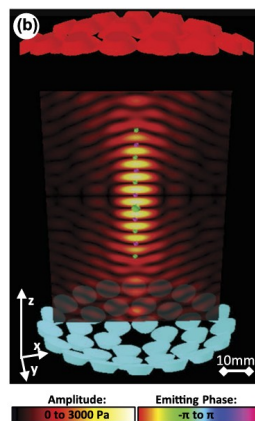


Figura 1.1: Simulazione del campo acustico generato dal levitatore non risonante TinyLev[2]

- risonanti ad emettitore singolo

In questa categoria è possibile trovare, ad esempio, i levitatori Langevin Horn. In generale funzionano grazie alla generazione di onde ad alta frequenza da parte di un trasduttore ultrasonico, che si riflettono su un riflettore, posto ad una distanza dal trasduttore che

sia un multiplo intero di metà della lunghezza d'onda[3], portando così alla formazione di un'onda stazionaria.

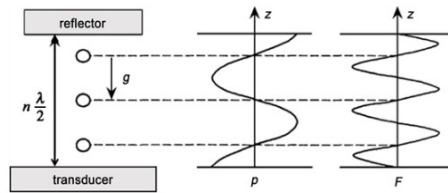


Figura 1.2: Schema di un levitatore risonante[3]

La necessità di una distanza così precisa, tuttavia, li rende particolarmente suscettibili a variazioni delle condizioni ambientali, essendo la velocità del suono, e quindi la lunghezza d'onda, dipendente dalla temperatura. Inoltre, a causa della presenza di inefficienze nel sistema, i trasduttori sono soggetti al surriscaldamento, che causa inevitabili variazioni di temperatura.

Il funzionamento ad elevati voltaggi, nell'ordine di 100 – 1000 V, in ogni caso, permette la generazione di forti campi di pressione acustica, nonostante la presenza di un singolo emettitore.

1.2.1 Influenza della temperatura sulla lunghezza d'onda

Come affermato precedentemente, c'è una stretta correlazione tra la temperatura ambientale e la lunghezza d'onda acustica nell'aria, essendone la velocità del suono direttamente dipendente.

T (°C)	c (m/s)	λ (mm)
+15	340.5	8.513
+20	343.4	8.585
+25	346.3	8.658
+30	349.2	8.730

Tabella 1.1: c e λ in funzione della temperatura

La velocità del suono si ricava dalla formula:

$$c = \sqrt{\frac{\gamma \cdot R_0 \cdot T}{M_a}} \quad (1.1)$$

In cui:

$$\gamma = \frac{c_p}{c_v} = 1.4$$

$$R_0 = 8.314 \frac{J}{K_{mol}} \text{ costante assoluta dei gas}$$

T temperatura assoluta in Kelvin

$$M_a = 28.96 \text{ g/mol peso molecolare dell'aria}$$

La lunghezza d'onda è, invece, ricavabile nel seguente modo:

$$\lambda = \frac{c}{f} \quad (1.2)$$

con f =frequenza

Considerando un range di temperature differenti, è possibile osservare nella tabella 1.1 come variano la velocità del suono e la lunghezza d'onda.

Conoscendo la lunghezza d'onda è possibile, inoltre, ricavare la distanza tra i nodi dell'onda stazionaria, questa vale infatti:

$$d = \frac{\lambda}{2} \quad (1.3)$$

1.2.2 Forza di radiazione acustica

La forza di radiazione acustica che agisce su un piccolo oggetto di forma sferica posto in levitazione può essere calcolata tramite il potenziale di Gor'Kov[4]:

$$U_G(\mathbf{r}) = \frac{4\pi a^3}{3} \left[\frac{1}{2} f_1 \frac{\langle p^2(\mathbf{r}) \rangle}{\rho_0 c_0^2} - \frac{3}{4} f_2 \rho_0 \langle v^2(\mathbf{r}) \rangle \right] \quad (1.4)$$

In cui:

a è il raggio della sfera

$\langle p^2 \rangle$ è il valore quadratico medio della pressione acustica (Pa^2)

$\langle v^2 \rangle$ è il valore quadratico medio della velocità del fluido (m/s)²

$$f_1 = 1 - \frac{K_p}{K_0}$$

$$f_2 = \frac{2(\rho_p - \rho_0)}{2\rho_p + \rho_0}$$

K_p e K_0 rappresentano la comprimibilità della particella e del fluido

ρ_p e ρ_0 sono le rispettive densità.
 c_0 è la velocità del suono nel fluido

Questa teoria è valida solo se è verificata la condizione del regime di Rayleigh:

$$\frac{a}{\lambda} \ll 1 \quad (1.5)$$

Ovvero se la particella sospesa ha dimensioni di molto inferiori rispetto alla lunghezza d'onda.
Inoltre devono essere verificate le condizioni di:

- fluido ideale
- onde acustiche lineari
- particella rigida e sferica
- campo acustico stazionario
- assenza di effetti di streaming

Infine, la forza di radiazione acustica è calcolabile come:

$$F_{ac} = -\nabla U_G \quad (1.6)$$

Capitolo 2

Caratterizzazione del levitatore acustico

2.1 Levitatore acustico

Oggetto di questo capitolo è lo studio e la caratterizzazione del levitatore acustico non risonante a due emettitori TinyLev [2].

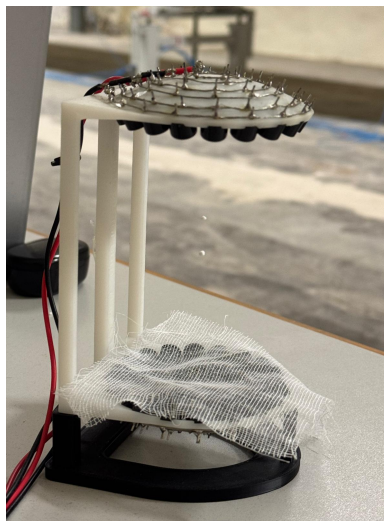


Figura 2.1: Levitazione di sfere di polistirolo con TinyLev

Come già detto nel Capitolo 1, essendo costituito da due emettitori distinti, garantisce una maggiore indipendenza dalle condizioni ambientali e in particolare funziona in un range di tem-

peratura che varia tra -40 e $+40$ °C.

I due emettitori sono costituiti da 36 speakers *Murata* [5] ciascuno, per un totale di 72 trasduttori.

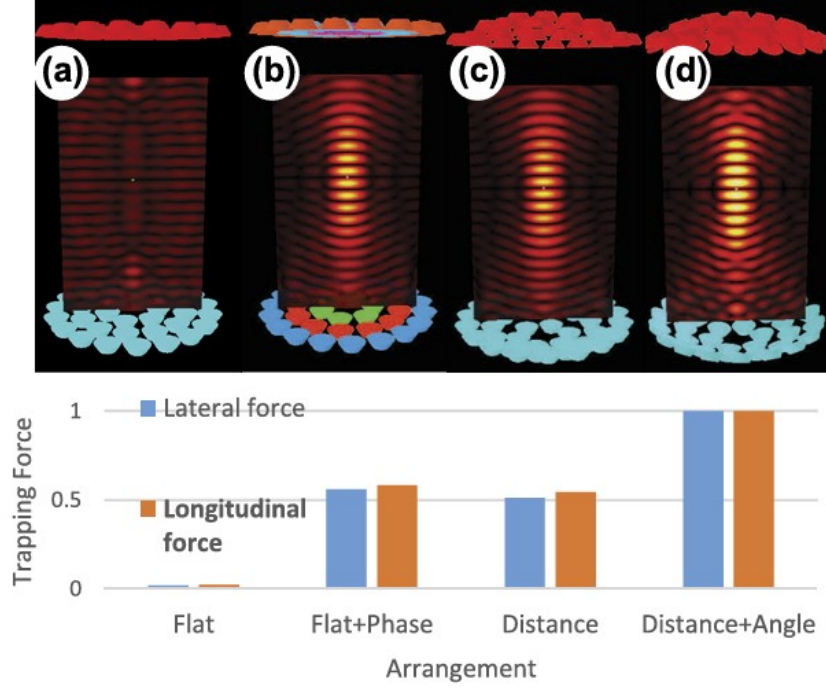


Figura 2.2: Variazioni della forza laterale e longitudinale dovute a diverse possibili configurazioni del TinyLev [2]

La configurazione degli speakers è stata progettata per concentrare la radiazione acustica nel centro del levitatore, effetto supportato anche dalla sincronizzazione della fase dei due segnali emessi. Come già affermato, la presenza di due onde acustiche che viaggiano in direzioni opposte permette la formazione di un'onda stazionaria e la levitazione è possibile in prossimità dei suoi nodi.

Il segnale elettrico, costituito da due onde quadre caratterizzate da un'ampiezza di 5 V e una frequenza di 40 kHz, viene generato da una scheda Arduino Nano ed è trasmesso ad un amplificatore L297N *Dual H-Bridge*, collegato ai due emettitori.

Facendo riferimento alla (1.4), essendo in condizioni del regime di Rayleigh, ovvero considerando la levitazione di particelle di dimensione inferiore a $\frac{\lambda}{2}$, si nota la dipendenza del

potenziale di Gor’Kov e quindi della forza di radiazione acustica dalla densità della particella stessa. TinyLev è in grado di sospendere oggetti non più densi di $2,2 \text{ g/cm}^3$. Ciò è dovuto all’impossibilità di modificare il voltaggio del segnale. È stato, infatti, dimostrato che oggetti caratterizzati da una densità maggiore possono essere levitati, ma richiedono un incremento del voltaggio.

2.2 Setup sperimentale

2.2.1 Analisi della trasmissione del segnale

Per comprenderne al meglio il funzionamento e valutare, come si vedrà successivamente, una possibile modifica del segnale al fine di variare alcuni parametri caratterizzanti dell’onda, quali ampiezza e frequenza, e valutarne gli effetti sul funzionamento del levitatore, si è deciso di procedere con uno studio di caratterizzazione del segnale del TinyLev.

Come accennato nel paragrafo 2.1, il segnale del levitatore viene generato da una scheda Arduino, collegata agli ingressi IN1, IN2, IN3, IN4 di un amplificatore di segnale L297N *Dual H-Bridge*.

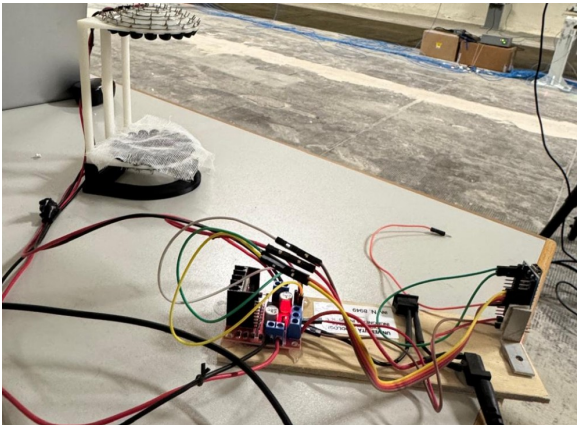


Figura 2.3: Collegamento tra scheda Arduino e amplificatore

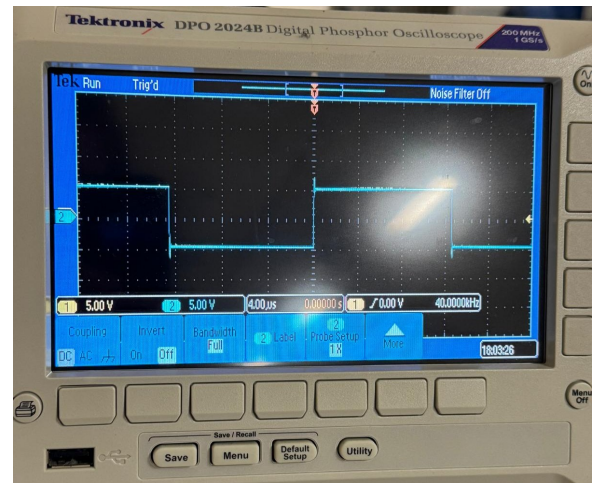


Figura 2.4: Acquisizione del segnale

Per condurre tale studio, si è deciso di acquisire il segnale tramite l’impiego dell’oscilloscopio digitale *Digilent Analog Discovery 3*[8]. Affinché ciò fosse possibile, si è reso necessario

modificare parzialmente i collegamenti dei cavi del TinyLev (fig 2.3), in modo da potervi connettere le due sonde dell'oscilloscopio. Il segnale è, infine, stato acquisito grazie all'impiego del software *WaveForms* [9], che permette di registrare l'andamento di due segnali in ingresso, e successivamente verificato con l'utilizzo di un oscilloscopio analogico (fig.2.4).

2.2.2 Effetto della frequenza del segnale sulla levitazione

Come si è visto nel paragrafo 1.2.1, la frequenza del segnale ha una forte influenza sulla lunghezza d'onda e quindi sulla distanza tra i nodi dell'onda stazionaria.

Per poter modificare la frequenza del segnale, è stato necessario scollegare l'amplificatore dalla scheda Arduino e collegarlo all'oscilloscopio digitale, in grado anche di funzionare come generatore di onde grazie all'estensione *WaveGen* del software *WaveForms*.

Dopo aver ipotizzato diverse possibili configurazioni, si è deciso di scollegare i quattro cavi relativi agli input IN1, IN2, IN3, IN4 che connettevano l'amplificatore con la scheda Arduino e di collegarli direttamente alle due sonde dell'oscilloscopio (fig. 2.5), escludendo, dunque, la scheda Arduino dalla generazione del segnale. Le due sonde sono state collegate agli output W1 e W2 del *Digilent*.

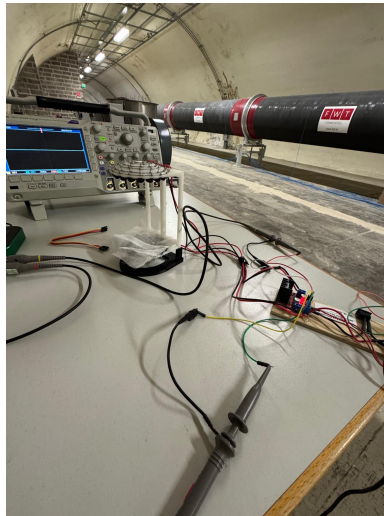


Figura 2.5: Levitatore in funzionamento, segnale in input da *Digilent*

Utilizzando *WaveGen* è possibile generare onde di diverse forme, ad esempio sinusoidali, quadre, etc. In questo caso, si è deciso, per coerenza, di impiegare la stessa forma d'onda del

segnale generato dalla scheda Arduino, ovvero l'onda quadra.

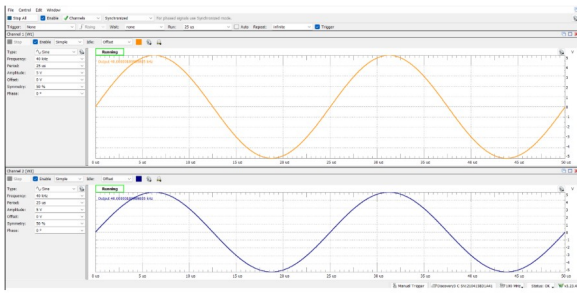


Figura 2.6: Segnali sinusoidali sincronizzati da *WaveForms*

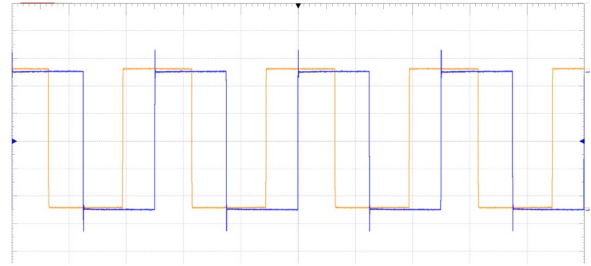


Figura 2.7: Confronto dei segnali provenienti dall'*Analog Discovery 3* (in blu) e dalla scheda Arduino (in giallo)

L'impiego di oscilloscopio analogico ha, infine, permesso di valutare l'accuratezza delle acquisizioni precedentemente eseguite e ha dato la possibilità di verificare che il segnale emesso dal *Digilent* coincidesse con quello generato dalla scheda Arduino.

2.2.3 Variazione dell'ampiezza

Sfruttando la configurazione precedentemente realizzata al fine di variare la frequenza del segnale, sono state eseguite delle prove modificandone l'ampiezza.

Setup di acquisizione dati

Per comprendere al meglio gli effetti dovuti alla variazione di ampiezza del segnale sul campo acustico, sono state effettuate cinque acquisizioni dati per ciascun valore (5 V, 4 V e 2 V), ognuna in una posizione differente del campo acustico: posizione centrale, centrale superiore e centrale inferiore, laterale destra e laterale sinistra.

Per garantire l'accuratezza delle acquisizioni è stato necessario realizzare un setup che permettesse di posizionare con precisione la sonda dell'oscilloscopio (figg. 2.8 e 2.9). A quest'ultima è stato connesso uno speaker, che ha funzionato da trasduttore tra segnale acustico e segnale elettrico.

Prima di procedere con le misurazioni, utilizzando la sonda con lo speaker si è verificato il corretto funzionamento di tutti gli speakers del levitatore.

In questa fase l'*Analog Discovery* è stato utilizzato sia per acquisire i dati che per generare il segnale in input all'amplificatore.

