
**ALMA MATER STUDIORUM – UNIVERSITA' DI BOLOGNA
SEDE DI CESENA**

**SECONDA FACOLTA' DI INGEGNERIA CON SEDE A CESENA
CORSO DI LAUREA IN INGEGNERIA ELETTRONICA E DELLE
TELECOMUNICAZIONI**

**PROGETTO DI UN SENSORE DI POSIZIONE PER
APPLICAZIONI AERONAUTICHE BASATO SU TECNICHE
DI VISIONE ARTIFICIALE**

Tesi di Laurea in Controlli Automatici

Relatore: Prof. Ing. Paolo Castaldi

Candidato: Tommaso Dall'Arno

Correlatore: Ing. Nicola Mimmo

III sessione
anno accademico 2011/2012

Indice

0. Introduzione.....	pag. 3
1. Layout di sistema.....	pag. 7
1.1. Webcam USB.....	
1.2. Setup del sistema di acquisizione delle immagini.....	
1.3. Supporto hardware/software.....	
2. Sviluppo del codice.....	pag. 19
2.1. Acquisizione dell'immagine.....	
2.2. Pre-elaborazione.....	
2.3. Estrazione dell'informazione.....	
2.4. Calcolo del parametro di interesse.....	
3. Filtraggio adattativo dei dati.....	pag. 38
3.1. Caratteristica adattativa.....	
3.2. Processo di sintesi.....	
3.3. Considerazioni sulla frequenza di campionamento.....	
4. Risultati ottenuti.....	pag. 63
4.1. Analisi delle caratteristiche statiche.....	
4.2. Analisi dinamica.....	
5. Conclusioni.....	pag. 68

Capitolo 0

Introduzione

Nella presente tesi viene descritto lo sviluppo e l'implementazione di un sistema di acquisizione ed elaborazione delle immagini installato a bordo di un aeroplano con l'obiettivo di determinare la deflessione dell'aletta compensatrice, anche denominata trim tab.



Figura 0.1. Aeroplano di cui si vogliono identificare i parametri aerodinamici per la sintesi del sistema di controllo.

Il lavoro di tesi rientra all'interno di un progetto di ricerca nazionale, PRIN2008, volto alla creazione di innovativi algoritmi non lineari di identificazione, di stima e di controllo di un velivolo ultraleggero (figura 0.1), dell'Università di Bologna, denominato LA.U.R.A. (Laboratorio Universitario di Ricerche Aerospaziali) [1]. Uno dei passi fondamentali per la progettazione di leggi di controllo ad alte performance è la determinazione del modello matematico del velivolo mediante tecniche di identificazione. Queste ultime sono fondate sulla ricerca di relazioni

funzionali tra le variabili misurate del sistema in esame. Una volta identificate le variabili di ingresso e quelle di uscita, si procede ad una serie di esperimenti nei quali queste variabili vengono misurate. Nel caso di un velivolo, gli ingressi sono costituiti dall'angolo di deflessione delle superfici di comando (alettoni, timone, elevatore e trim tab). Le variabili di uscita sono invece le rotazioni del velivolo attorno ai tre assi principali.

Il trim tab, la cui deflessione è la grandezza che si intende stimare, è una piccola superficie posta al termine del piano di coda del velivolo che stabilizza l'assetto longitudinale senza la necessità da parte del pilota di eseguire continue correzioni con i controlli principali. Nell'aeroplano studiato la deflessione del trim tab viene modificata tramite un meccanismo (figura 0.3), posto nella parte superiore dell'abitacolo, che agisce su un cavo metallico che trasferisce il movimento all'aletta. Tipicamente la posizione del trim tab viene modificata dal pilota prima di determinate fasi del volo, come il decollo e l'atterraggio, e rimane costante durante questi periodi.



Figura 0.2. La piccola superficie rettangolare posta al termine del piano di coda, denominata trim tab.

