

PIV image analysis

Claudio Caccia - 820091

27 febbraio 2017

Sommario

Analisi di coppie di immagini PIV relative al campo di moto intorno ad un profilo alare di tipo NACA 23012.

1 Introduzione

La *Particle Image Velocimetry* (PIV) è un metodo per la visualizzazione qualitativa e quantitativa del campo di moto di un fluido mediante l'uso di particelle traccianti illuminate da una lama di luce e fotografate in due istanti successivi. I dati rilevanti per l'esperimento sono i seguenti:

- profilo alare *NACA 23012* di corda *30cm*,
- velocità galleria: *30m/s*,
- dimensione delle immagini: 1024×1280 pixel *grayscale* (cfr. Figura 5)
- dimensione della finestra di analisi: *103 × 82mm*
- intervallo tra i fotogrammi: $\Delta t = 10\mu s$

L'analisi si compone di tre fasi: *preprocessing*, *correlazione* ed *analisi* dei risultati.

2 Preprocessing

Prima di procedere alla determinazione del campo di velocità è opportuno analizzare le caratteristiche delle immagini ed eventualmente migliorarne la qualità. Le tecniche di miglioramento possono riguardare:

- *Background subtraction*: tecnica utile ad eliminare lo sfondo comune alle immagini ed evidenziare solo il moto delle particelle. In genere serve una serie di immagini (almeno una decina) per poter eliminare lo sfondo in modo affidabile, pertanto non è stata impiegata qui.
- *Filtraggio*: varie tecniche utili a migliorare la distinzione tra particelle e sfondo e a ridurre il rumore di fondo.

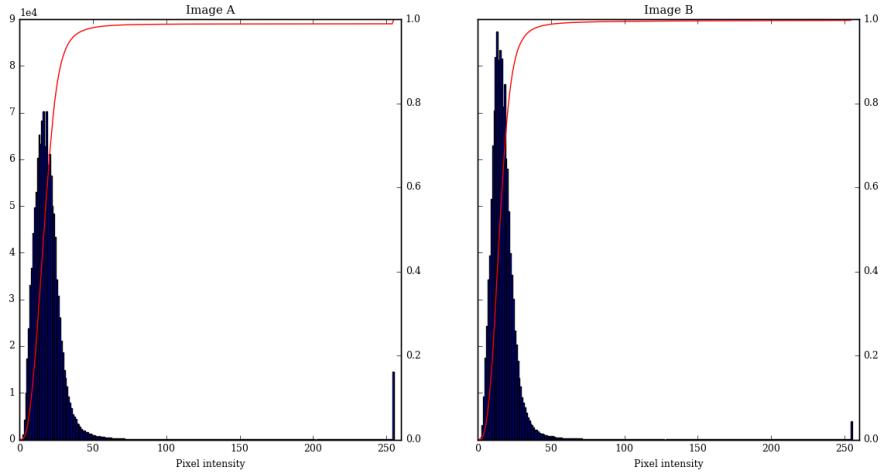


Figura 1: Istogrammi della distribuzione delle intensità di grigio

Un utile passo preliminare consiste nell’analizzare la distribuzione dell’intensità di grigio dei pixel, come illustrato in Figura 1. Si evidenzia come sia presente una netta separazione, in entrambe le immagini, tra lo sfondo (maggioranza dei pixel scuri) e le particelle ad intensità massima.

Per questo motivo si è deciso di utilizzare le immagini originali senza l’applicazione di filtri. Inoltre, alcune analisi preliminari hanno mostrato¹ come l’utilizzo di immagini filtrate possa in questo caso produrre un maggior numero di correlazioni spurie rispetto alle immagini originali, a parità di condizioni di test (cfr. ad es. Figura 6).