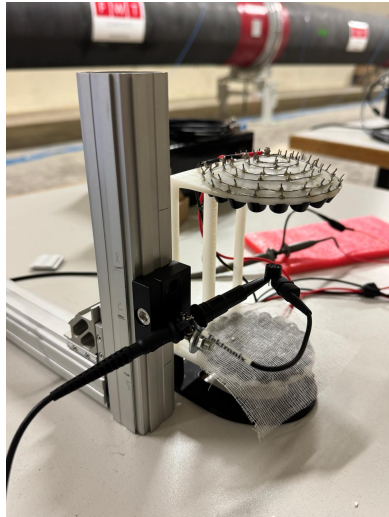


Figura 2.8: Vista laterale del setup

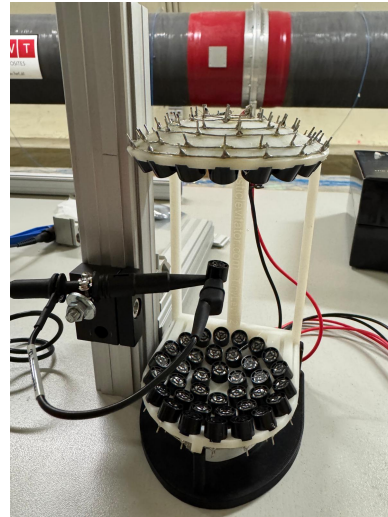


Figura 2.9: Vista frontale del setup

Per verificare la costanza dei dati acquisiti, inoltre, sono state effettuate, utilizzando la stessa configurazione, 10 misurazioni ad ampiezza costante pari a 5 V, mantenendo la sonda nel centro del campo acustico, con generazione del segnale da parte del *WaveGen*.

2.3 Risultati

2.3.1 Analisi del segnale

La caratterizzazione del segnale effettuata tramite l'impiego dell'oscilloscopio, ha permesso di comprendere che:

- La scheda Arduino genera due distinti segnali sotto forma di onda quadra, con ampiezza di 5V e frequenza di 40 kHz, trasmessi all'amplificatore rispettivamente dai cavi collegati ad IN1-IN2 e IN3-IN4.
- L'amplificatore trasmette uno dei due segnali al gruppo di speaker superiori e l'altro a quelli inferiori.
- Gli speakers fungono da trasduttori, trasformano il segnale in ingresso sotto forma di onda quadra in un'onda acustica con frequenza pari a 40 kHz .

2.3.2 Variazione della frequenza

In tabella 2.1 è possibile osservare, considerando una temperatura ambientale costante di 20 °C e dunque una velocità del suono in aria $c=343$ m/s, i valori assunti dalla lunghezza d'onda al variare della frequenza e di conseguenza la relativa distanza tra i nodi.

f (kHz)	λ (mm)	d (mm)
10	34.34	17.17
20	17.17	8.59
30	11.45	5.72
40	8.59	4.30
50	6.87	3.44

Tabella 2.1: Distanza tra i nodi al variare della frequenza

I valori presenti nella tabella 2.1, tuttavia, sono puramente teorici.

Infatti, come riportato nella scheda tecnica degli speakers *Murata* [5], questi funzionano in maniera ottimale solamente nell'intorno dei 40 kHz di frequenza, pertanto i tentativi di modificarla non hanno condotto a validi risultati.

2.3.3 Variazione dell'ampiezza

L'intensità del campo acustico presenta una forte dipendenza dal voltaggio del segnale. Effettuando diversi tentativi di levitazione di una stessa sfera di polistirolo ad ampiezze d'onda decrescenti, si è infatti notato che:

- A 5 V si ha il funzionamento ottimale
- A 4 V la particella levita, sebbene il campo acustico appaia meno intenso
- A 2 V la particella non levita

Tale effetto si verifica conseguentemente alla diminuzione dell'intensità del campo acustico, dovuta ad una minore ampiezza del segnale in input. In ogni caso, come affermato nel paragrafo 2.1, campi acustici meno intensi possono comunque levitare particelle caratterizzate da una minore densità.

Analisi comparativa

Dopo aver mappato il campo acustico in 5 punti di controllo per ogni voltaggio del segnale, si è deciso di confrontarne l'andamento. La prima rilevazione dati è stata effettuata mantenendo collegata la scheda Arduino, per avere un campo acustico di confronto con quelli generati dal segnale del *Digilent*.

Il codice MATLAB (Appendice A.1) sviluppato ha permesso di ottenere una rappresentazione grafica del segnale acustico rilevato in ogni punto di controllo, sovrapposta al segnale di confronto (Appendice B.1).

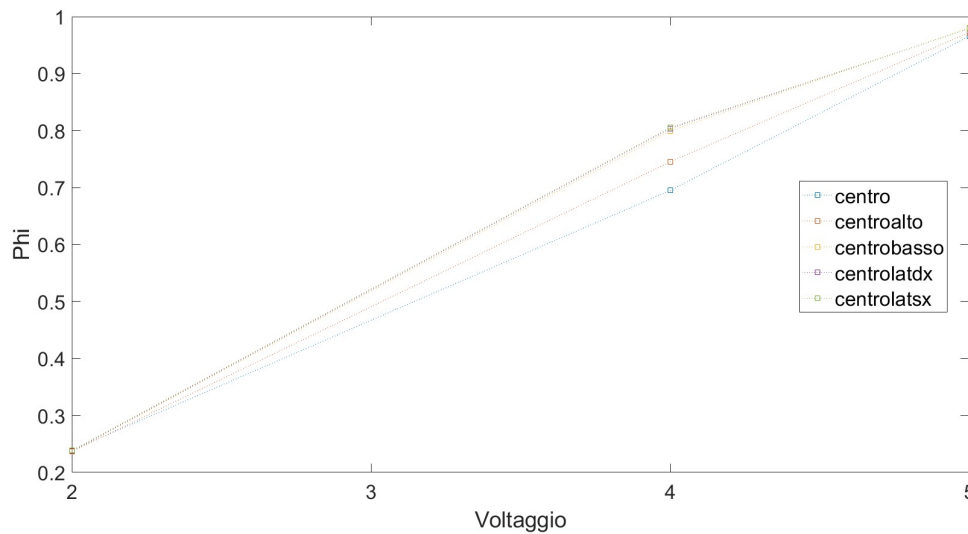


Figura 2.10: Rappresentazione dati adimensionalizzati

Per avere una rappresentazione di più immediata comprensione, i dati sono stati adimensionalizzati, rapportando il voltaggio picco-picco del segnale Arduino con quello del segnale dell'oscilloscopio tramite la variabile ϕ (fig.2.10).

È immediato notare, innanzitutto, che il segnale più intenso, relativamente a ciascuno dei voltaggi, è quello rilevato nella zona centrale. Ciò risulta essere coerente con la geometria del levitatore, progettato per concentrare al suo centro la forza di radiazione acustica. Inoltre, facendo riferimento alla fig. 2.10, è possibile osservare che, come ci si aspetterebbe, nel caso di ampiezza del segnale pari a 5 V, i segnali acquisiti in ognuna delle posizioni di riferimento hanno valori molto vicini a quelli di riferimento.

Infine, la figura 2.10 evidenzia chiaramente la diminuzione dell'intensità del campo acustico al ridursi dell'ampiezza del segnale.

Analisi spettrale del segnale

Per valutare la costanza del segnale generato dall'oscilloscopio digitale, è stata eseguita un'analisi spettrale sui 10 segnali ad ampiezza costante pari a 5 V acquisiti nel centro del campo acustico. (Appendice A.2).

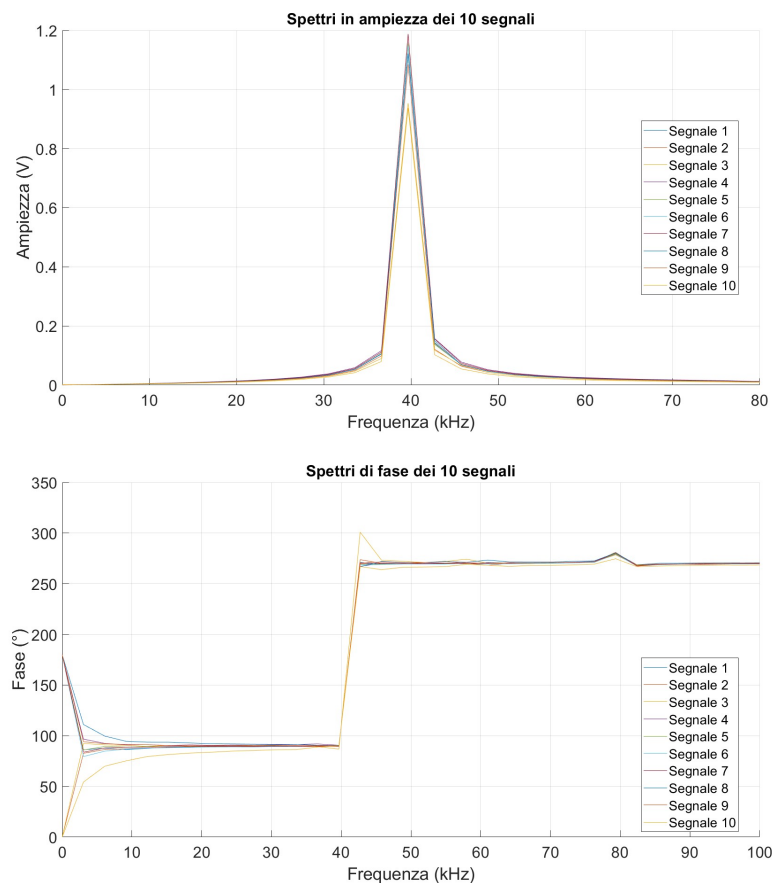


Figura 2.11: Spettri di ampiezza e di fase dei segnali comparativi

Osservando la fig. 2.11, si nota come i segnali siano sostanzialmente identici, presentando solamente leggere oscillazioni nell'ampiezza.

Capitolo 3

Analisi acustico-fluidodinamica

3.1 Metodologia

3.1.1 Particle Image Velocimetry

La *Particle Image Velocimetry* (PIV) è un metodo di misura indiretta dei campi di velocità di un fluido, che sfrutta la presenza di particelle traccianti aggiunte al flusso, rese visibili ad una telecamera ad alta velocità grazie alla presenza di un piano di luce laser e allo sfruttamento del fenomeno di diffusione della luce da parte delle stesse.

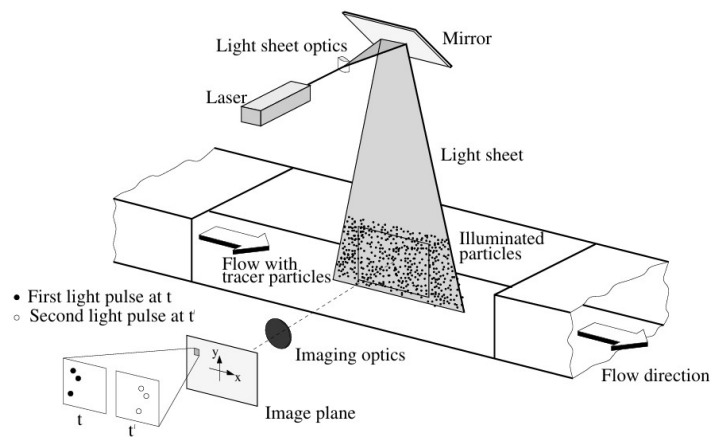


Figura 3.1: Rappresentazione schematica di un possibile setup per un'analisi PIV [1]

In figura 3.1 sono rappresentati gli elementi fondamentali di un setup PIV.

Un laser genera un fascio di luce collimata ad alta densità di energia, che, grazie all'impiego di lenti cilindriche e/o sferiche, assume la forma di un piano, che a sua volta può essere direzionato con degli specchi.

Al flusso da analizzare viene aggiunto un *seeding*, costituito da particelle traccianti caratterizzate da una densità simile a quella del fluido stesso (par. 3.1.2), che permette di osservarne lo spostamento e quindi di calcolare i vettori del campo di velocità.

Per ottenere dei risultati accurati si impiega una media densità di particelle traccianti, distribuite in maniera omogenea. Per campi ad alta densità di *seeding* si applica la *Laser Speckle Velocimetry* (LSV), per campi a bassa densità la *Particle Tracking Velocimetry* (PTV).

La durata dell'impulso di illuminazione del laser deve essere sufficientemente breve da "congelare" il moto delle particelle ed evitare così la formazione di scie durante l'esposizione.

Il lasso di tempo che intercorre tra due impulsi di illuminazione, inoltre, deve essere al contempo abbastanza lungo da permettere di determinare con una buona risoluzione lo spostamento delle particelle traccianti e abbastanza breve da evitare che particelle caratterizzate da una componente di velocità fuori dal piano escano da questo nel frattempo.

La presenza di una *high speed camera* permette, infine, di registrare immagini ad elevata frequenza.

Il campo di interesse viene suddiviso in aree di interrogazione e, per ognuna di esse, viene confrontata la posizione delle particelle in due frame consecutivi. I vettori di spostamento locale vengono derivati tramite metodi statistici, quali auto- e cross-correlazione.

La dimensione di queste aree deve essere sufficientemente piccola da ridurre l'influenza dei gradienti di velocità sui risultati.

La loro dimensione, inoltre, determina il numero di vettori di velocità indipendenti e dunque la massima risoluzione spaziale del campo di velocità, vincolata in ogni caso alla risoluzione del sensore impiegato per la registrazione.

Oggi, grazie all'impiego di dispositivi CCD (*Charged Couple Device*) è possibile ottenere più di 100 registrazioni al minuto. Inoltre, la registrazione ad alta velocità su sensori CMOS (*Complementary Metal-Oxide Semiconductor*) consente persino l'acquisizione a frequenze nell'ordine dei kHz.

Tra i vantaggi che questa tecnologia presenta, sicuramente si annoverano la sua capacità di registrare immagini che includono ampie porzioni del campo di interesse e la sua versatilità di applicazione, in mezzi liquidi e gassosi.

Inoltre, è una tecnica non intrusiva, dunque permette di analizzare anche flussi ad alta velocità e *boundary layers* in prossimità della parete.

Software PIVLab

Una volta registrate le immagini, queste necessitano di una fase di pre e post-processing ed elaborazione. Il software impiegato per questa tesi è un *toolbox* di MATLAB, chiamato PIVLab [7]. Permette di analizzare singole immagini o frame multipli e supporta sia analisi PIV 2D classiche che analisi più avanzate, come *multipass*, *image deformation*, *sub-pixel estimation*.

I risultati ottenuti possono essere visualizzati tramite mappe vettoriali e grafici e i dati possono essere esportati.

Le immagini importate devono, inizialmente, essere sottoposte ad una fase di pre-processing, in cui, grazie all'impiego di vari filtri, le particelle di *seeding* sono rese ben visibili. Viene, poi, scelta la dimensione delle finestre di interrogazione, parametro che definisce la risoluzione del campo vettoriale.

È possibile, inoltre, effettuare una calibrazione per passare ad unità di misura reali.

Infine, nella fase di post-processing, vengono corretti eventuali errori nel campo di velocità attraverso l'applicazione di filtri di validazione, che consentono di identificare ed eliminare i vettori errati o incoerenti rispetto al flusso circostante.

3.1.2 Seeding

La scelta del *seeding* adatto è un passaggio fondamentale al fine di ottenere dei risultati accurati, pertanto in questo paragrafo si tratteranno i fattori principali che ne influenzano la scelta.

Forze gravitazionali

La *Particle Image Velocimetry* è fondata sull'elaborazione e analisi del moto delle particelle di *seeding* presenti nel fluido da studiare.

Proprio per questa ragione, tuttavia, risulta essere un metodo indiretto, essendo misurata la velocità delle particelle e non quella del fluido.

Una rilevante fonte di errore deriva dall'influenza delle forze gravitazionali, che ricorre in caso di differenza rilevante tra la densità del fluido ρ e la densità della particella di *seeding* ρ_p .

Facendo riferimento alla legge del *drag* di Stokes [1], valida per una sfera immersa in un fluido

viscoso a Reynolds molto bassi, è possibile derivare la velocità indotta dalle forze gravitazionali:

$$U_g = \frac{d_p^2 (\rho_p - \rho) g}{18 \mu} \quad (3.1)$$

In cui:

g è l'accelerazione di gravità

μ è la viscosità dinamica del fluido

d_p è il diametro della particella

A partire dalla 3.1 si ricava la velocità di deviazione di una particella in un fluido accelerato uniformemente:

$$U_s = U_p - U = \frac{d_p^2 (\rho_p - \rho) a}{18 \mu} \quad (3.2)$$

Dove U_p rappresenta la velocità della particella e U quella del fluido.

Se $\rho_p \gg \rho$, allora vale la relazione:

$$U_p(t) = U \left(1 - \exp \left(-\frac{t}{\tau_s} \right) \right) \quad (3.3)$$

In cui τ_s rappresenta il tempo di rilassamento.

$$\tau_s = \frac{d_p^2 \rho_p}{18 \mu} \quad (3.4)$$

Questo fenomeno risulta essere meno impattante nel caso in cui il fluido sia un liquido, rispetto al caso in cui questo sia un gas. Osservando la 3.1, si nota la stretta dipendenza della velocità di deviazione dal diametro della particella, si può pertanto dedurre la necessità di utilizzare un *seeding* le cui particelle abbiano un d_p piccolo.

Tuttavia ci sono dei limiti sul diametro, a causa dei fenomeni di diffusione della luce, che ne dipendono direttamente.

Diffusione della luce

Spesso le particelle utilizzate per il *seeding* non sono direttamente visibili alla telecamera, poiché possono avere dimensioni estremamente ridotte; tuttavia, diventano osservabili grazie al fenomeno della diffusione della luce.