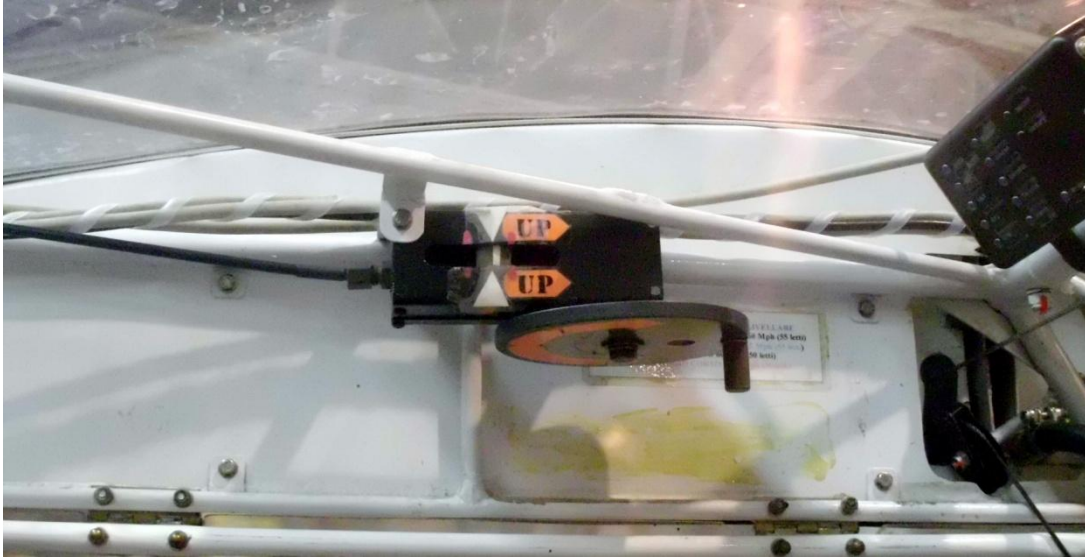


Figura 0.3. Meccanismo posto nella parte superiore dell'abitacolo, con cui il pilota modifica manualmente la deflessione del trim tab.

La necessità di disporre di una misura della deflessione del trim tab, unita alla difficoltà tecnica di realizzare tale stima tra due superfici in movimento pur garantendo i necessari livelli di sicurezza in volo, ha orientato il progetto verso la scelta di un mezzo semplice, leggero, senza contatti meccanici e potenzialmente polivalente, come una videocamera.

Analizzando più nel dettaglio il sistema di misura sviluppato, si distinguono tre parti fondamentali. Il ruolo della videocamera in questo caso è simile a quello del campionatore all'interno di un convertitore analogico-digitale, fornendo la successione di immagini contenenti l'informazione da ricavare. Tali immagini sono processate in tempo reale da algoritmi software con l'obiettivo di estrarne l'informazione richiesta in forma di segnale tempo-discreto monodimensionale. Infine, il segnale ottenuto viene elaborato in modo da attenuare la sua componente di rumore.

Nella figura 0.4 le tre successioni i_n , x_n ed y_n contengono rispettivamente le immagini acquisite, i valori di deflessione ottenuti tramite elaborazione, ed i valori filtrati.

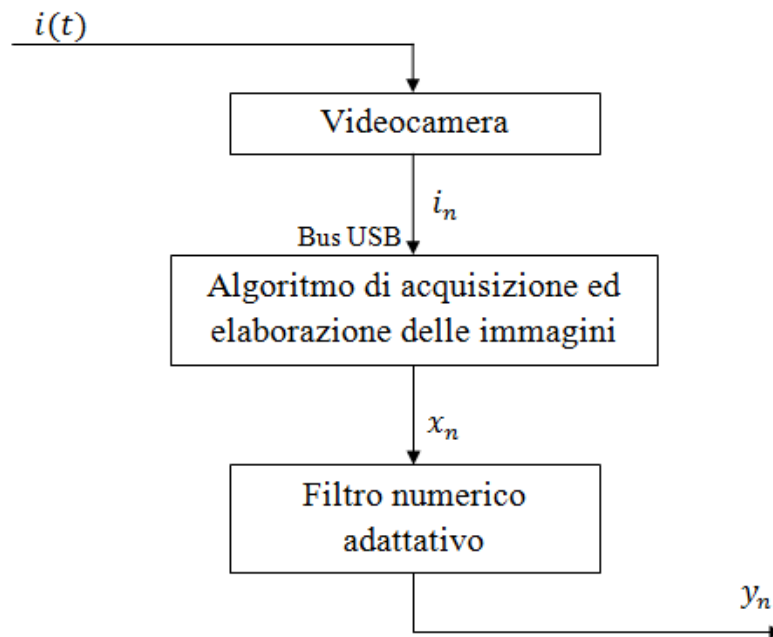


Figura 0.4. Suddivisione in blocchi funzionali del sistema implementato.