3 Algoritmo per PIV

Il campo di velocità è ottenuto tramite *correlazione* delle due immagini opportunamente traslate. I passi dell’algoritmo sono i seguenti:

3.1 Definizione dell’algoritmo di correlazione

Esistono diversi algoritmi [2] per effettuare la correlazione di due immagini. Sono stati considerati i seguenti:

Sum of Squared Differences [SSD] Consiste in un algoritmo computazionalmente semplice, la miglior correlazione si trova nel punto di *minimo* di:

$$R[x, y] = \sum_{i,j=-N/2}^{N/2} (A[i+x, j+y] - B[i, j])^2 \quad (1)$$

¹Test effettuati utilizzando semplici filtri a soglia o di tipo *min/max*[1]

Normalized Cross Correlation [NCC] [3],[4] Richiede il calcolo di media e deviazione standard delle immagini, la miglior correlazione si trova nel punto di *massimo* di:

$$R[x, y] = \frac{\sum_{i,j=-N/2}^{N/2} (A[i+x, j+y] - \mu_A) \cdot (B[i, j] - \mu_B)}{\sigma_A \cdot \sigma_B} \quad (2)$$

Discrete Fourier Transformation [DFT] Richiede il calcolo delle trasformate delle immagini, la miglior correlazione si trova nel punto di *massimo* di:

$$R = \mathcal{F}^{-1}\{\mathcal{F}\{A\} \circ \mathcal{F}\{B\}^*\} \quad (3)$$

3.2 Scelta delle finestre di interrogazione

Trascurando aspetti computazionali, la dimensione delle finestre di interrogazione comporta un *trade-off* tra una definizione accurata del campo di velocità, che richiede finestre di interrogazione piccole, e la bontà della correlazione, più affidabile con finestre grandi. I risultati più significativi sono stati ottenuti con finestre da **16** e **32** pixel.

Anche il massimo spostamento è un parametro da considerare: dai dati a disposizione, una velocità di $50m/s$ corrisponde ad uno spostamento di circa $6.24px$ pertanto si considerano spostamenti massimi di **8** pixel. Le finestre di interrogazione sono pertanto:

$$I = [8 + jN : 7 + (j+1)N, 8 + kN : 7 + (k+1)N] \quad N \in \{16, 32\} \quad j, k = 0 \dots \quad (4)$$

3.3 Interpolazione sub-pixel

La correlazione definisce solamente spostamenti di pixel interi. Per raffinare il calcolo è possibile utilizzare algoritmi di interpolazione, come ad esempio descritto in [6].

parabolic-fit estimator

$$\epsilon = \frac{R_{-1} - R_{+1}}{2(R_{-1} + R_{+1} - 2R_0)} \quad (5)$$

3 point gaussian-fit estimator

$$\epsilon = \frac{\ln(R_{-1}) - \ln(R_{+1})}{2(\ln(R_{-1}) + \ln(R_{+1}) - 2\ln(R_0))} \quad (6)$$

Le equazioni 5 e 6 sono da applicare in entrambe le direzioni.

Nel caso di **NCC** è stato adottato l'algoritmo di 5 in quanto immune a problemi di condizionamento legato ai logaritmi. Nel caso di **DFT**, l'algoritmo adottato [5] consente automaticamente di definire un livello di precisione *sub-pixel*.

3.4 Fitting del profilo

I dati geometrici a disposizione hanno permesso di effettuare il *fitting* (manuale) del profilo alare rispetto alle immagini a disposizione. Questo ha consentito di comprendere meglio il campo di moto (soprattutto rispetto alla scala dimensionale del profilo) e di escludere tutte le correlazioni di elementi interni all'ala. Il fitting migliore è stato trovato a 7.2 di inclinazione rispetto all'asse della fotocamera ed è illustrato in figura 7 dove si può anche visualizzare l'andamento della velocità.

4 Risultati

Di seguito si riportano i principali risultati ottenuti dall'analisi PIV.

4.1 SSD senza interpolazione sub-pixel

Pur utilizzando una stima della correlazione semplice, si ottengono risultati interessanti, con pochi vettori di velocità palesemente spuri (cfr. figura 2). L'unica zona difficile da correlare risulta essere la porzione visibile di intradosso, dove probabilmente vi è un afflusso non ottimale di particelle o il moto è particolarmente complesso. L'assenza di raffinamento *sub-pixel* evidenzia un andamento della velocità costituito da regioni uniformi. Adottando una finestra a $16px$ comporta molti risultati spuri nello strato limite, come evidenziato in figura 2(a).