Considerando particelle di diametro molto maggiore rispetto alla lunghezza d'onda della luce incidente, vale la teoria di Mie [1].

Si definisce un diametro normalizzato q :

$$q = \frac{\pi d_p}{\lambda} \quad (3.5)$$

Dove λ rappresenta la lunghezza d'onda e d_p il diametro della particella.

Se $q > 1$, nella distribuzione angolare di diffusione della luce appaiono circa q picchi locali.

Se ne deduce, quindi, che il fenomeno di diffusione aumenta con il crescere del diametro delle particelle stesse.

In condizioni di validità della teoria di Mie, la luce viene diffusa maggiormente in avanti (0°) e ad intensità minore a 180° , ovvero all'indietro.

Sarebbe vantaggioso, infatti, registrare a 0° , ovvero in condizioni di diffusione in avanti, ma a causa della limitata profondità di campo si tende a registrare a 90° .

Per aumentare l'intensità della diffusione, un'alternativa all'impiego di particelle di diametro maggiore è l'utilizzo di un numero elevato di particelle. Si induce, infatti, un fenomeno di *multiscattering*, che incrementa l'intensità della luce diffusa.

Una soluzione ulteriore ricade nell'impiego di seeding composti da particelle di dimensioni differenti, tuttavia ciò potrebbe alterare il risultato delle analisi PIV, essendo le particelle di diametro maggiore più visibili.

3.1.3 Ottica e fenomeni di diffrazione

Come affermato precedentemente, al fine di condurre un'analisi PIV, è necessario illuminare il campo d'interesse con un piano di luce laser.

Tuttavia, il laser emette un fascio di luce collimata, pertanto vengono impiegate delle lenti cilindriche e sferiche per trasformarlo in un piano.

Come prima lente può essere impiegata una lente cilindrica divergente, ovvero con focale negativa, questo permette di evitare linee focali. In corrispondenza di queste, infatti, potrebbero bruciarsi eventuali particelle di polvere presenti nell'aria.

L'utilizzo di diverse tipologie di lenti, sia cilindriche che sferiche, permette successivamente di regolare le dimensioni del piano di luce.

Inoltre, le lenti non rivestite presentano una riflettività che si attesta intorno al 4%, calcolabile come [1]:

$$\left(\frac{n-1}{n+1}\right)^2 \quad (3.6)$$

Dove n rappresenta l'indice di rifrazione del materiale.

Solitamente questo effetto si ritiene trascurabile, tuttavia può causare danni all'attrezzatura circostante se ne è direttamente esposta. In ogni caso, ciò può essere prevenuto orientando correttamente le lenti.

Fenomeni di diffrazione

Una particella di piccole dimensioni, lontana dall'obiettivo di una telecamera, quando ripresa non appare puntiforme, bensì assume una figura di diffrazione di Fraunhofer.

Nel caso di bassa esposizione si ottiene una trama circolare che prende il nome di *disco di Airy*, che può essere circondata da *anelli di Airy* in caso di alta esposizione.

Il diametro d_{diff} del disco di Airy rappresenta l'immagine della più piccola particella che può essere ottenuta in una determinata configurazione.

In generale la dimensione della trama dipende strettamente dal diametro di apertura D_a dell'obiettivo. A D_a maggiori corrisponde la formazione di dischi più piccoli [1]:

$$\frac{d_{diff}}{2x_0} = 1.22 \quad (3.7)$$

Dove $x_0 = \frac{\lambda}{D_a}$

Il diametro del disco di Airy può essere trovato anche con la seguente formula [1]:

$$d_{diff} = 2.44 f\# (M + 1) \lambda \quad (3.8)$$

In cui:

M rappresenta il fattore di ingrandimento

$f\# = \frac{f}{D_a}$ è l'*f-number*

f è la focale dell'obiettivo della telecamera

La distanza focale f può essere trovata con la seguente formula[1]:

$$\frac{1}{z_0} + \frac{1}{Z_0} = \frac{1}{f} \quad (3.9)$$

Dove z_0 rappresenta la distanza tra il piano dell'immagine e la lente e Z_0 , invece, è la distanza tra la lente e il piano dell'oggetto.

Il fattore di ingrandimento si definisce, invece, come [1]:

$$M = \frac{z_0}{Z_0} \quad (3.10)$$

Può, dunque, essere determinato il diametro della particella nell'immagine [1]:

$$d_\tau = \sqrt{(Md_p)^2 + d_{\text{diff}}^2} \quad (3.11)$$

Facendo riferimento alla 3.11, è possibile notare come il diametro della particella nell'immagine sia dipendente in parte dalle sue dimensioni geometriche e dall'ingrandimento dell'immagine e in parte dal diametro della trama di diffrazione.

In condizioni di validità della teoria di Mie, ovvero per particelle con $d_\tau \gg \lambda$, il fenomeno della diffrazione assume una rilevanza di gran lunga minore, essendo in questo caso d_τ principalmente dipendente da proprietà geometriche, ovvero dal valore $(Md_p)^2$.

Un comune errore nel calcolo dei campi di velocità in un'analisi PIV è dato dalla dimensione assunta dalle particelle nell'immagine. Ridurre d_τ permette di minimizzare questo effetto.

Inoltre, per ottenere abbastanza luce da ogni particella presente nel piano illuminato, è necessario mantenere un'apertura sufficientemente grande. Questo conduce anche alla formazione di dischi di Airy di dimensioni inferiori e quindi porta ad avere immagini più nitide.

Tuttavia, un'apertura maggiore causa importanti fenomeni di aberrazione, pertanto deve essere limitata.

3.2 Realizzazione del setup ottico

Al fine di eseguire la caratterizzazione fluidodinamica del levitatore, è stato sviluppato un apposito setup ottico (fig. 3.2) costituito da:

- A) High speed camera, obiettivo SIGMA 180mm f/3.5 EX DG HSM MACRO
- B) Laser continuo MGL-FN-532-1W
- C) Specchio, lente cilindrica e lente sferica
- D) Generatore di fumo SAFEX FOG 2010
- E) Lastre di plexiglas e cartone nero

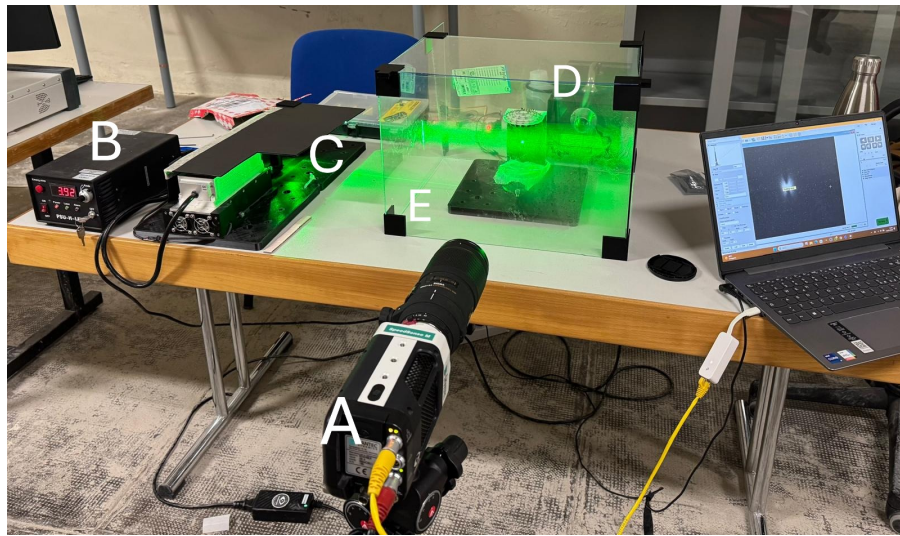


Figura 3.2: Setup ottico in funzionamento: A) high speed camera, B) laser continuo MGL-FN-532-1W, C) specchio, lente cilindrica e lente sferica, D) generatore di fumo SAFEX FOG 2010, E) lastre di plexiglas e cartone nero

3.2.1 Ottica

In questo setup sono state impiegate, in ordine, una lente cilindrica divergente e una lente sferica con distanza focale di 500 mm. Proprio a tale distanza è stato posizionato il levitatore. La prima lente ha lo scopo di trasformare il fascio collimato in un piano di luce, la seconda ha, invece, l'effetto di modificare lo spessore del piano stesso.

L'utilizzo di uno specchio correttamente direzionato ha, poi, permesso di deviare il piano di luce così creato, indirizzandolo verso la particella in levitazione.

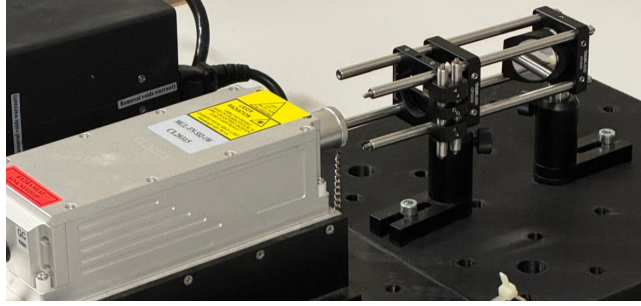


Figura 3.3: Laser continuo e ottica di formazione del piano di luce laser

Il sistema di lenti e specchi è stato realizzato utilizzando componenti della THORLABS [6] e poi è stato fissato, così come il laser, a delle piastre forate ottenute tramite stampa 3D. Ciò ha permesso di allineare correttamente tutti gli elementi. Il tutto è stato, infine, coperto con un rivestimento in cartone nero, in modo da schermare possibili riflessioni.

3.2.2 Realizzazione del box

Al fine di ottenere una densità di *seeding* adeguata si è deciso di costruire un box in plexiglas da posizionare al di sopra del levitatore.

La parte posteriore è stata lasciata aperta in prossimità del generatore di fumo ed è stata chiusa dopo ogni getto di fumo con una lastra appositata. La parete frontale al laser è stata realizzata in cartone nero, così come il rivestimento posto sull'ottica, per questione di sicurezza. La superficie scura e opaca del cartone, infatti, limita al minimo gli effetti di riflessione della luce.

Essendo il box costituito da lastre separate, in modo da favorirne la modularità e permettere la sostituzione di una parete con il cartone, sono stati disegnati al CAD e poi mandati in stampa, utilizzando il software *IdeaMaker* [10], i componenti angolari necessari al suo assemblaggio.

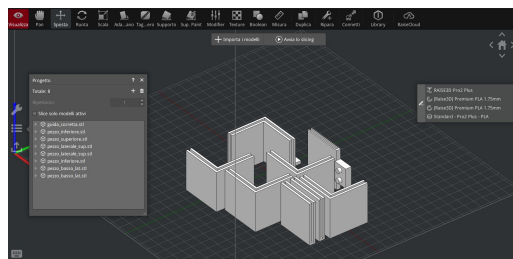


Figura 3.4: Ordine di stampa di alcuni dei componenti angolari del box

3.2.3 Generatore di fumo e high speed camera

La scelta del tipo di *seeding* è dipesa dalle ragioni già citate nel paragrafo 3.1.2, ed è ricaduta sull'impiego di particelle traccianti di fumo. Il generatore di fumo *SAFEX FOG 2010* è stato posizionato dietro il levitatore, ma spostato lateralmente, in modo che l'emissione di fumo non disturbasse la particella in levitazione.

La *high speed camera*, dotata dell'obiettivo SIGMA 180mm f/3.5 EX DG HSM MACRO, è stata posta frontalmente ed ortogonalmente al piano di luce, registrando dunque in *scattering* a 90 °.

Le immagini sono state acquisite con il software *Phantom Camera Control* (PCC) 3.11 [11], associato alla videocamera.

3.3 Risultati

Al centro di questa tesi si è posto uno studio dei campi di velocità indotti dalla levitazione di sfere di polistirolo e gocce d'acqua in aria.

Le immagini, come già affermato nel paragrafo precedente, sono state acquisite con una *high speed camera*, ad una frequenza pari a 3700 Hz.



Figura 3.5: Goccia d'acqua in levitazione, immagine acquisita con videocamera ad alta velocità

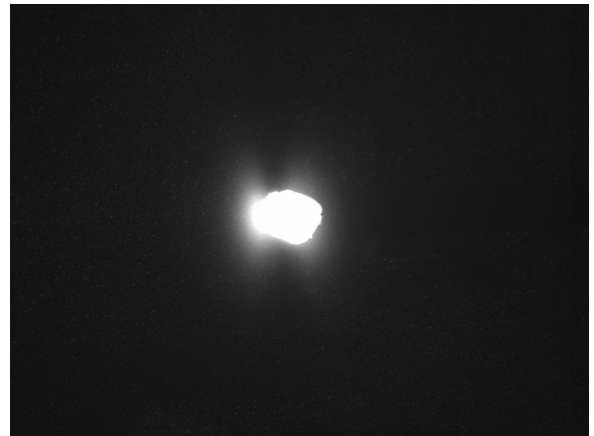


Figura 3.6: Sfera di polistirolo in levitazione, immagine acquisita con videocamera ad alta velocità

3.3.1 Impostazioni PIVlab

Le immagini sono state, successivamente, sottoposte ad una fase di pre-processing con il software PIVlab.

Per rendere le particelle di *seeding* maggiormente visibili e diminuire il disturbo dovuto alla riflessione della luce laser da parte del polistirolo e dell'acqua, sono stati applicati diversi filtri. Il *CLAHE* (*Contrast Limited Adaptive Histogram Equalization*) permette di migliorare il contrasto locale dell'immagine e rendere le particelle maggiormente visibili in tutto il campo. Il filtro *passa alto*, invece, rimuove le componenti a bassa frequenza e l'*auto contrast stretch* agisce sul contrasto globale.

Sono state, poi, definite le impostazioni relative al processamento delle immagini.

L'algoritmo selezionato per la PIV è il *Multipass FFT window deformation*, che impiega la configurazione a più pass e utilizza la FFT (*Fast Fourier Transform*) come tecnica di cross-correlazione e sfrutta la possibilità di deformare le finestre di interrogazione per ottenere una maggiore accuratezza nei calcoli.

Sono stati impostati tre pass di dimensioni differenti, il primo con area di interrogazione di 64x64 px, con step di 32 px, il secondo di 32x32 px, con step di 16 px e il terzo di 16x16 px, con step di 8 px.

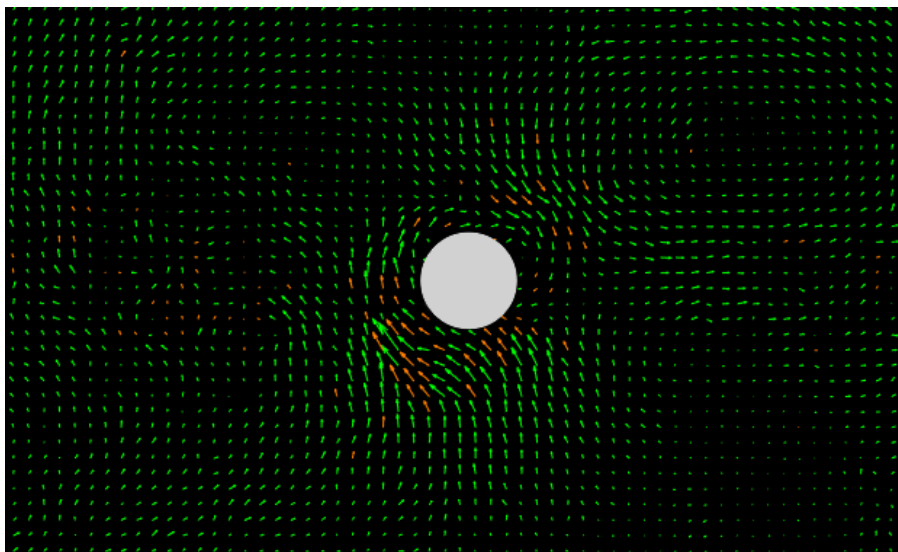
L'utilizzo di più pass permette di ottenere una maggiore risoluzione spaziale del campo vettoriale, ma, al contempo, aumenta il peso computazionale.

Per ottenere una maggiore precisione nel calcolo dello spostamento delle particelle, ovvero nell'identificazione del massimo della funzione di cross-correlazione, è stato impostato un *sub-pixel estimator* di tipo *Gauss 2x3 point*.

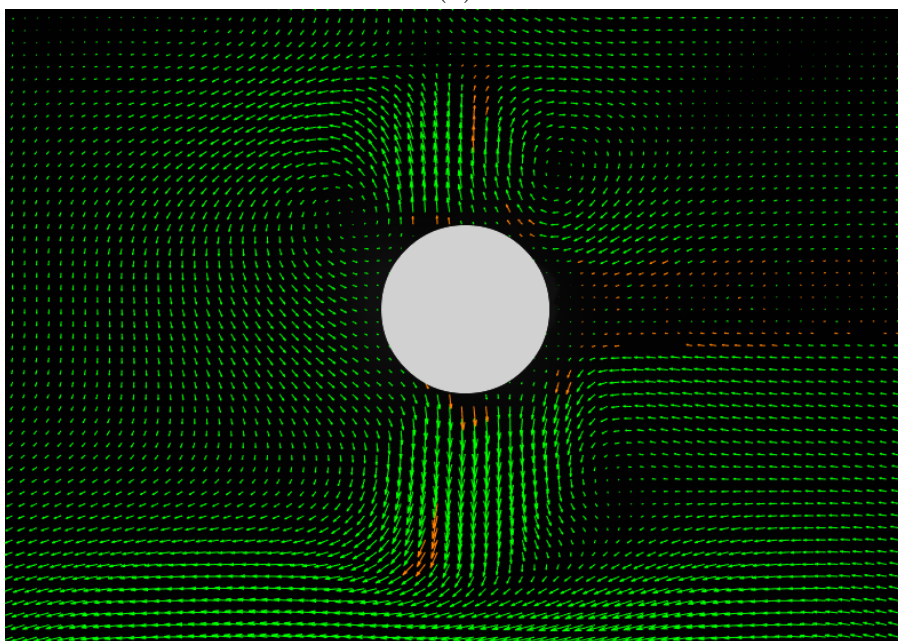
L'impiego di immagini di calibrazione ha permesso di riportare i valori di velocità in m/s.