In questo testo saranno discusse le motivazioni che hanno portato alla scelta di un sistema di visione artificiale e verranno illustrati i contenuti dei tre sottosistemi che lo compongono.

In particolare nel cap. 1 sono messe a confronto alcune possibili soluzioni implementative per questa applicazione e si illustrano i dettagli della scelta adottata e del supporto hardware/software con cui è stata realizzata.

Il cap. 2 riguarda le procedure software applicate all'immagine acquisita per l'estrazione dell'informazione richiesta.

Nel cap. 3 è affrontato il problema del filtraggio dei dati ottenuti nel passaggio precedente. Si discute inoltre la scelta di un filtraggio di tipo "adattativo".

Il cap. 4 riassume il lavoro svolto fornendo alcuni risultati esemplificativi delle caratteristiche del sistema realizzato.

Nelle conclusioni sono riassunti i passi che hanno portato alla realizzazione del sistema ed è descritta un'ulteriore applicazione per un sistema di visione artificiale installato a bordo di un aeroplano.

Capitolo 1

Layout di sistema

La stima della deflessione di una superficie mobile può essere eseguita con diversi tipi di sensori. I più comuni sono gli encoder e i potenziometri: i primi sono spesso utilizzati nel caso in cui l'attuatore che agisce sulla posizione delle ali sia un motore elettrico, i potenziometri invece sono adatti alla stima della traslazione di un cavo, nel caso in cui il meccanismo di controllo sia meccanico.

In campo aeronautico la scelta del sistema di sensing deve essere influenzata da considerazioni sull'aspetto relativo alla sicurezza del volo. L'uso di sensori meccanicamente collegati alle linee di comando comporta un aumento del rischio di bloccaggio delle superfici mobili (es. potenziometro lineare calettato sui cavi di comando che va in blocking meccanico¹). Una soluzione che garantisce il minimo rischio di bloccaggio delle linee di comando del velivolo è data dall'assenza di contatto meccanico tra il sensore e le superfici mobili di cui si vuole determinare la posizione. Per questo motivo è giustificato l'utilizzo di una videocamera per il monitoraggio del movimento delle superfici di comando dell'aereo.

Si deve notare, inoltre, che, in alcuni casi, l'utilizzo di un sensore realizzato con tecniche di visione artificiale è sicuramente consigliabile, se non strettamente necessario come, ad esempio, nel caso della stima della deflessione delle ali per effetto del carico aerodinamico. In questo senso, un sistema di visione artificiale risulta avere potenzialmente molti più impieghi rispetto un sensore tradizionale (genericamente dedicato ad una sola misura). Un esempio di uno sviluppo futuro di questa applicazione è mostrato nel capitolo conclusivo.

¹ Un potenziometro lineare utilizzato come sensore fornisce un valore di resistenza elettrica proporzionale al valore della posizione relativa di due cilindri coassiali. Oltre a necessitare di manutenzioni come la lubrificazione o la rimozione di polveri, la deformazione meccanica di una struttura di questo tipo può causare il blocco della parte mobile e quindi l'impossibilità di agire su un comando di volo.

1.1 Webcam USB

L'impiego di una videocamera di tipo webcam USB è motivata da diversi fattori:

- Costo contenuto;
- Dimensioni ridotte;
- Utilizzo del protocollo USB 2.0;
- Supporto dei driver di National Instrument per l'utilizzo con LabView delle funzioni di acquisizione ed elaborazione delle immagini.