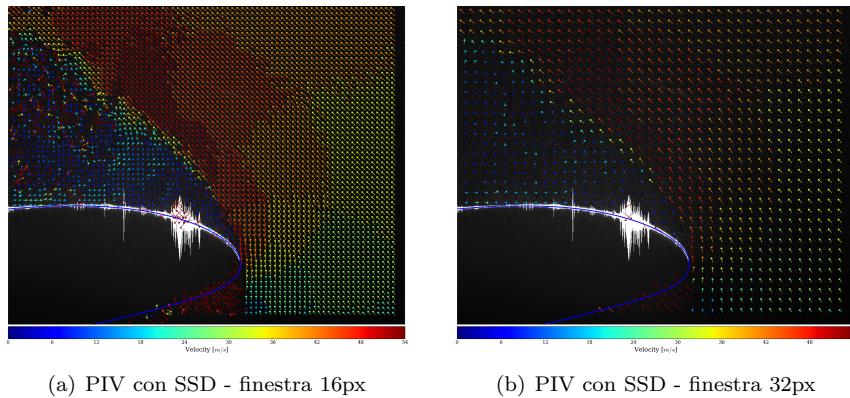


Figura 2: PIV con correlazione SSD

4.2 NCC con interpolazione sub-pixel

La correlazione secondo l'eq.2, unitamente all'interpolazione secondo l'eq.5, porta a risultati soddisfacenti, sia considerando finestre di interrogazione da $32px$ (figura 3(b)), in cui sono praticamente assenti correlazioni spuri, sia considerando finestre di interrogazione da $16px$ (figura 3(a)), in cui il campo di velocità è risolto molto bene, ad eccezione di pochi punti probabilmente spuri nello strato limite e la zona dell'intradosso.

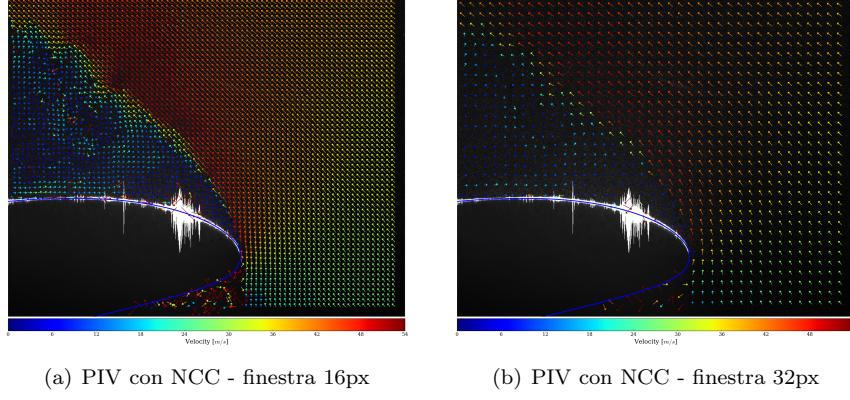


Figura 3: PIV con correlazione NCC

4.3 DFT

Seguendo l'esempio descritto in [7], è stata implementata la PIV utilizzando funzioni di libreria che consentono di determinare lo *shift* tra due immagini tramite *trasformata di Fourier* con upsampling per ottenere una precisione sub-pixel. Utilizzando una finestra di interrogazione da $32px$ (figura 4(b)) il risultato è sostanzialmente identico a NCC. Una finestra di interrogazione da $16px$ (figura 4(a)) comporta un elevato numero di vettori spuri nella regione al di fuori dello strato limite.