Infine, i risultati ottenuti sono stati sottoposti ad una fase di post-processing, in cui l'impiego di un *Local median filter Threshold* con soglia pari a 2 ha permesso di eliminare i vettori errati, ovvero quelli la cui differenza con la mediana assumeva un valore maggiore del doppio della deviazione media.

I campi di velocità medi sono stati calcolati su 925 frames, ovvero in un intervallo temporale di 0.25 s. La fig. 3.7 mostra due campi di velocità ottenuti con queste impostazioni.



(a)



(b)

Figura 3.7: (a) campo di velocità, goccia d'acqua
(b) campo di velocità, sfera di polistirolo

3.3.2 Post-processing con MATLAB

I dati ottenuti sono stati, in seguito, rielaborati tramite un codice MATLAB (Appendice [A.3](#) e [A.4](#)), in modo da ottenere, oltre alla rappresentazione del campo vettoriale, anche delle *colormaps* per mettere in evidenza i valori assunti dalla velocità in ogni zona del campo.

Sono state ottenute le rappresentazioni grafiche dei campi di velocità relativi alla velocità media, alla componente u (diretta lungo l'asse x) e alla componente v (diretta lungo l'asse y).

Essendo l'area di maggiore interesse nel campo vettoriale, l'unica zona mostrata dalle immagini è quella nell'immediato intorno degli oggetti in levitazione.

Alle immagini sono state aggiunte delle circonferenze per indicare la dimensione e la posizione sia della sfera di polistirolo che della goccia d'acqua. Come già affermato, le particelle in levitazione non possono avere una dimensione maggiore di $\frac{\lambda}{2}$, ovvero della distanza tra due nodi consecutivi, che nel caso del TinyLev risulta essere 4 mm.

La sfera di polistirolo misura circa 2.4 mm, la goccia d'acqua ha, invece, un diametro di 1 mm.

Le immagini ottenute mostrano la variazione dei campi di velocità ogni 0.25 s, pertanto è possibile studiare come varia il flusso dell'aria in un arco temporale di 1.25 s.

Sfera di polistirolo, modulo della velocità

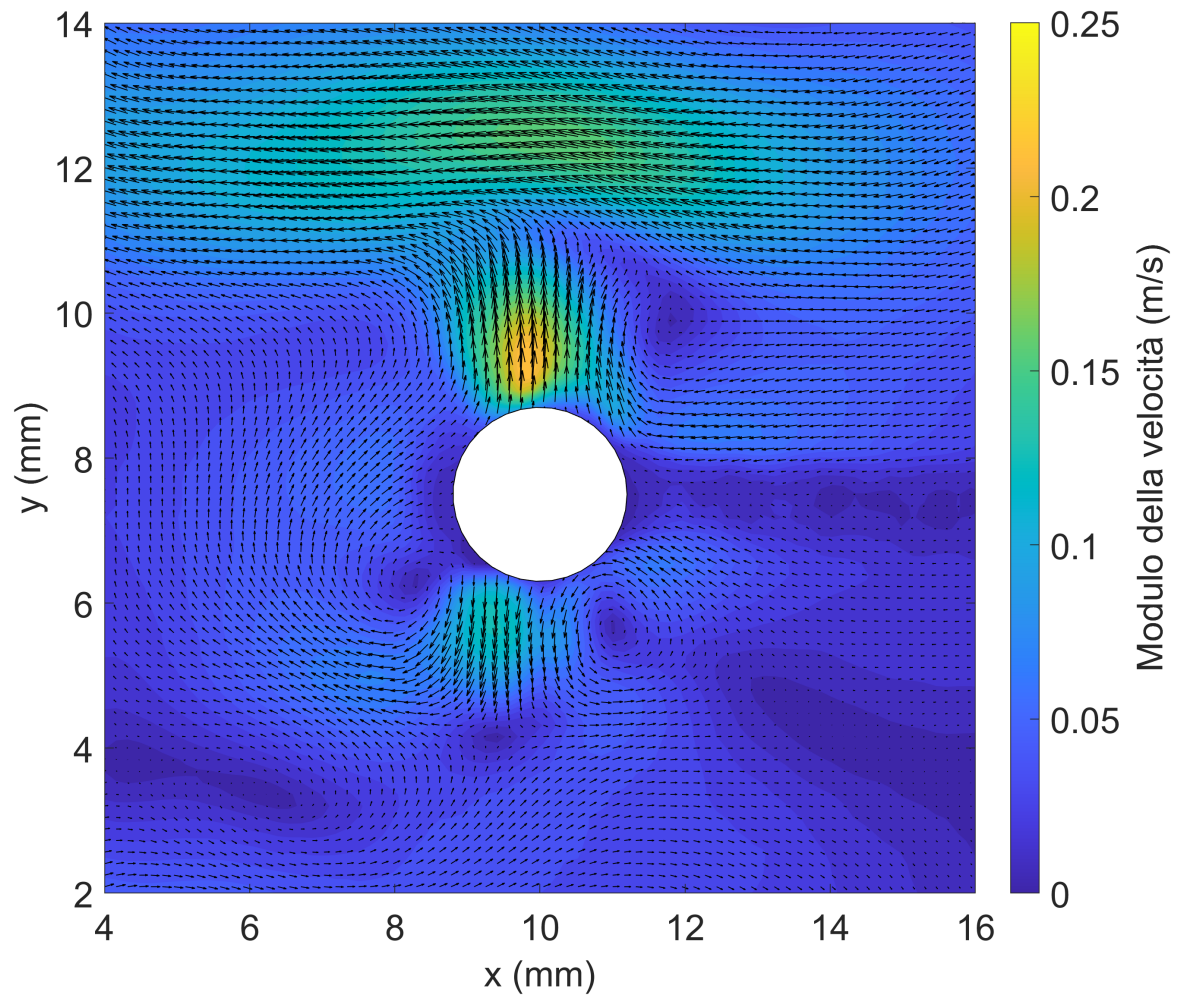


Figura 3.8: Campo del modulo della velocità intorno alla sfera di polistirolo: 0.25 s

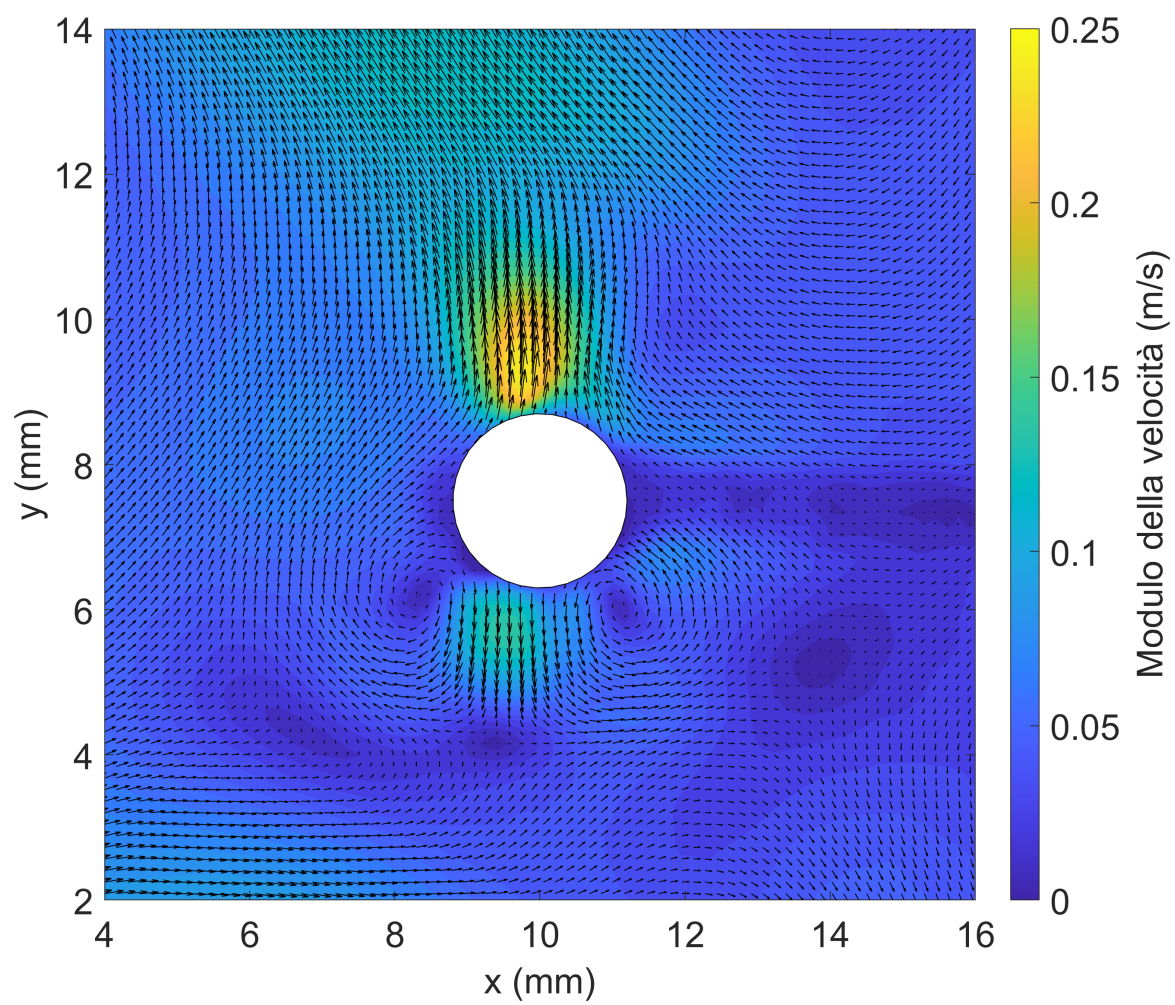


Figura 3.9: Campo del modulo della velocità intorno alla sfera di polistirolo: 0.50 s

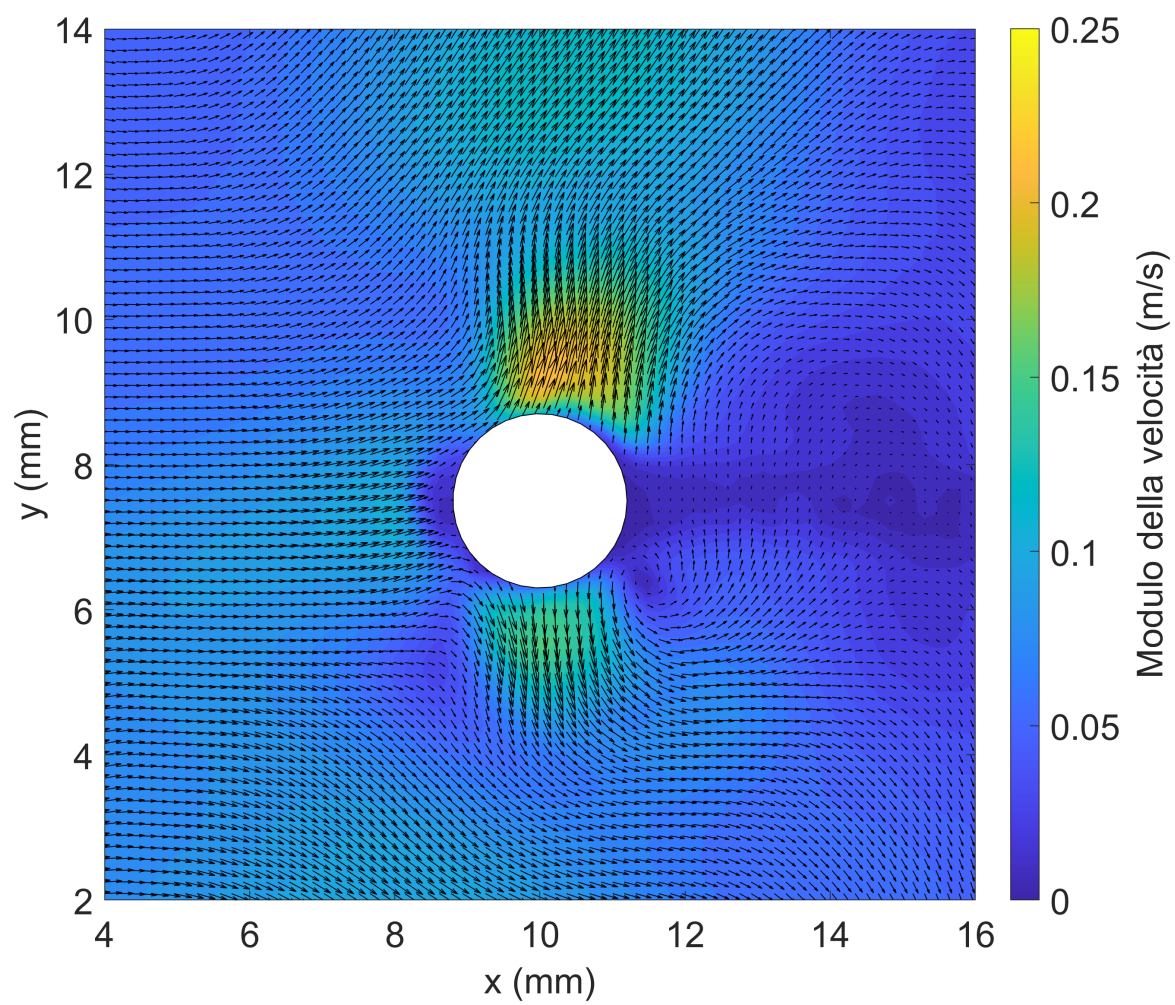


Figura 3.10: Campo del modulo della velocità intorno alla sfera di polistirolo: 0.75 s

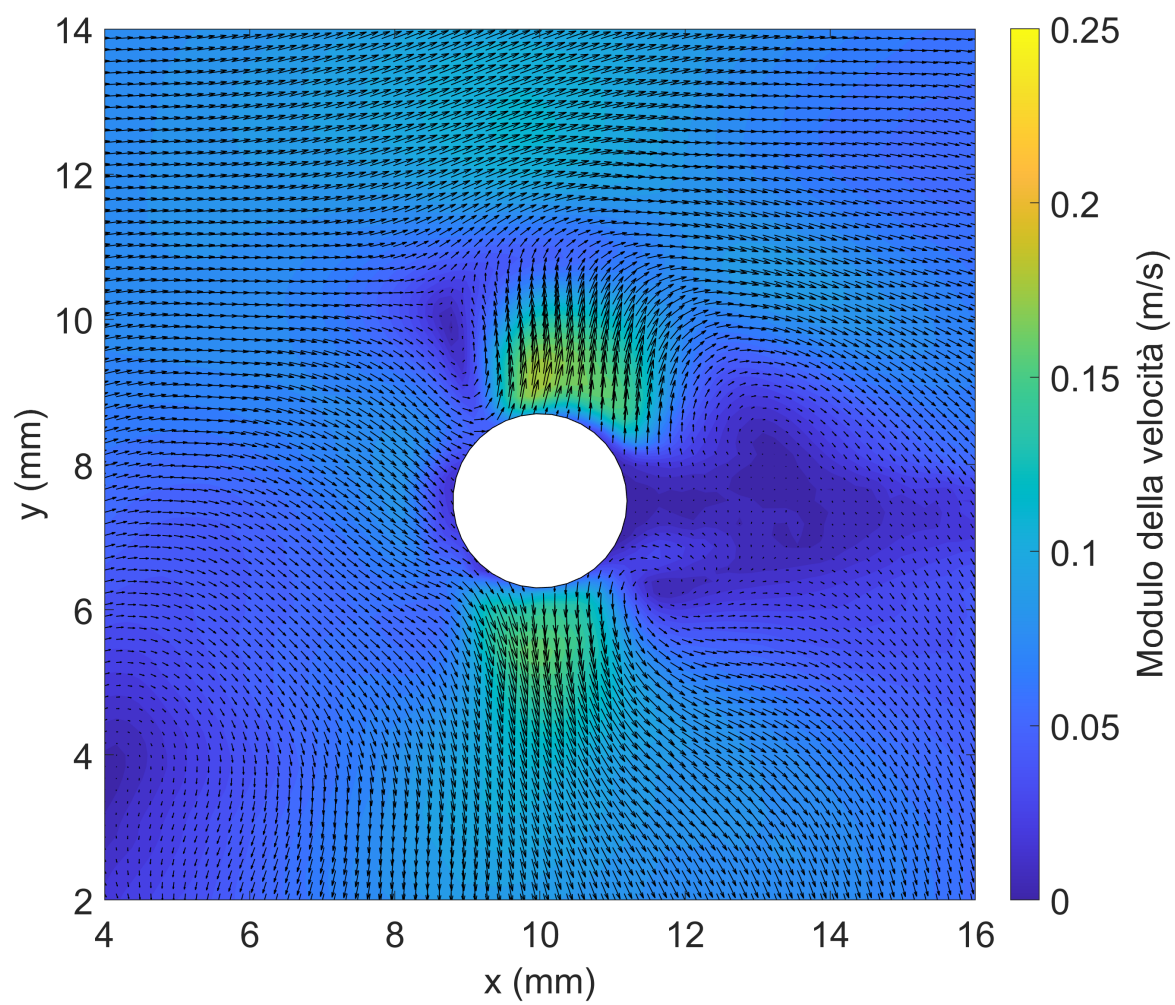


Figura 3.11: Campo del modulo della velocità intorno alla sfera di polistirolo: 1.00 s