Senza dubbio devono essere presi in considerazione almeno altrettanti aspetti negativi; su tutti la modesta qualità del sensore di immagini e della parte ottica di una webcam commerciale. Periferiche video più performanti come le videocamere industriali possiedono sensori ottici con maggiore immunità al rumore, elevata risoluzione, obiettivi professionali² e alte frequenze di acquisizione delle immagini supportate da protocolli che realizzano bit rate elevati, ad esempio 1000Mbit/s nel caso del Gigabit Ethernet. In ogni caso per questa particolare applicazione l'utilizzo di una videocamera economica sembra comunque adeguato dato che non sono richiesti né frame rate né risoluzione particolarmente elevati.

Un'altra soluzione pensata per sistemi di visione artificiale è la cosiddetta "smart camera": un unico sistema che integra videocamera ed architettura dedicata hardware/software per l'esecuzione delle funzioni di visione. I vantaggi di tale struttura sono molteplici, tuttavia l'elaborazione richiesta in questa applicazione può essere comunque sostenuta da una unità centralizzata senza che le prestazioni del sistema vengano compromesse.

Le ridotte dimensioni e costi delle webcam USB sono dovuti soprattutto alla semplicità costitutiva dell'obiettivo e al sensore di immagini impiegato, tipicamente realizzato con tecnologia CMOS. Questo dispositivo sfrutta l'effetto

² Tali obiettivi, oltre a possedere caratteristiche qualitative migliori riguardo ad esempio i materiali costitutivi, offrono la possibilità di modificare alcuni parametri ottici come ad esempio lo zoom (obiettivi a lunghezza focale variabile).

fotoelettrico del silicio, ottenendo una struttura a matrice in cui l'elemento principale di ogni singola cella (o pixel) è analogo a un fotodiodo [2]. La maggior parte delle videocamere commerciali utilizza un sensore ottico APS (Active Pixel Sensor), in cui ogni pixel contiene tre transistori MOS. Il tempo di esposizione per ogni fotogramma è modulato tramite un transistor di reset, ottenendo così un sistema di otturazione elettronica. Il valore della carica accumulata in ogni pixel viene trasformato in un valore di tensione da un amplificatore e viene trasferito al bus attraverso un terzo transistor CMOS.

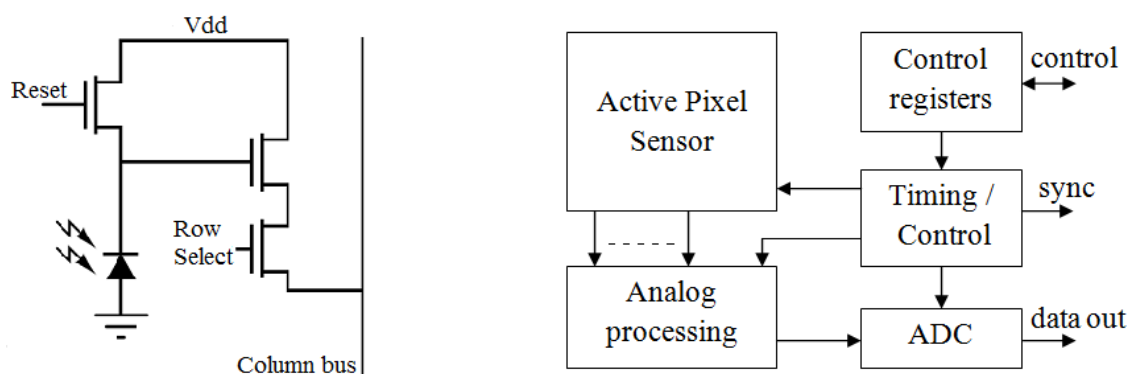


Figura 1.1. A sinistra: modello di un pixel di un sensore di immagini CMOS attivo. A destra: schema a blocchi di un tipico circuito integrato contenente un sensore ottico CMOS, il processo di digitalizzazione dell'immagine avviene all'interno del chip stesso. I segnali di controllo sono trasmessi con un particolare protocollo (es. I²C) per modificare alcune caratteristiche del processo di acquisizione delle immagini quali dimensione o tempo di esposizione.