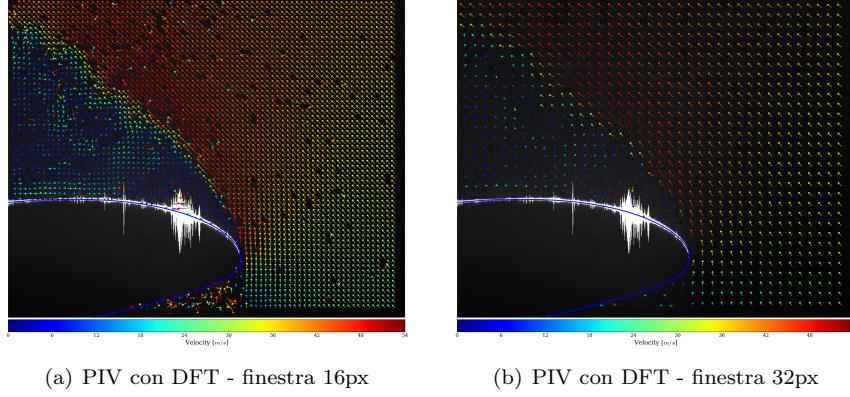


Figura 4: PIV con correlazione DFT

5 Post-Processing

Una volta determinato il campo di spostamenti in pixel è necessario eseguire alcune operazioni di *post-processing*.

Conversione : è necessario passare dagli spostamenti in pixel al valore di velocità, secondo la formula:

$$v [m/s] = \sqrt{\Delta x^2 + \Delta y^2} px \cdot \frac{1}{10 \cdot 10^{-6}s} \cdot \frac{0.082m}{1024px} \quad (7)$$

Il fattore di conversione di eq. 7 è stato utilizzato per tutte le figure presentate.

Dati spuri : tutti i dati di velocità spuri possono teoricamente essere sostituiti utilizzando i valori degli 8 elementi vicini, se ritenuti corretti. Il procedimento richiederebbe di identificare vettori molto diversi dai confinanti in termini di modulo e direzione e sostituirli con opportune medie. Poiché la miglior correlazione ottenuta (**NCC**) non presenta un numero sufficiente di dati palesemente spuri, si è deciso di non implementare questa operazione.

6 Codice di calcolo

Il linguaggio utilizzato per effettuare l'analisi è *Python* ed il codice implementato è visibile all'indirizzo https://github.com/Ccaccia73/FS_report nella cartella CODE.

Un esempio della sequenza di operazioni è visibile (non eseguibile) all'indirizzo http://nbviewer.jupyter.org/github/ccaccia73/FS_report/blob/master/code/PIV.ipynb

7 Confronto con OpenPIV

OpenPIV è un pacchetto software (<http://www.openpiv.net/>) open-source per l'analisi ed il post-processing di immagini PIV. È implementato in diversi linguaggi e fornisce diverse funzionalità per il filtraggio e l'analisi. Come ulteriore analisi e per valutare la qualità dei risultati ottenuti, sono state eseguite alcune analisi con la versione del programma in C++.