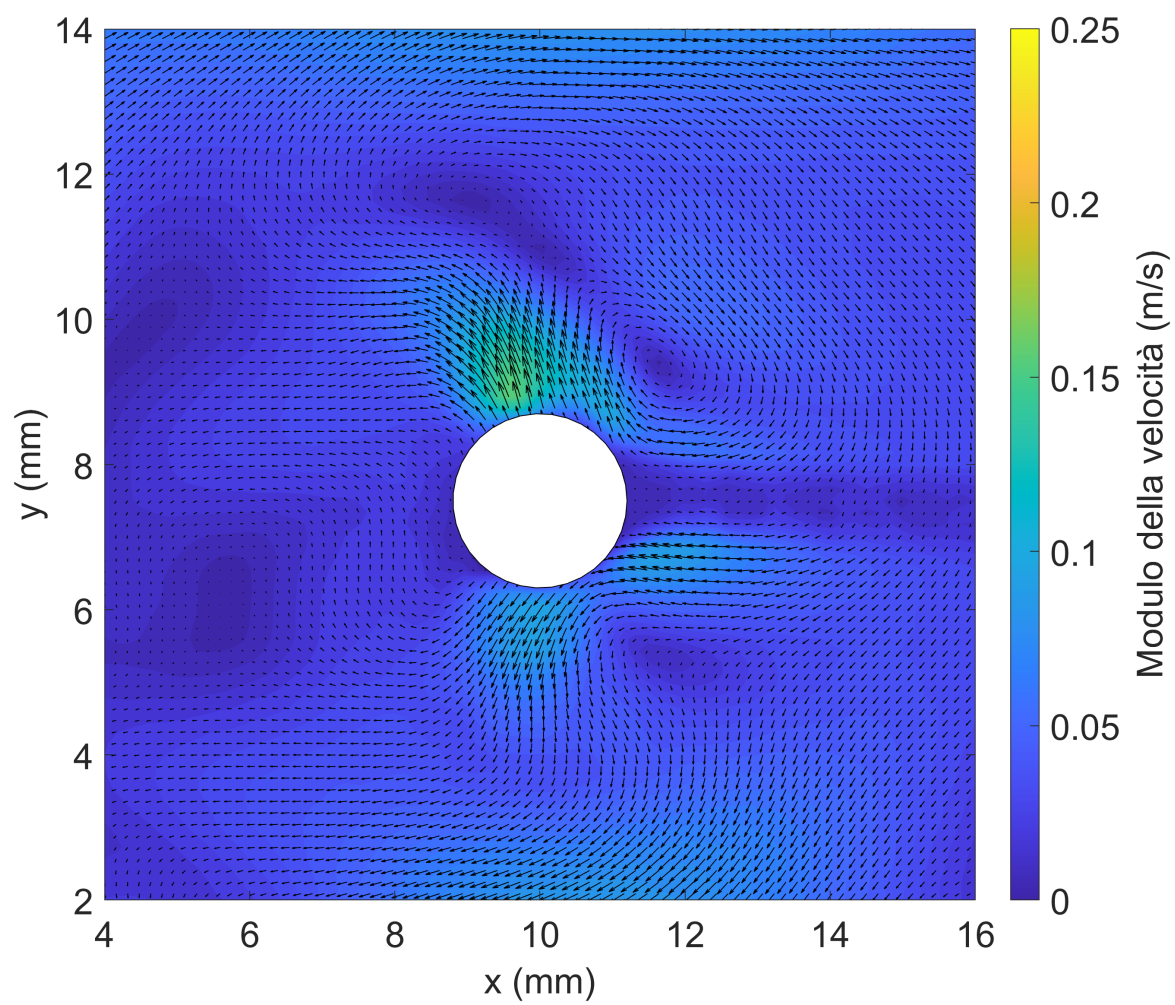


Figura 3.12: Campo del modulo della velocità intorno alla sfera di polistirolo: 1.25 s

Sfera di polistirolo, componente u della velocità

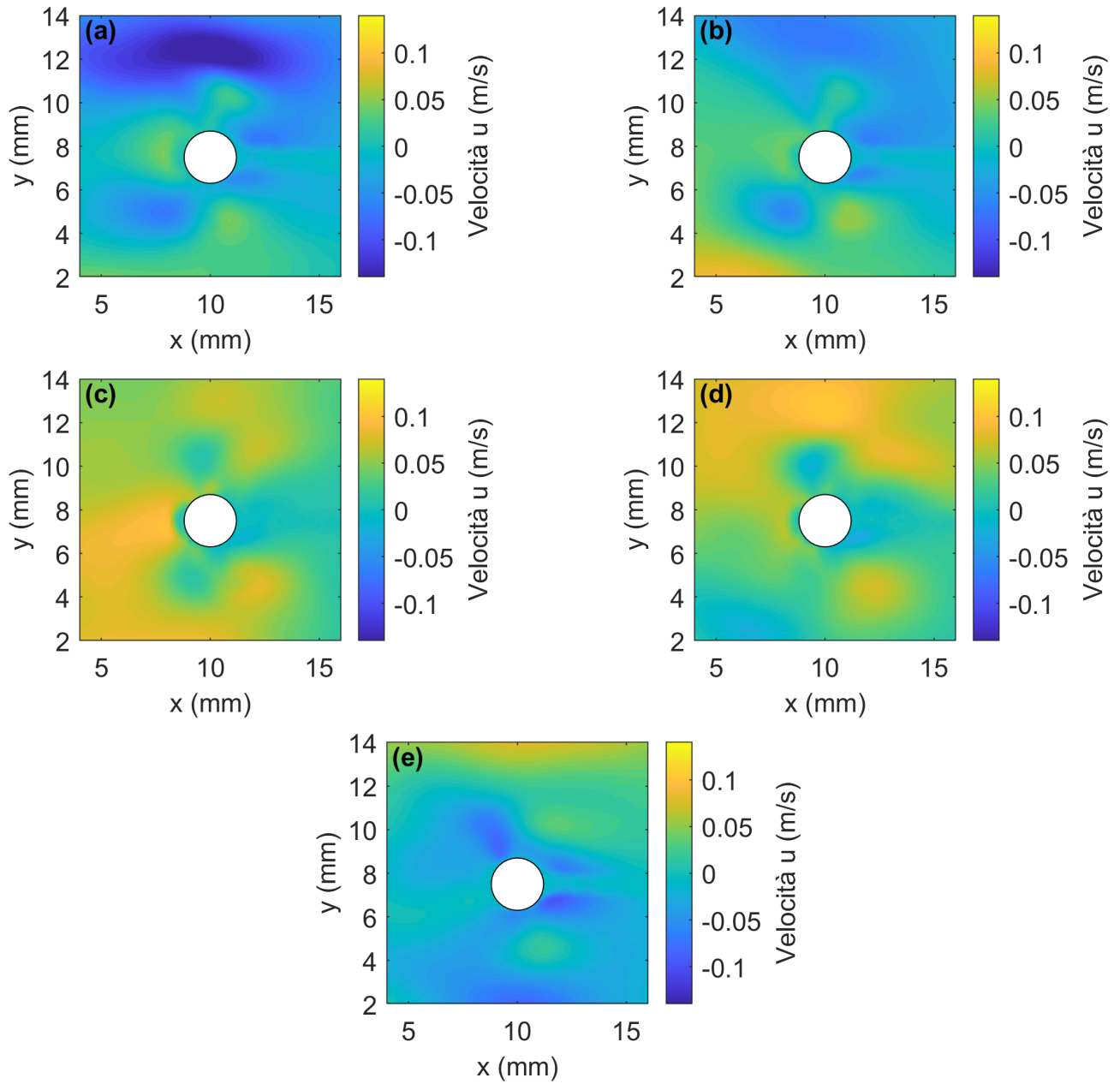


Figura 3.13: Componente u della velocità intorno alla sfera di polistirolo: (a) 0.25 s, (b) 0.5 s, (c) 0.75 s, (d) 1.0 s, (e) 1.25 s.

Sfera di polistirolo, componente v della velocità

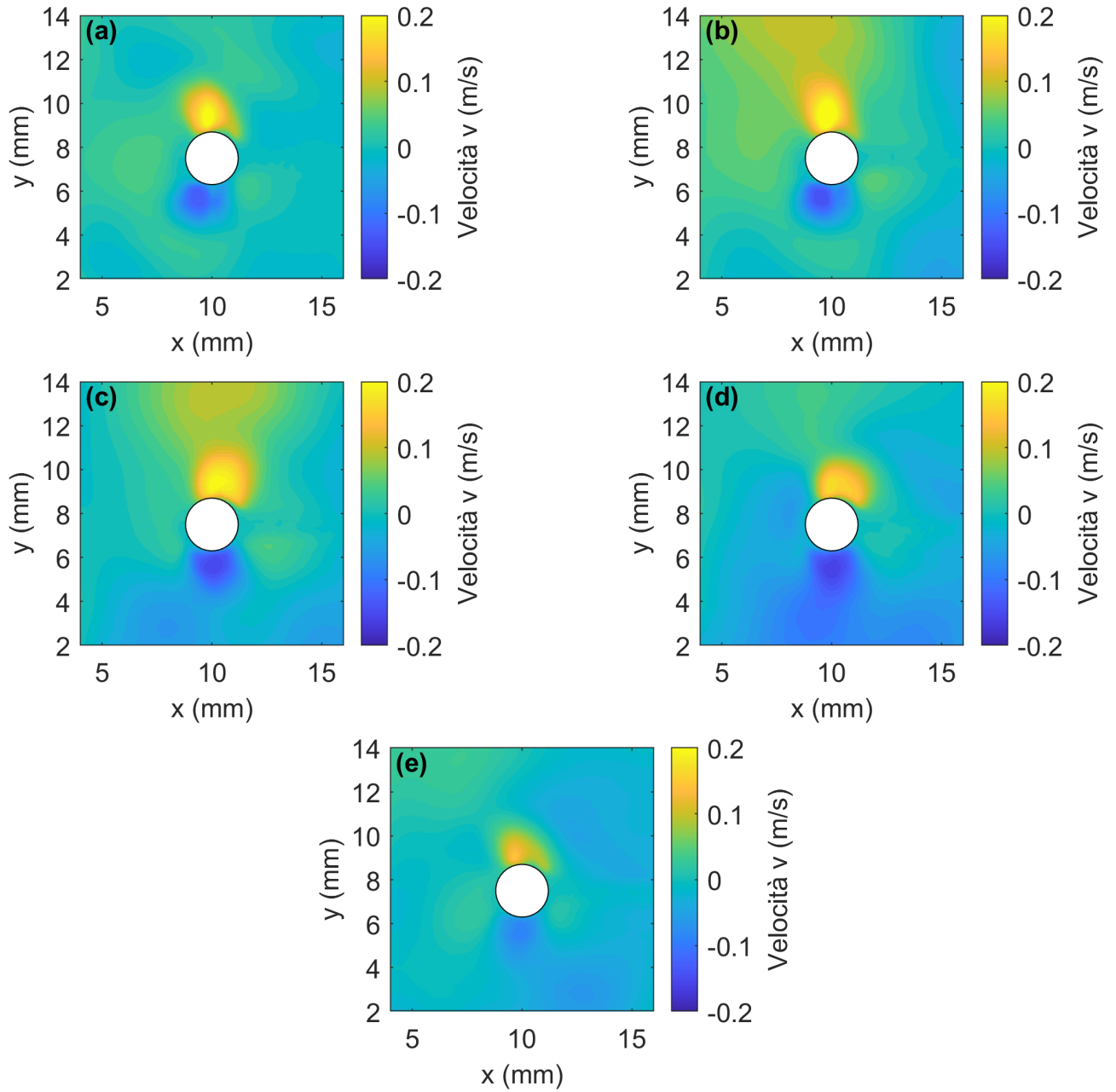


Figura 3.14: Componente v della velocità intorno alla sfera di polistirolo: (a) 0.25 s, (b) 0.5 s, (c) 0.75 s, (d) 1.0 s, (e) 1.25 s.