Questo dispositivo ha il pregio di necessitare di una minima circuiteria esterna e di essere realizzabile con un processo standard, giustificando un costo decisamente contenuto. Tuttavia possiede una basso rapporto segnale rumore e una scarsa sensibilità alla luce dovuta ad un'area fotosensibile limitata dal fatto che sono necessari più transistori per ciascun pixel (basso fattore di riempimento), nonostante la tecnologia alla base del funzionamento di questo sensore sia in evoluzione. L'utilizzo di sensori con caratteristiche migliori come i CCD o i CMOS con microlenti, comporta un aumento dell'ingombro nel primo caso e una aumento di costo nel secondo.

L'utilizzo dello standard USB 2.0 come interfaccia tra calcolatore e videocamera, oltre a permettere l'impiego dei driver NI realizzati appositamente per

l'acquisizione di flussi video, influisce anche sulla configurazione hardware del sistema. La trasmissione dell'alimentazione tramite bus USB verso dispositivi a basso consumo³ permette un ridotto ingombro all'interno dell'abitacolo. Secondo le specifiche USB2.0 la modalità di trasferimento di un flusso di dati video è quella isocrona, per cui la totalità della banda richiesta (massima velocità di trasferimento teorica pari a 480Mbps, ridotta considerando l'overhead di protocollo) è a disposizione del device, che ha anche la massima priorità rispetto alle altre periferiche. Per questo tipo di funzionamento non è prevista una fase di handshake tra host (calcolatore) e device, perciò è massimo il rischio di errori di trasmissione, a fronte di una massima velocità di comunicazione.

1.2 Setup del sistema di acquisizione delle immagini

Il sistema di visione artificiale deve avere una struttura fisica (elettronica/ottica) che permetta l'acquisizione di immagini con la qualità sufficiente per poter estrarne l'informazione richiesta. Un ulteriore aspetto fondamentale è costituito dalle caratteristiche dell'ambiente in cui il sistema opera. In seguito vengono descritti i principali parametri che determinano le caratteristiche qualitative dell'immagine riprodotta dalla videocamera e che devono essere considerati in fase di setup del sistema di acquisizione [3].

- Risoluzione

Riguardo a un sensore di immagini esistono due concetti distinti di risoluzione. La "pixel resolution" o "sensor resolution" consiste nel numero minimo di pixel necessari per rappresentare il soggetto in esame senza perdere l'informazione rappresentata dai suoi dettagli. Tale valore può essere determinato considerando il particolare più piccolo da ispezionare. E' consigliato fare in modo che almeno due pixel rappresentino il dettaglio di dimensioni più ridotte. Se L è la dimensione massima dell'oggetto da rappresentare, e l è la lunghezza del più piccolo dettaglio, il valore della pixel resolution può essere determinato dall'equazione:

³ L'assorbimento medio di potenza di un sensore ottico per applicazioni commerciali è pari a circa 100mW in condizioni di funzionamento al massimo frame rate, tipicamente 30fps.

$$pixel\ resolution = \frac{L}{l} \times 2 \quad (1.1)$$

Se il numero di pixel utilizzati per rappresentare l'immagine è uguale al valore pixel resolution allora l'oggetto da ispezione occuperà l'intero campo visivo della videocamera.

Il secondo tipo di risoluzione riguarda invece la quantità di dettagli che la videocamera può riprodurre. Tale concetto di risoluzione, più generale del primo, influisce profondamente sulle prestazioni del sistema di visione artificiale: una scarsa risoluzione genera immagini sfocate e senza dettagli. Tre fattori incidono sulla risoluzione: ampiezza di campo, dimensioni e numero di pixel del sensore. Note queste caratteristiche è possibile calcolare la lunghezza focale dell'obiettivo necessaria per ottenere la massima risoluzione.

- Ampiezza di campo

L'ampiezza di campo, pari alla superficie riprodotta dall'inquadratura, è un parametro strettamente legato alla pixel resolution, infatti coincide con il parametro L nel caso in cui il soggetto occupi l'intero campo visivo e la pixel resolution assume il valore minimo.