8 Figure Aggiuntive

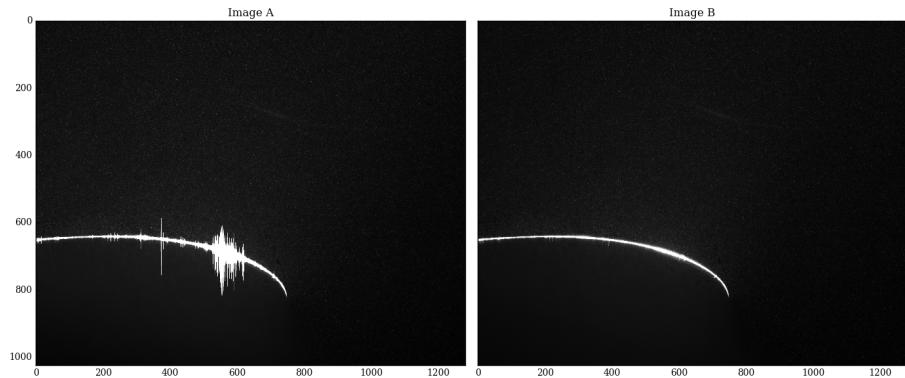


Figura 5: Immagini originali

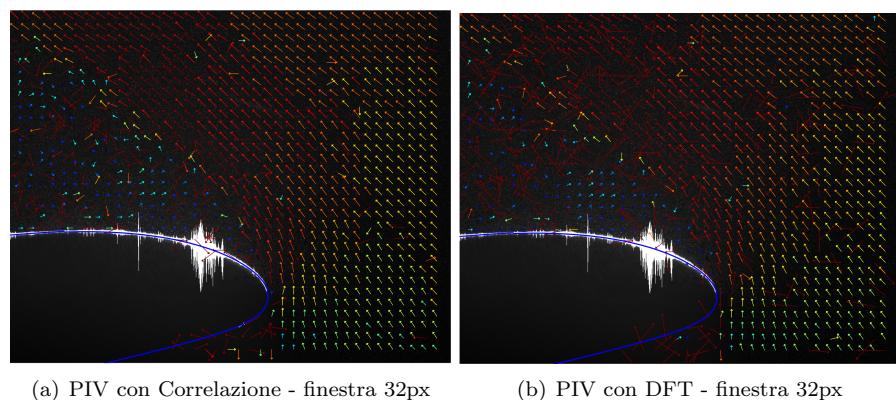


Figura 6: PIV su immagini filtrate

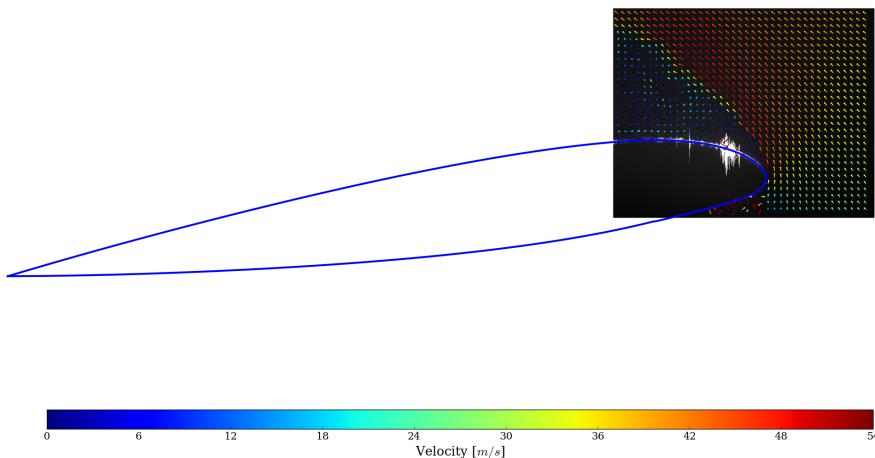


Figura 7: Profilo alare

Riferimenti bibliografici

- [1] Deen, Niels G., et al. "On image pre-processing for PIV of single-and two-phase flows over reflecting objects." *Experiments in fluids* 49.2 (2010): 525-530.
- [2] OpenCV object Detection, http://docs.opencv.org/2.4/modules/imgproc/doc/object_detection.html?highlight=matchtemplate#matchtemplate
- [3] Pan, Bing, and Kai Li. "A fast digital image correlation method for deformation measurement." *Optics and Lasers in Engineering* 49.7 (2011): 841-847.
- [4] Zhao, Feng, Qingming Huang, and Wen Gao. "Image matching by normalized cross-correlation." *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on.* Vol. 2. IEEE, 2006.
- [5] Guizar-Sicairos, Manuel, Samuel T. Thurman, and James R. Fienup. "Efficient subpixel image registration algorithms." *Optics letters* 33.2 (2008): 156-158.
- [6] Westerweel, Jerry. "Digital particle image velocimetry." Delft University (1993): 75-78.
- [7] Scikit-Image cross-correlation, http://scikit-image.org/docs/dev/auto_examples/transform/plot_register_translation.html
- [8] Liberzon, A., R. Gurka, and Z. Taylor. "Openpiv home page." (2009).