Goccia d'acqua, modulo della velocità

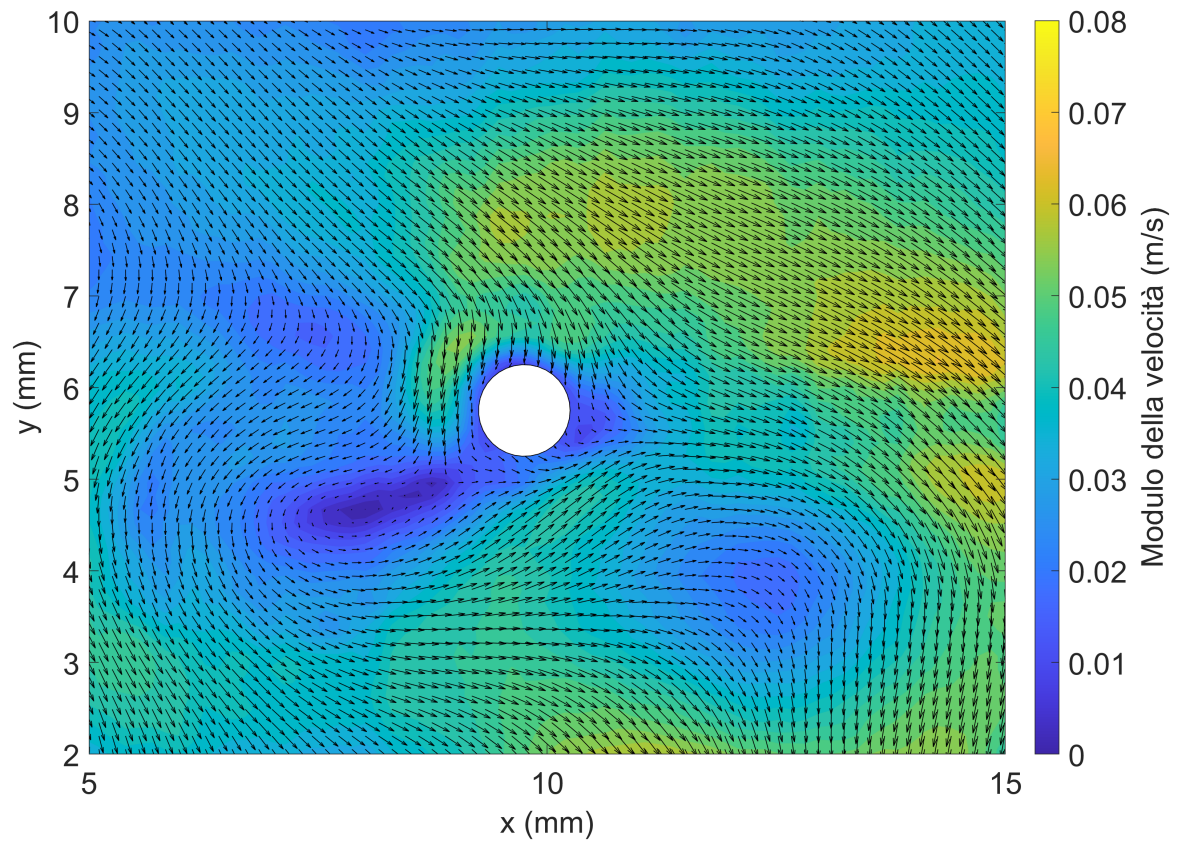


Figura 3.15: Campo del modulo della velocità intorno alla goccia d'acqua: 0.25 s

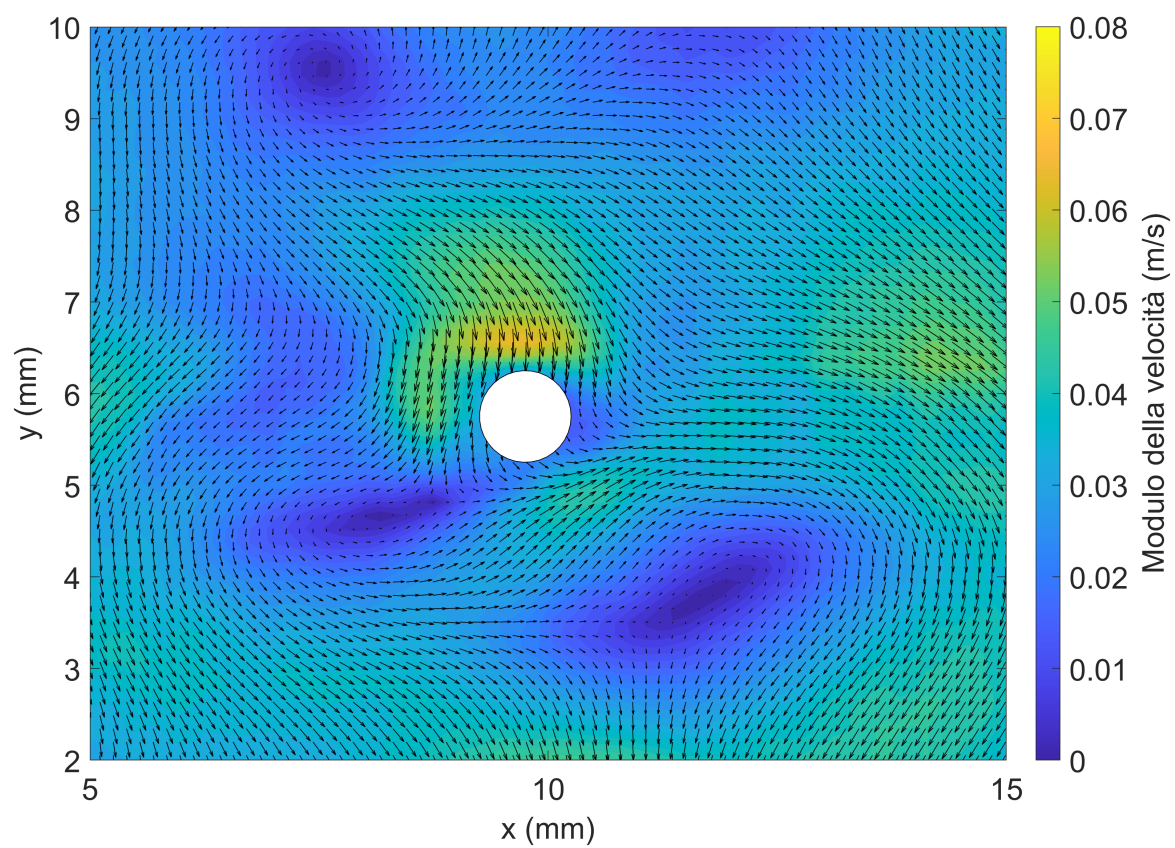


Figura 3.16: Campo del modulo della velocità intorno alla goccia d'acqua: 0.50 s

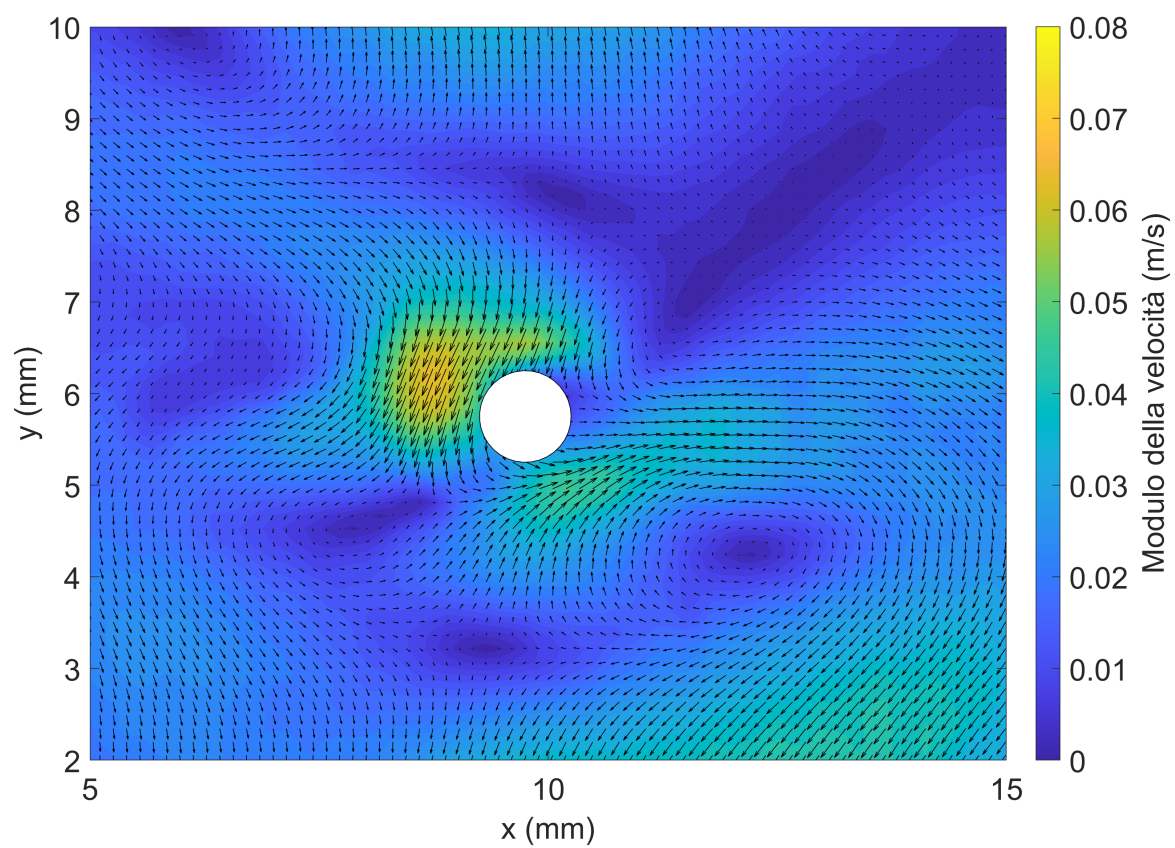


Figura 3.17: Campo del modulo della velocità intorno alla goccia d'acqua: 0.75 s

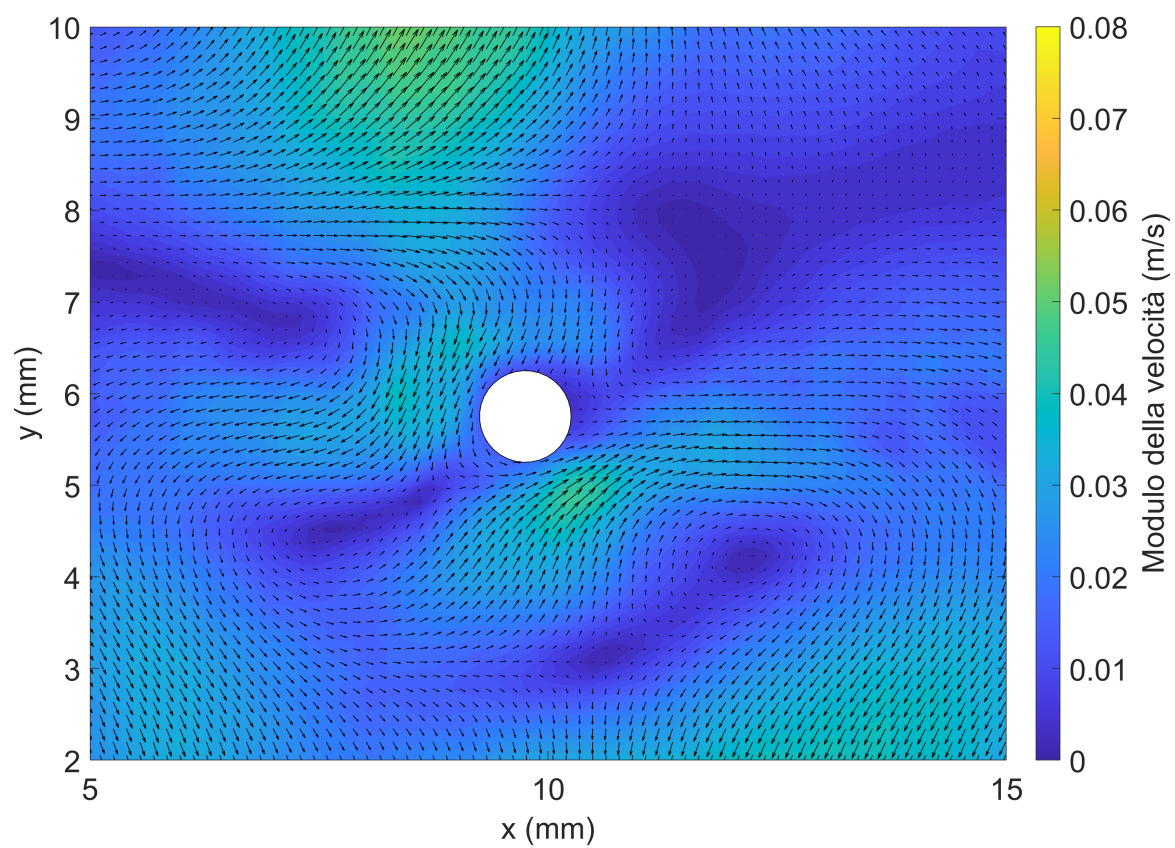


Figura 3.18: Campo del modulo della velocità intorno alla goccia d'acqua: 1.00 s

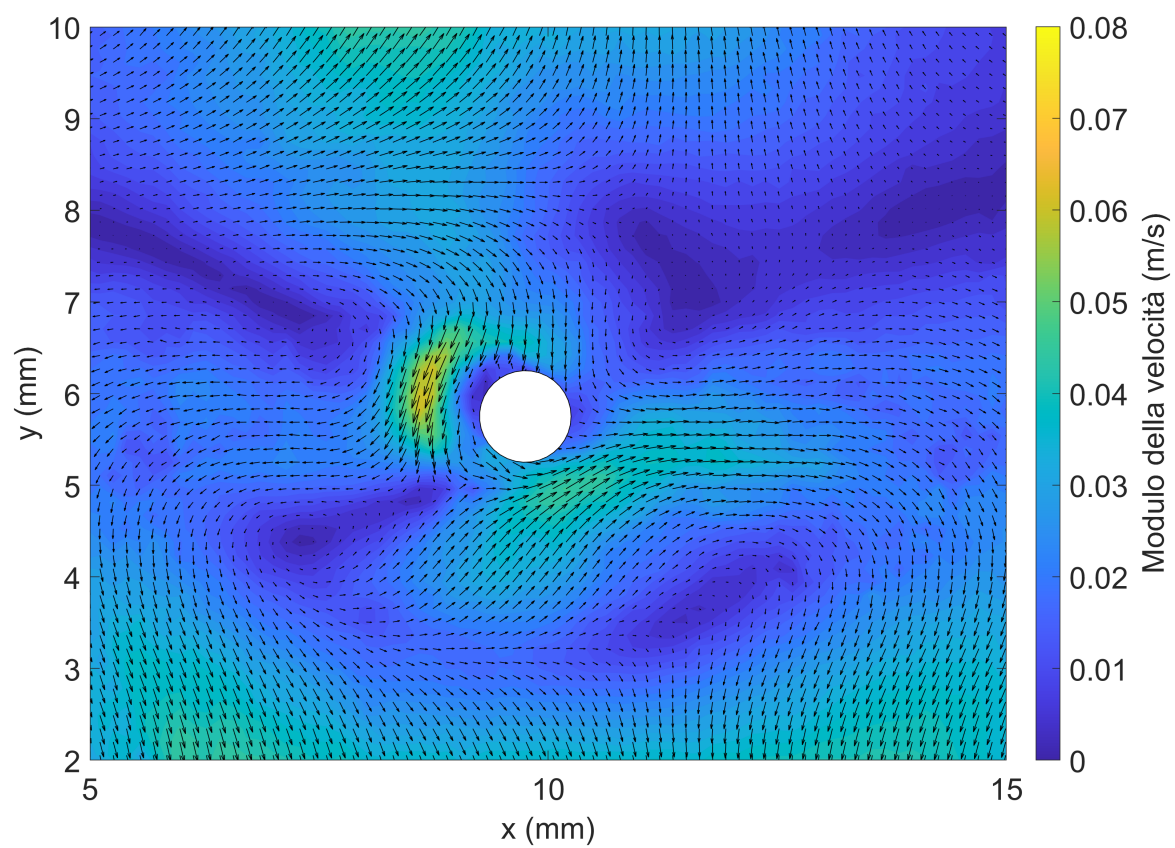


Figura 3.19: Campo del modulo della velocità intorno alla goccia d'acqua: 1.25 s

Goccia d'acqua, componente u della velocità

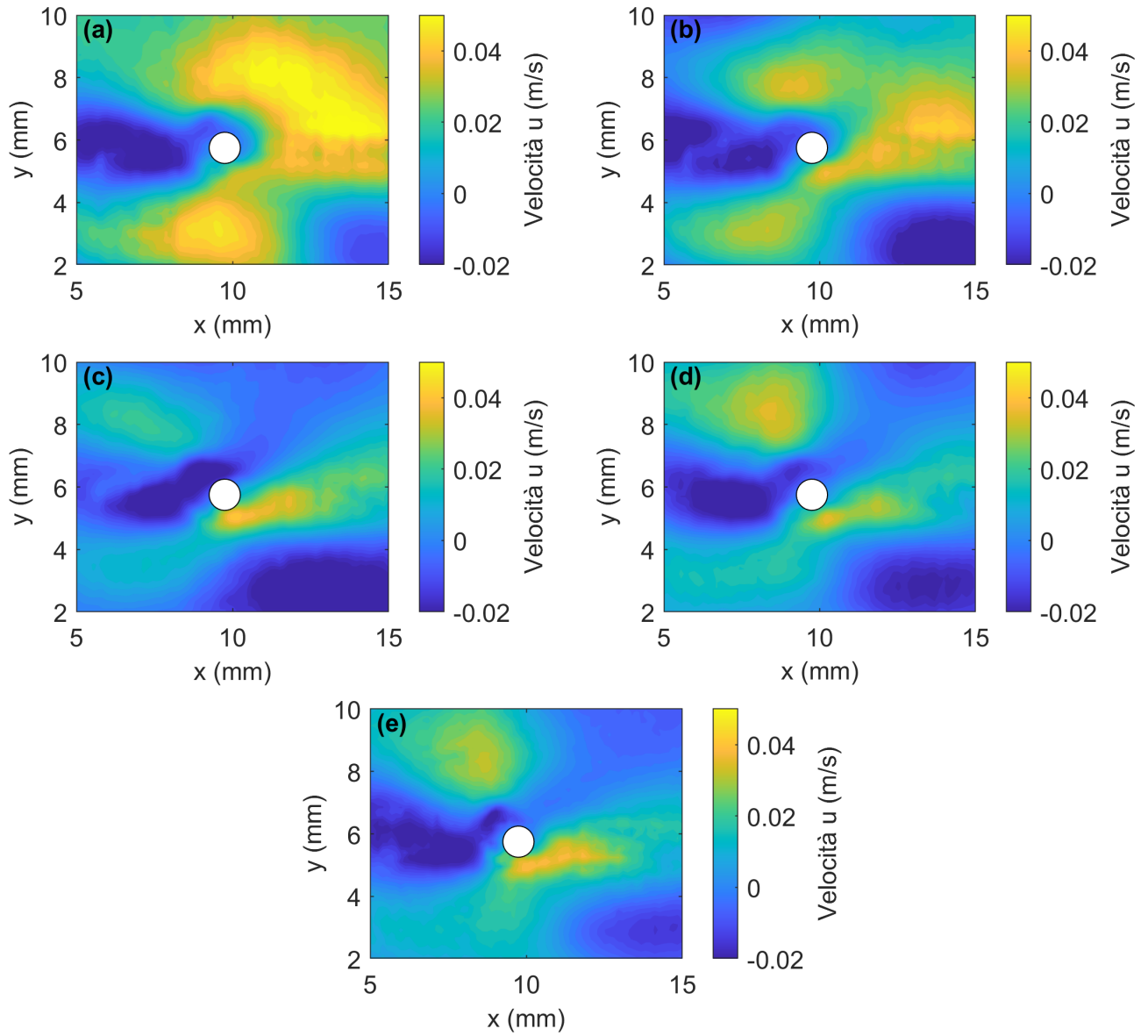


Figura 3.20: Componente u della velocità intorno alla goccia d'acqua: (a) 0.25 s, (b) 0.5 s, (c) 0.75 s (d) 1.0 s, (e) 1.25 s.

Goccia d'acqua, componente v della velocità

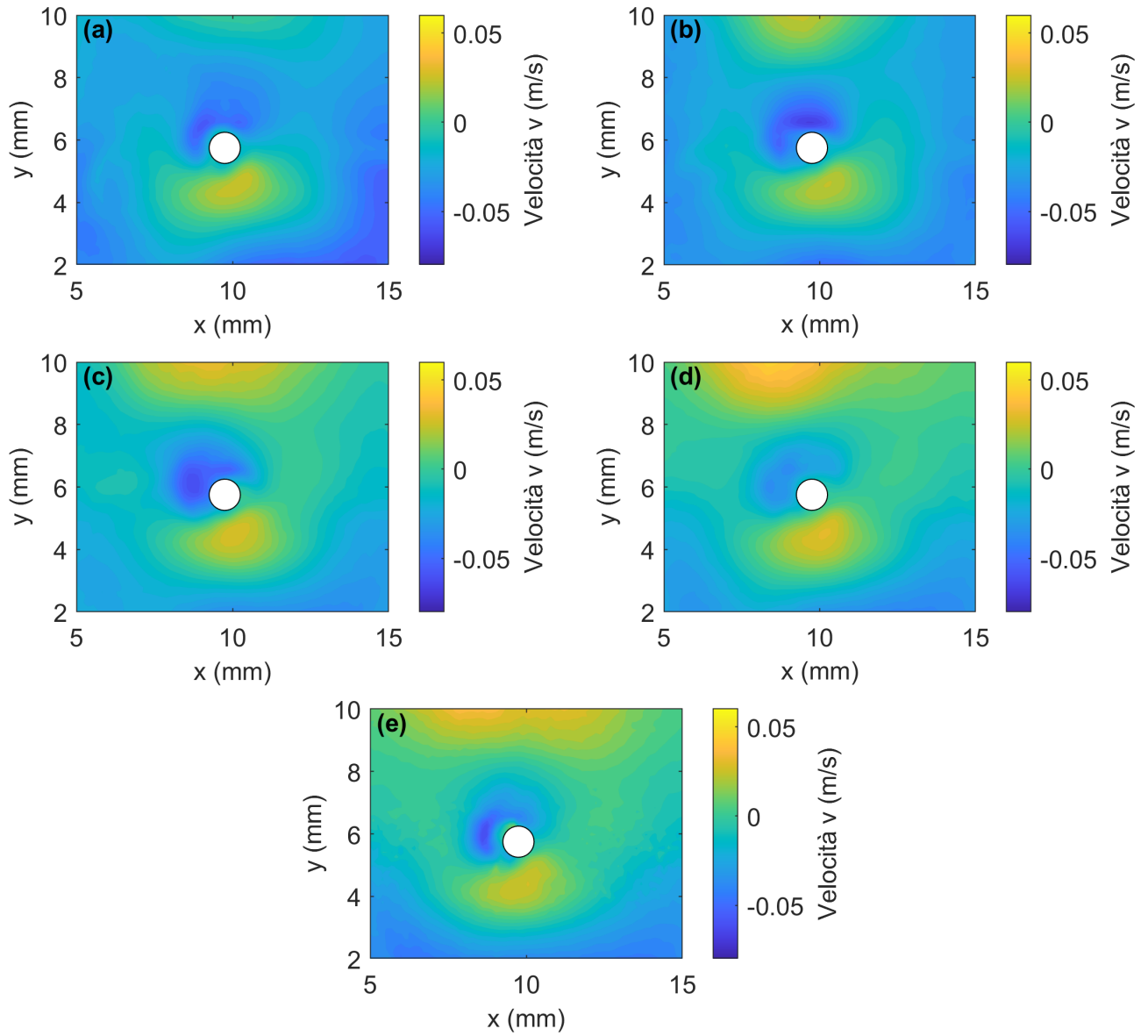


Figura 3.21: Componente v della velocità intorno alla goccia d'acqua: (a) 0.25 s, (b) 0.5 s, (c) 0.75 s (d) 1.0 s, (e) 1.25 s.