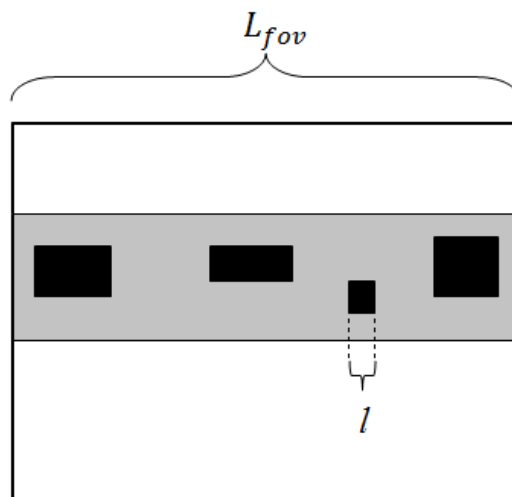


Figura 1.2. In figura l'oggetto da rappresentare (in grigio) occupa l'intero campo visivo. Il valore l è la dimensione del più piccolo dettaglio del soggetto inquadrato. Il numero minimo di pixel è definito dall'equazione $L_{fov}/l \times 2$.

- Dimensioni e numero di pixel del sensore

Il valore di superficie fotosensibile del sensore ottico è un parametro che determina l'ampiezza di campo, che, come detto prima, è un elemento determinante nella determinazione del numero minimo di pixel necessari per riprodurre tutti i dettagli del soggetto inquadrato. La dimensione del sensore è tipicamente indicata con la lunghezza della diagonale della superficie fotosensibile. Il numero di pixel del sensore deve essere uguale o maggiore del valore pixel resolution ottenuto con l'equazione (1.1).

- Lunghezza focale

Questo è il principale parametro che caratterizza l'obiettivo. Dalla lunghezza focale dipende la capacità di messa a fuoco della lente. Viene definita come la distanza tra il centro ottico della lente e il piano focale nel caso in cui l'oggetto da mettere a fuoco sia posto all'infinito. In questo caso il piano focale coincide con il piano fotosensibile del sensore.

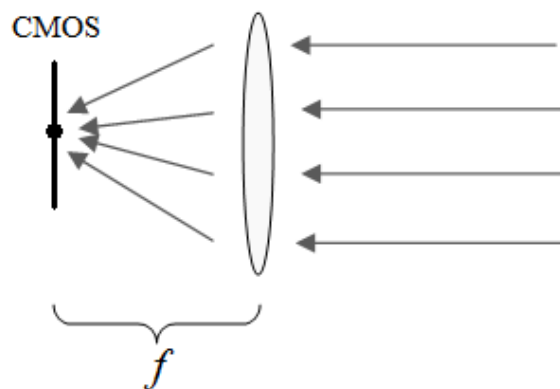


Figura 1.3. Rappresentazione della distanza focale f secondo la definizione. Dalla figura si nota che i raggi luminosi entrano nella lente convergente in modo parallelo dato che l'oggetto messo a fuoco è posto a distanza infinita.

Esiste una relazione tra lunghezza focale, distanza di lavoro, dimensioni del sensore e ampiezza di campo. Riprendendo il caso di figura 1.2, definendo l_{CMOS} la larghezza del sensore di immagini, d la distanza tra l'oggetto inquadrato e l'obiettivo, la lunghezza focale f vale:

$$f = \frac{d \cdot l_{CMOS}}{L_{fov} + l_{CMOS}} \quad (1.2)$$

Questo valore, espresso in millimetri, può talvolta essere modificato manualmente in modo da far valere l'equazione soprascritta, ma può anche essere controllato in modo automatico.

La struttura del sistema realizzato prevede una distanza tra la videocamera e i markers di circa 60cm, una ampiezza di campo pari a 20cm e il sensore ottico ha larghezza 3,4m, perciò la distanza focale opportuna risulta circa 10mm. La messa a fuoco è effettuata agendo su una ghiera posta sull'obiettivo.

- Profondità di campo

La profondità di campo è l'intervallo dei valori delle distanze dalla lente per cui l'oggetto ripreso risulta avere una sfocatura impercettibile, o comunque accettabile. Nonostante questo parametro non sia di fondamentale importanza per questa applicazione, in cui tutti i markers sono posti a distanze circa uguali dall'obiettivo, va comunque notato che i valori di d che mantengono l'immagine riprodotta sufficientemente a fuoco appartengono a un intervallo in cui il valore calcolato dalla 1.2 è solo un elemento.

- Luminosità e contrasto

Un'importante caratteristica visiva di un'immagine è l'intensità media dei pixel che la compongono. Valori elevati o troppo bassi di questo parametro causano saturazione, che si manifesta in un eccessivo numero di pixel bianchi o neri, con conseguente possibile perdita dell'informazione contenuta nell'immagine. Nel setup di un sistema di visione artificiale deve essere preso in considerazione il problema dell'illuminazione dell'oggetto in esame. Esistono in commercio diversi tipi di apparati di illuminazione, ad esempio array di LED che circondano l'obiettivo e mantengono un determinato livello di illuminazione uniforme.

Il contrasto è un altro fattore determinante della qualità dell'immagine riprodotta. Consiste nella variazione di intensità tra l'oggetto sotto ispezione e lo sfondo. Il sistema deve garantire valori di contrasto sufficientemente elevati da rendere

possibile la distinzione dell'oggetto con lo sfondo. Considerando per semplicità un'immagine in scala di grigi, il valore di contrasto può essere quantificato considerando I_{max} e I_{min} i valori di intensità massimo e minimo nell'immagine:

$$Contrasto = \frac{I_{max} - I_{min}}{I_{max} + I_{min}} \in [0,1]$$

I valori di luminosità e di contrasto dell'immagine riprodotta sono ovviamente legati alle condizioni di luminosità ambientale. In questa applicazione tale condizione è variabile e non può essere predeterminata con precisione, quindi diventa necessario mantenere le caratteristiche visive accettabili con algoritmi software. Anche questo aspetto verrà approfondito nel capitolo 2.

- Prospettiva

Errori di prospettiva insorgono quando l'asse ottico della videocamera non è perpendicolare alla superficie dell'oggetto inquadrato. Ciò causa nell'immagine riprodotta variazioni delle dimensioni di dettagli posti a diverse distanze dalla lente e quindi distorsione, intesa come alterazione del contenuto informativo dell'immagine. Questo difetto compromette le funzionalità dei sistemi di visione in cui l'elaborazione effettuata necessita di una trasformazione di lunghezze misurate in pixel in valori espressi con unità di misura più facilmente interpretabili (es. millimetri). Nel caso in cui non si possa installare la videocamera perpendicolarmente al soggetto da visualizzare, questo tipo di errore può essere ridotto tramite un processo di calibrazione eseguito in fase di inizializzazione dell'applicazione software che implementa le funzioni di visione artificiale. La calibrazione del sistema viene illustrata nel capitolo 2.

1.3 Supporto hardware/software

L'aeroplano di cui si vogliono identificare i parametri aerodinamici è stato equipaggiato con una strumentazione che permette l'elaborazione di una grande quantità di dati prodotti dai sensori installati⁴.

⁴ Oltre al sistema in esame, sono presenti sensori inerziali (giroscopi e accelerometri), sensori di velocità, altitudine, e un ricevitore GPS.

- Componente hardware

Per la realizzazione del sistema di misura si dispone di una architettura hardware modulare PXI (PCI eXtension for Instrumentation) montata a bordo del velivolo. Tale dispositivo possiede caratteristiche meccaniche, elettriche e software tali da garantire l'acquisizione e la manipolazione di una grande quantità di dati in un ambiente caratterizzato da intense vibrazioni meccaniche, rumore elettromagnetico e rapide variazioni di temperatura e umidità.

La piattaforma PXI è costituita da un controller e da diverse unità modulari rimovibili che aggiungono interfacce i/o, il tutto alloggiato a rack in uno chassis con supporto per l'alimentazione e per la comunicazione tra le slot.