Capitolo 4

Discussione

Le immagini PIV presentate nel Capitolo 3 evidenziano sostanziali differenze nei campi di velocità indotti dalla levitazione della sfera di polistirolo e della goccia d'acqua. Le analisi sono state effettuate ad intervalli regolari, da $t=0.25$ s a $t=1.25$ s, permettendo di compararne la variazione nel tempo.

4.1 Sfera di polistirolo

Modulo della velocità

La levitazione della sfera di polistirolo induce campi di velocità il cui modulo ha un valore minimo che si attesta a 0.02 m/s e un valore massimo che varia istantaneamente:

- $t=0.25$ s: 0.2 m/s
- $t=0.50$ s: 0.22 m/s
- $t=0.75$ s: 0.2 m/s
- $t=1.00$ s: 0.16 m/s
- $t=1.25$ s: 0.14 m/s

La velocità massima è raggiunta al di sopra e al di sotto della sfera (figg. 3.8 - 3.12).

Componente orizzontale u

Osservando, invece, la variazione della sola componente u della velocità (fig. 3.13), questa assume valori che oscillano tra 0.1 m/s e -0.01 m/s, ma che cambiano notevolmente istante per istante.

Componente verticale v

Relativamente alla componente v (fig. 3.14), il valore massimo, compreso tra 0.1 e 0.2 m/s, si raggiunge al di sopra della sfera di polistirolo e il valore minimo, che si attesta tra -0.05 e -0.15 m/s, si presenta al di sotto della particella.

4.2 Goccia d'acqua

Modulo della velocità

I campi di velocità indotti dalla levitazione della goccia d'acqua sono caratterizzati dal raggiungimento di valori massimi e minimi inferiori rispetto al caso del polistirolo.

Il modulo della velocità (figg. 3.15 - 3.19) assume infatti i seguenti valori:

- $t=0.25$ s: massimo 0.08 m/s, minimo 0.01 m/s
- $t=0.50$ s: massimo 0.07 m/s, minimo 0.01 m/s
- $t=0.75$ s: massimo 0.06 m/s, minimo 0.01 m/s
- $t=1.00$ s: massimo 0.055 m/s, minimo 0.005 m/s
- $t=1.25$ s: massimo 0.06 m/s, minimo 0.01 m/s

Componente orizzontale u

La componente u (fig. 3.20) raggiunge valori massimi di 0.04 m/s al di sotto e a destra della goccia e valori minimi dell'ordine di -0.02 m/s nell'area al di sopra e a sinistra della goccia.

Componente verticale v

La componente v (fig. 3.21), invece, presenta i suoi valori massimi al di sotto della goccia e quelli minimi al di sopra di essa. Tali valori sono compresi nell'intervallo tra -0.08 m/s e 0.04 m/s

4.3 Osservazioni

Sia nel caso della goccia d'acqua che in quello del polistirolo, i risultati ottenuti nella fascia centrale del lato destro delle immagini potrebbero risultare poco affidabili.

Infatti, in tale zona è presente una fascia d'ombra causata dalla riflessione della luce laser sulla superficie delle sfere poste in levitazione, illuminate da sinistra. Le particelle traccianti sono rese, dunque, meno visibili, influenzando l'accuratezza dei risultati numerici. Possibili errori di calcolo sono, comunque, stati corretti in fase di post-processing, come già descritto nel paragrafo 3.3.1.

Inoltre, essendo la temperatura del fumo superiore a quella dell'aria, i risultati ottenuti potrebbero essere stati influenzati dalla generazione di eventuali moti convettivi. In ogni caso, prima di procedere con l'acquisizione delle immagini, si è atteso un tempo sufficientemente lungo perchè la quantità di *seeding* fosse adeguata ad un'analisi PIV e per permettere il raffreddamento del fumo.

È interessante osservare, infine, che le particelle in levitazione sono soggette a dei moti di *spinning* apparentemente casuali, che certamente influiscono sui campi di velocità raffigurati.

In ogni caso, i risultati ottenuti evidenziano la grande influenza delle caratteristiche fisiche degli oggetti levitati, quali dimensioni, forma e scabrezza della superficie, sui campi di velocità da essi indotti.

La levitazione della sfera di polistirolo genera campi di velocità più intensi e la direzione del flusso risulta essere fortemente influenzata dalla forma non perfettamente sferica del polistirolo e dalla maggiore scabrezza della sua superficie.

La goccia d'acqua, al contrario, induce campi di velocità che raggiungono valori massimi e minimi inferiori e il flusso curva attorno alla superficie sferica della goccia, seguendone la rotazione. La sostanziale differenza nell'intensità dei flussi di streaming nei due casi potrebbe essere spiegata da alcune osservazioni. La superficie dell'acqua è libera di muoversi, mentre per la superficie

del polistirolo è valida la condizione di non scorrimento a parete. Tale effetto potrebbe causare una variazione dello strato limite acustico sulla goccia d'acqua, riducendo i moti di streaming. Infine, i moti di rotazione del polistirolo, probabilmente indotti da momenti generati dalle forze di radiazione acustica, potrebbero contribuire a rafforzare il flusso.

Capitolo 5

Conclusioni

TinyLev è un levitatore acustico ad asse singolo non risonante a due emettitori. Le applicazioni della levitazione acustica sono molteplici e spaziano in numerosi ambiti.

La formazione dell'onda stazionaria, ai cui nodi è possibile sospendere piccoli oggetti grandi fino a 4 mm e con una densità massima di $2,2 \text{ g/cm}^3$, è permessa dalla presenza di due onde acustiche che viaggiano in direzioni opposte ad una frequenza di 40 kHz.

Il segnale, composto da due onde quadre con ampiezza di 5 V e frequenza di 40 kHz, viene generato da una scheda Arduino Nano e trasmesso ad un amplificatore L297N *Dual H-Bridge*, che a sua volta è collegato ai due emettitori.

La frequenza e l'ampiezza del segnale sono due parametri fondamentali, che regolano l'intensità del campo acustico e influenzano fortemente il funzionamento del levitatore.

L'impiego dell'oscilloscopio digitale *Digilent Analog Discovery 3* ha permesso, in una prima fase, di caratterizzare il segnale elettrico e studiarne la trasmissione e, in un secondo momento, modificando appositamente i collegamenti dei cavi, di usarlo in sostituzione della scheda Arduino. L'oscilloscopio, infatti, impiegando la funzione *WaveGen* del software *WaveForms*, permette di generare segnali con diverse forme d'onda e variabili in ampiezza e frequenza.

Dalla frequenza del segnale dipende la distanza tra i nodi dell'onda stazionaria, che dunque potrebbe essere modificata semplicemente variando questo parametro. Tuttavia gli speakers *Murata*, di cui sono costituiti i due emettitori, funzionano solamente nell'intorno dei 40 kHz, pertanto tale modifica non può essere applicata concretamente.

L'ampiezza, al contrario, fissa a 5 V nel segnale generato dalla scheda Arduino, può essere

modificata collegando il *Digilent*.

L'intensità del campo acustico dipende strettamente dal voltaggio del segnale. A voltaggi superiori corrisponde anche la capacità di levitare oggetti più densi. Si è visto che al diminuire del voltaggio, prima 5 V poi 4 V e 2 V, calava notevolmente l'intensità del campo acustico. Probabilmente il campo acustico generato a 2 V sarebbe stato in grado di levitare oggetti di densità inferiore.

La realizzazione di un'apposito setup ha permesso di acquisire il segnale acustico al variare dell'ampiezza del segnale in input, in varie posizioni del campo acustico. Ne è emerso, come ci si aspettava, che il campo acustico è più intenso nella zona centrale del levitatore, a prescindere dal voltaggio del segnale. Ciò è dovuto alla geometria degli emettitori, progettati per concentrare la forza di radiazione acustica nel centro del TinyLev. Inoltre, tutte le acquisizioni sono state comparate con quelle relative al campo acustico generato dal segnale della scheda Arduino, in modo da paragonarne l'intensità.

Per verificare la costanza del segnale emesso dall'oscilloscopio, sono state effettuate 10 acquisizioni nella stessa posizione, quella centrale, con ampiezza costante a 5 V e in seguito è stata eseguita un'analisi spettrale. I segnali sono risultati sostanzialmente identici, con lievi fluttuazioni nell'ampiezza.

E' stato interessante eseguire una caratterizzazione fluidodinamica degli effetti di streaming acustico sulla levitazione di una sfera di polistirolo e di una goccia d'acqua e confrontarne i risultati.

A tale scopo, è stato realizzato un setup ottico costituito da una *high speed camera*, un laser continuo, un sistema ottico di formazione del piano di luce laser e un box in plexiglas e cartone nero. Le immagini sono state acquisite ad una frequenza di 3700 Hz e in seguito analizzate con il metodo PIV, impiegando il software PIVlab per elaborarle e ottenere i campi di velocità medi su 925 frames, ovvero ad intervalli di 0.25 s, in un arco temporale di 1.25 s. I dati ottenuti sono stati rielaborati con MATLAB per ottenere delle *colormaps* che rendessero i risultati maggiormente comprensibili.

I risultati potrebbero essere stati influenzati da effetti di convezione naturale, causati dalla differenza di temperatura tra il fumo e l'aria, e da possibili errori di calcolo dovuti alla presenza di una fascia d'ombra generata dalla riflessione della luce laser sulla superficie delle particelle sospese. Questi effetti sono stati in ogni caso limitati: il primo attendendo il raffreddamento

del fumo per procedere con le acquisizioni, il secondo applicando filtri di validazione in fase di post-processing.

I campi di velocità osservati nella levitazione del polistirolo risultano avere valori massimi e minimi maggiori rispetto a quelli raggiunti nel caso della goccia e, in generale, i risultati ottenuti risultano dipendere fortemente dalla geometria e dalle caratteristiche degli oggetti levitati. Il flusso curva attorno alla goccia, seguendone il profilo sferico e la rotazione. Ciò invece non si verifica per il polistirolo, il cui campo di velocità indotto dallo streaming acustico è fortemente influenzato da una maggiore scabrezza superficiale e una geometria non perfettamente sferica.

Appendice A

Codici Matlab

A.1 Analisi comparativa

```
clear , close all ,

V = [2 , 4 , 5];
pos = { 'centro ' , 'centroalto ' , 'centrobasso ' ,
        'centrolatdx ' , 'centrolatsx ' };

for jj = 1:3
    figure(jj)
    for ii = 1:5
        namea = [ 'dati_inarduino_ ' , pos{ii} , '.csv ' ];
        name = [ 'dati_indigilent ' , num2str(V(jj)) , 'v_ ' , pos{ii} , '.csv ' ];
        dat = readtable(name);
        data = readtable(namea);
        subplot(2,3,ii)
        plot(data.Time_s_ , data.Channel1_V_ , dat.Time_s_ ,
             dat.Channel1_V_)
        xlabel('t (s)' , FontSize=20)
        ylabel('V (V)' , FontSize=20)
```

```

    title('Confronto segnali ',pos{ii},'FontSize',20)
    legend('Segnale Arduino 5V',[ 'Segnale WaveGen
    ',num2str(V(jj)),'V'],'FontSize',10)
    ax = gca;
    ax.XAxis.FontSize = 15;
    ax.YAxis.FontSize = 15;

    Va_pp(ii) = max(data.Channel1_V_)-min(data.Channel1_V_);
    V_pp(ii) = max(dat.Channel1_V_)-min(dat.Channel1_V_);

    phi(jj,ii) = V_pp/Va_pp;
    phi_name{jj,ii}=[name namea];
end
end
%%
%Matrice phi
% Prima riga: confronto di tutti i punti di misura con segnale digi 2 volt
% Seconda riga: confronto di tutti i punti di misura con segnale digi 4 volt
% Terza riga: confronto di tutti i punti di misura con segnale digi 5 volt
% Quindi ad esempio la colonna 1 rappresenta i phi della misurazione
% centrale, la seconda di quella centroalto e cosi via.
figure(4)
for k=1:5
    plot(V,phi(:,k),'s')
    hold on
    legenda{k} = [pos{k}];
end
    legend(legenda,FontSize=20)
xlabel('Voltaggio',FontSize=20)
ylabel('Phi',FontSize=20)
xticks([0 2 3 4 5]);
ax = gca;

```

```
ax.XAxis.FontSize = 20;
ax.YAxis.FontSize = 20;
```

A.2 Analisi spettrale

```
clear , close all ,

num_acq = 1:1:10;

%Scivo in una matrice le acquisizioni del canale 1 per ogni misurazione.
for jj = 1:length(num_acq)

    acq= [ 'dati_indigilent_ ', num2str(num_acq(jj)) , '_confronto ' , '.csv ' ];
    dat = readtable(acq);
    matrix(:,jj)=(dat.Channel1_V_);

end

time=(dat.Time_s_); %il tempo di acquisizione è lo stesso
per tutte le misurazioni

%%

[L, N] = size(matrix);      % L = n. campioni , N = n. segnali
dt = mean(diff(time));      % intervallo medio tra campioni
Fs = 1 / dt;                % frequenza di campionamento stimata

freq = Fs*(0:(L/2)) / L;    % asse delle frequenze fino a Fs/2
```

```

ampiezza = zeros(L/2+1, N);
fase = zeros(L/2+1, N);

for i = 1:N
    signal = matrix(:, i);
    Y = fft(signal);
    P2 = abs(Y / L);
    P1 = P2(1:L/2+1);
    P1(2:end-1) = 2 * P1(2:end-1);

    ampiezza(:, i) = P1;
    fase(:, i) = angle(Y(1:L/2+1));
end

% Plot spettro in ampiezza
figure;
hold on;
for i = 1:N
    plot(freq/1000, ampiezza(:, i));
    legend("Segnale " + string(1:N), 'Location', 'best', FontSize=17);
end
xlabel('Frequenza (kHz)', FontSize=20);
ylabel('Ampiezza (V)', FontSize=20);
title('Spettri in ampiezza dei 10 segnali', 'FontSize', 20);
grid on;
xlim([0 80])
ax = gca;
ax.XAxis.FontSize = 20;
ax.YAxis.FontSize = 20;

% Plot spettro di fase
figure;

```

```

hold on;
    flag=fase <0;
    fase(flag)=fase(flag)+2*pi;
for i = 1:N
    plot(freq/1000, fase(:, i)*(180/pi));
    legend("Segnale " + string(1:N), 'Location', 'best',FontSize=17);
end
xlabel('Frequenza (kHz)',FontSize=20);
ylabel('Fase (°)',FontSize=20);
title('Spettri di fase dei 10 segnali','FontSize',20);
grid on;
xlim([0 100])
legend
ax = gca;
ax.XAxis.FontSize = 20;
ax.YAxis.FontSize = 20;

```

A.3 Rielaborazione PIV, sfera di polistirolo

A.3.1 Colormaps con vettori

```
close all; clear; clc;

% Crea cartella per salvare le immagini
output_folder = 'immagini_campi_vel_polistirolo';
if ~exist(output_folder, 'dir')
    mkdir(output_folder);
end

file_list = dir('PIVlab_data_poli_*.mat');

for k = 1:length(file_list)
    file_name = file_list(k).name;
    data = load(file_name);
    [~, base_name, ~] = fileparts(file_name);

    num_str = regexp(base_name, '\d+$', 'match');
    if isempty(num_str)
        file_num = 'X';
    else
        file_num = num_str{1};
    end

    set(gcf, 'Color', 'w');
    scaling_factor = 3;
    x_mm = data.x * 1e3;
    y_mm = data.y * 1e3;

    r_sphere = 1.2;
```

```

th = linspace(0, 2*pi, 36);
x_sphere = r_sphere * cos(th) + 10;
y_sphere = r_sphere * sin(th) + 7.5;

components = {'u', 'v', 'modulo'};

for i = 1:length(components)
    comp = components{i};
    figure('Position', [100, 100, 1200, 1000]);

    switch comp
        case 'u'
            field = data.u_filtered;
            quivX = data.u_filtered * scaling_factor;
            quivY = zeros(size(data.u_filtered));
            label_cbar = 'Velocita u (m/s)';
        case 'v'
            field = data.v_filtered;
            quivX = zeros(size(data.v_filtered));
            quivY = data.v_filtered * scaling_factor;
            label_cbar = 'Velocita v (m/s)';
        case 'modulo'
            field = sqrt(data.u_filtered.^2 + data.v_filtered.^2);
            quivX = data.u_filtered * scaling_factor;
            quivY = data.v_filtered * scaling_factor;
            label_cbar = 'Modulo della velocita (m/s)';
    end

    contourf(x_mm, y_mm, field, 30, 'LineStyle', 'none');
    hold on;
    quiver(x_mm, y_mm, quivX, quivY, 0, 'k');
    patch(x_sphere, y_sphere, 'w');

```



```

xlim([4 16]);
ylim([2 14]);
clim([0 0.25]);
daspect([1 1 1]);

cb = colorbar;
cb.Label.String = label_cbar;
cb.Label.FontSize = 25;
cb.FontSize = 20;

set(gca, 'FontSize', 20);

xlabel('x (mm)', 'FontSize', 20);
ylabel('y (mm)', 'FontSize', 20);

filename = fullfile(output_folder, [base_name '_'
comp '_media.png']);
exportgraphics(gcf, filename, 'Resolution', 300);
close;
end
end

```

A.3.2 Colormaps senza vettori

```
close all; clear; clc;
fontsize = 12;

% Crea cartella per salvare solo le colormaps
output_folder = 'colormaps_3';
if ~exist(output_folder, 'dir')
    mkdir(output_folder);
end

file_list = dir('PIVlab_data_poli_*.mat'); % tutti i file

% Lista componenti da plottare
components = {'u', 'v', 'modulo'};

for c = 1:length(components)
    comp = components{c};

    % Crea figura con griglia 3 righe x 2 colonne
    figure('Units','centimeters','Position',[2, 2, 20, 18]);
    t = tiledlayout(3, 2, 'TileSpacing','compact','Padding',
        'compact');

    for k = 1:length(file_list)
        % Carica file
        file_name = file_list(k).name;
        data = load(file_name);
        [~, base_name, ~] = fileparts(file_name);

        % Coordinate in mm
        x_mm = data.x * 1e3;
```

```

y_mm = data.y * 1e3;

% Costruzione sfera
r_sphere = 1.2;
th = linspace(0, 2*pi, 36);
x_sphere = r_sphere * cos(th) + 10;
y_sphere = r_sphere * sin(th) + 7.5;

% Selezione campo
switch comp
    case 'u'
        field = data.u_filtered;
        label_cbar = 'Velocita' u (m/s)';
        clim_map = [-.14 .14];
    case 'v'
        field = data.v_filtered;
        label_cbar = 'Velocita' v (m/s)';
        clim_map = [-.2 .2];
    case 'modulo'
        field = sqrt(data.u_filtered.^2 + data.v_filtered.^2);
        label_cbar = 'Modulo della velocita' (m/s)';
        clim_map = [0 .25];
end

% Posizionamento tile
if k <= 4
    nexttile(k);
elseif k == 5
    nexttile([1 2]); % occupa tutta la riga in basso
else
    nexttile;
end

```

```

% Plot
contourf(x_mm, y_mm, field , 30, 'LineStyle', 'none');
hold on;
patch(x_sphere, y_sphere, 'w');
xlim([4 16]);
ylim([2 14]);
clim(clim_map);
daspect([1 1 1]);

% Colorbar
cb = colorbar;
cb.Label.String = label_cbar;
cb.Label.FontSize = fontsize;
cb.FontSize = fontsize;

set(gca, 'FontSize', fontsize);
xlabel('x (mm)', 'FontSize', fontsize);
ylabel('y (mm)', 'FontSize', fontsize);
%title(base_name, 'Interpreter', 'none', 'FontSize', fontsize);

% Etichetta (a), (b), ...
label = ['(' char('a' + k - 1) ')'];
text(0.02, 0.95, label, 'Units', 'normalized', 'FontSize', fontsize,
end

% Salva figura
filename = fullfile(output_folder, ['colormap_' comp '.png']);
exportgraphics(gcf, filename, 'Resolution', 300);
close;

```

A.4 Rielaborazione PIV, goccia d'acqua

A.4.1 Colormaps con vettori

```
close all; clear; clc;

% Crea cartella per salvare le immagini
output_folder = 'immagini_campi_vel_goccia';
if ~exist(output_folder, 'dir')
    mkdir(output_folder);
end

file_list = dir('PIVlab_data_goccia_*.mat');

for k = 1:length(file_list)
    file_name = file_list(k).name;
    data = load(file_name);
    [~, base_name, ~] = fileparts(file_name);

    num_str = regexp(base_name, '\d+$', 'match');
    if isempty(num_str)
        file_num = 'X';
    else
        file_num = num_str{1};
    end

    set(gcf, 'Color', 'w');
    scaling_factor = 5;
    x_mm = data.x * 1e3;
    y_mm = data.y * 1e3;

    r_sphere = 0.5;
    th = linspace(0, 2*pi, 36);
```

```

x_sphere = r_sphere * cos(th) + 9.75;
y_sphere = r_sphere * sin(th) + 5.75;

components = {'u', 'v', 'modulo'};

for i = 1:length(components)
    comp = components{i};
    figure('Position', [100, 100, 1200, 1000]);

    switch comp
        case 'u'
            field = data.u_original;
            quivX = data.u_original * scaling_factor;
            quivY = zeros(size(data.u_original));
            label_cbar = 'Velocita u (m/s)';
        case 'v'
            field = data.v_original;
            quivX = zeros(size(data.v_original));
            quivY = data.v_original * scaling_factor;
            label_cbar = 'Velocita v (m/s)';
        case 'modulo'
            field = sqrt(data.u_original.^2 + data.v_original.^2);
            quivX = data.u_original * scaling_factor;
            quivY = data.v_original * scaling_factor;
            label_cbar = 'Modulo della velocita (m/s)';
    end

    contourf(x_mm, y_mm, field, 30, 'LineStyle', 'none');
    hold on;
    quiver(x_mm, y_mm, quivX, quivY, 0, 'k');
    patch(x_sphere, y_sphere, 'w');
    xlim([5 15]);

```

```

ylim([2 10]);
clim([0 0.08])
daspect([1 1 1]);

cb = colorbar;
cb.Label.String = label_cbar;
cb.Label.FontSize = 25;
cb.FontSize = 20;

set(gca, 'FontSize', 20);

xlabel('x (mm)', 'FontSize', 20);
ylabel('y (mm)', 'FontSize', 20);

filename = fullfile(output_folder, [base_name '_'
comp '_media.png']);
exportgraphics(gcf, filename, 'Resolution', 300);
close;
end
end

```

A.4.2 Colormaps senza vettori

```
close all; clear; clc;
fontsize = 12;

% Crea cartella per salvare solo le colormaps
output_folder = 'colormaps_goccia';
if ~exist(output_folder, 'dir')
    mkdir(output_folder);
end

file_list = dir('PIVlab_data_goccia_*.mat'); % tutti i file

% Lista componenti da plottare
components = {'u', 'v', 'modulo'};

for c = 1:length(components)
    comp = components{c};

    % Crea figura con griglia 3 righe x 2 colonne
    figure('Units','centimeters','Position',[2, 2, 20, 18]);
    t = tiledlayout(3, 2, 'TileSpacing','compact','Padding','compact');

    for k = 1:length(file_list)
        % == Carica file ==
        file_name = file_list(k).name;
        data = load(file_name);
        [~, base_name, ~] = fileparts(file_name);

        % == Coordinate in mm ==
        x_mm = data.x * 1e3;
```



```

y_mm = data.y * 1e3;

% == Costruzione goccia ==
r_sphere = 0.5;
th = linspace(0, 2*pi, 36);
x_sphere = r_sphere * cos(th) + 9.75;
y_sphere = r_sphere * sin(th) + 5.75;

% == Selezione campo ==
switch comp
    case 'u'
        field = data.u_original;
        label_cbar = 'Velocita' u (m/s)';
        clim_map = [-0.02 0.05];
    case 'v'
        field = data.v_original;
        label_cbar = 'Velocita' v (m/s)';
        clim_map = [-0.08 0.06];
    case 'modulo'
        field = sqrt(data.u_original.^2 + data.v_original.^2);
        label_cbar = 'Modulo della velocita' (m/s)';
end

% == Posizionamento tile ==
if k <= 4
    nexttile(k);
elseif k == 5
    nexttile([1 2]); % occupa tutta la riga in basso
else
    nexttile;
end

```

```

% == Plot ==
contourf(x_mm, y_mm, field , 30, 'LineStyle', 'none');
hold on;
patch(x_sphere, y_sphere, 'w');
xlim([5 15]);
ylim([2 10]);
clim(clim_map);
daspect([1 1 1]);

% == Colorbar ==
cb = colorbar;
cb.Label.String = label_cbar;
cb.Label.FontSize = fontsize;
cb.FontSize = fontsize;

set(gca, 'FontSize', fontsize);
xlabel('x (mm)', 'FontSize', fontsize);
ylabel('y (mm)', 'FontSize', fontsize);

% == Etichetta (a), (b), ... ==
label = ['(' char('a' + k - 1) ')'];
text(0.02, 0.95, label, 'Units', 'normalized',
'FontSize', fontsize, 'FontWeight', 'bold');
end

% == Salva figura ==
filename = fullfile(output_folder, ['colormap_' comp '.png']);
exportgraphics(gcf, filename, 'Resolution', 300);
close;
end

```

Appendice B

Grafici

B.1 Grafici analisi comparativa

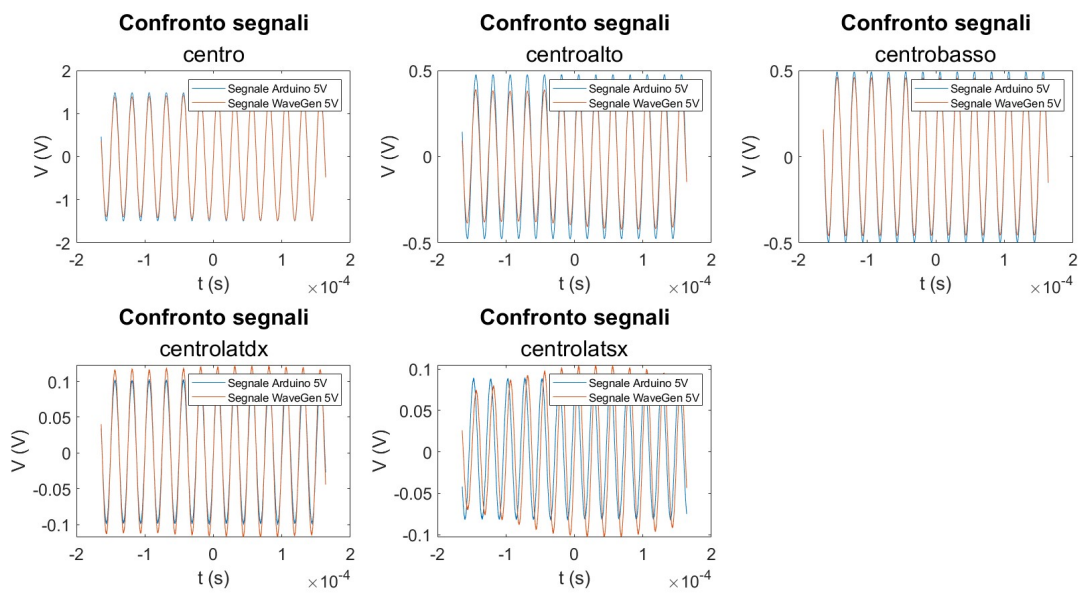


Figura B.1: Confronto segnale Arduino con segnale Digilent 5 V

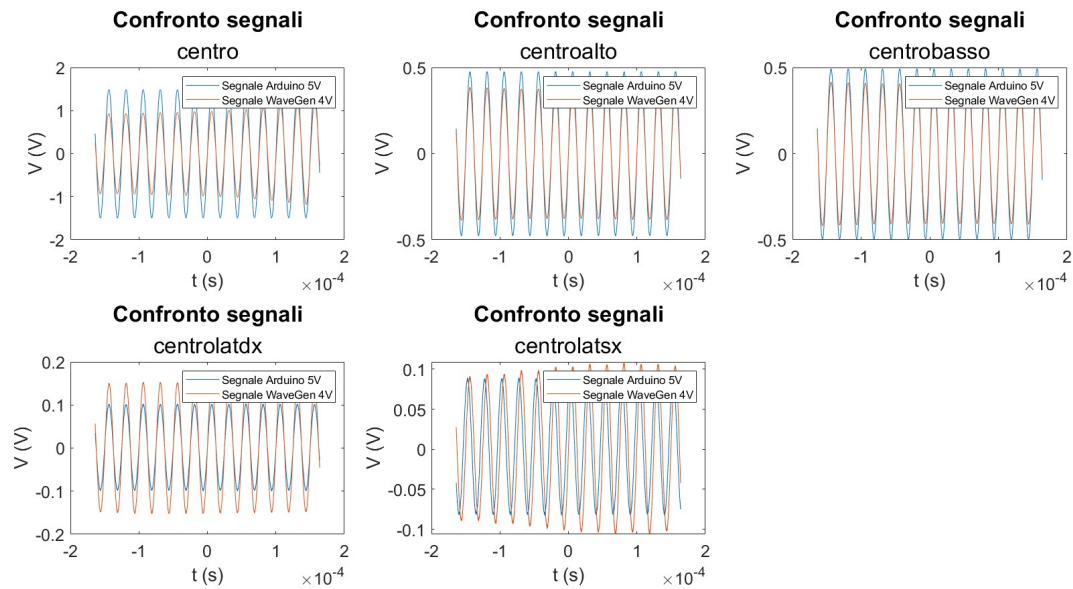


Figura B.2: Confronto segnale Arduino con segnale Digilent 4 V

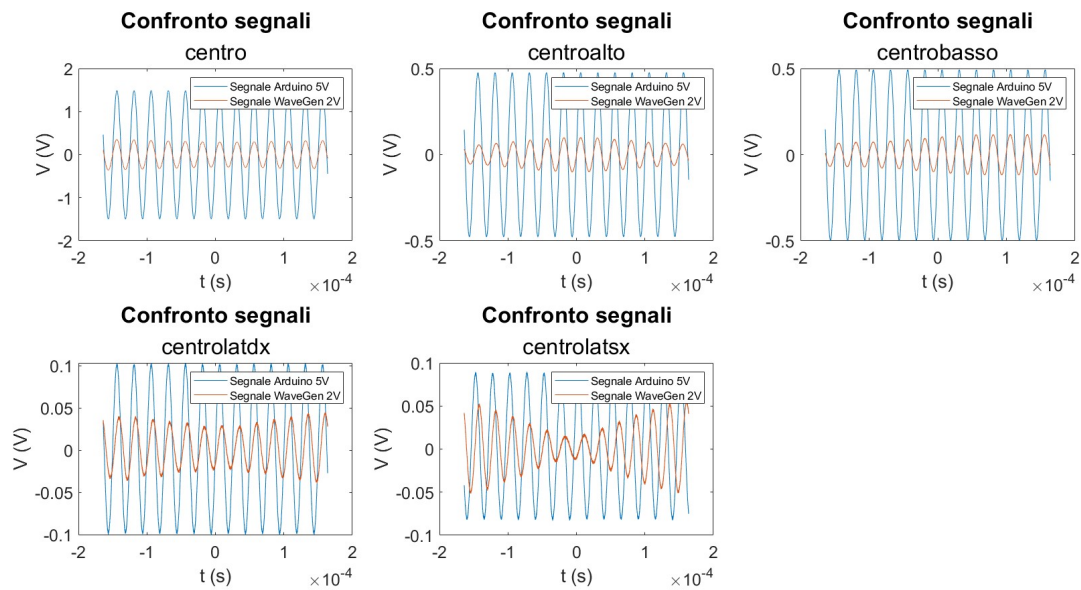


Figura B.3: Confronto segnale Arduino con segnale Digilent 2 V

Bibliografia

- [1] Raffel, Willert, Wereley; Kompenhans (2007). Particle Image Velocimetry (3-8, 14-21). Springer.
- [2] Marzo, Barnes, Drinkwater (2017). TinyLev: A multi-emitter single-axis acoustic levitator <https://doi.org/10.1063/1.4989995>
- [3] Fuqiang Zhang, Zelai Jin (2018). The Experiment of Acoustic Levitation and the Analysis by Simulation <https://www.scirp.org/journal/paperinformation?paperid=88024>
- [4] L. Gor’Kov (1962). On the forces acting on a small particle in an acoustical field in an ideal fluid (Soviet Physics Doklady, Vol. 6)
- [5] Murata Transducers [Datasheet](#)
- [6] THORLABS <https://www.thorlabs.com/>
- [7] PIVlab <http://pivlab.de/>
- [8] Digilent Analog Discovery 3 <https://digilent.com/reference/test-and-measurement/analog-discovery-3/reference-manual>
- [9] WaveForms <https://digilent.com/reference/software/waveforms/waveforms-3/reference-manual>
- [10] Software IdeaMaker <https://www.raise3d.com/ideamaker/>
- [11] Software PCC <https://www.phantomhighspeed.com/resourcesandsupport/phantomresources/pccsoftware>