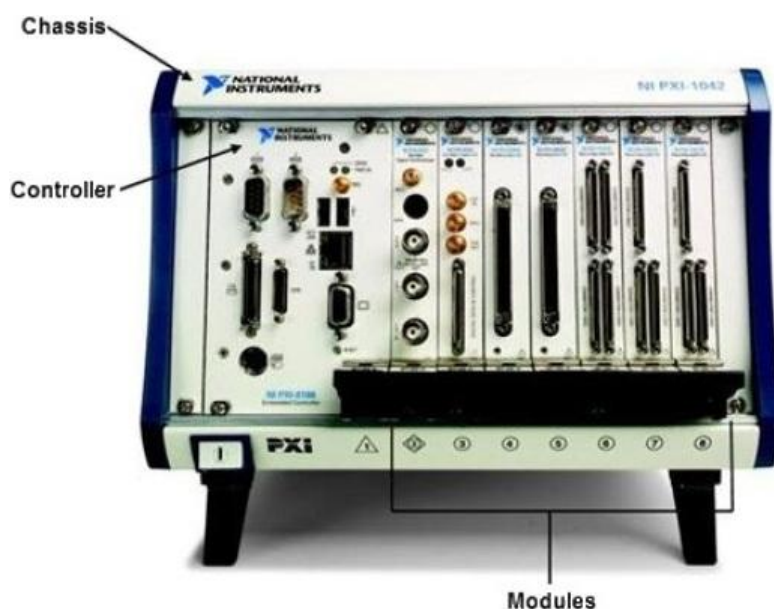


Figura 1.4. Chassis PXI standard contenente il controller (a sinistra) e sette moduli. Nel controller si distinguono due porte USB, una delle quali verrà utilizzata per questa applicazione.

Lo chassis garantisce robustezza al sistema e implementa lo standard hardware PCI aggiungendo funzionalità di sincronizzazione nella comunicazione tra i moduli e il controller. Nel controller risiede il processore, la memoria, e le principali periferiche I/O, come il disco fisso e le porte USB. La compatibilità con il sistema operativo è garantita con Labview Real-Time, sistema operativo realizzato da NI che offre prestazioni deterministiche in applicazioni di acquisizione dati e controllo.

- Sistema operativo real-time

Il sistema operativo real-time aggiunge alle caratteristiche di un sistema operativo general-purpose alcune funzionalità strettamente necessarie nella gestione di un processo come il controllo del volo di un aeroplano in cui è necessario ottenere una risposta del sistema in un tempo predeterminato e tipicamente molto breve. La principale proprietà implementata è dunque il determinismo rispetto al tempo di esecuzione delle funzioni di acquisizione ed elaborazione dei dati dai sensori installati nel velivolo: deve essere garantito che un task termini entro un intervallo di tempo fissato, denominato deadline. Per l'applicazione in esame questo tempo potrebbe essere fissato come l'inverso della massima frequenza di acquisizione delle immagini. Si vedrà più avanti che questo vincolo non è imprescindibile. Una precisa temporizzazione dell'esecuzione delle attività svolte dal calcolatore influisce positivamente sulle prestazioni del sistema di controllo, quindi anche sulla sicurezza del volo.

- Software applicativo LabView e librerie per la visione

Labview è un linguaggio di programmazione grafico compilato il cui codice sorgente, strutturato come un diagramma di flusso, è chiamato codice G. Nel programma eseguibile, con estensione .vi, l'elaborazione dei dati attraverso i nodi del codice determina l'ordine di esecuzione delle funzioni.

E' possibile aggiungere alle funzioni di programmazione di base le procedure per la gestione dell'acquisizione ed elaborazione delle immagini inviate dalla videocamera.

NI IMAQ (Image Acquisition) è il nome di un set di funzioni, o Application Program Interface (API), per il controllo tramite interfaccia grafica della gestione del processo di acquisizione e memorizzazione di immagini da videocamere NI [4]. La ricezione di flussi video da videocamere universali (es. webcam USB) è supportata dal software IMAQdx. Le funzionalità implementate vanno dall'inizializzazione delle caratteristiche video dell'immagine all'acquisizione ad alta velocità. L'ottimizzazione dell'utilizzo delle risorse è garantito dal supporto per il DMA (Direct Memory Access) che massimizza la velocità di ricezione dei dati provenienti dalla periferica, e dall'utilizzo di librerie caricate in modo

dinamico così da ridurre l'occupazione di memoria principale durante l'esecuzione del vi.

Nell'architettura NI IMAQ viene utilizzato un livello di astrazione hardware per garantire le stesse funzionalità su qualsiasi tipo di dispositivo di basso livello.

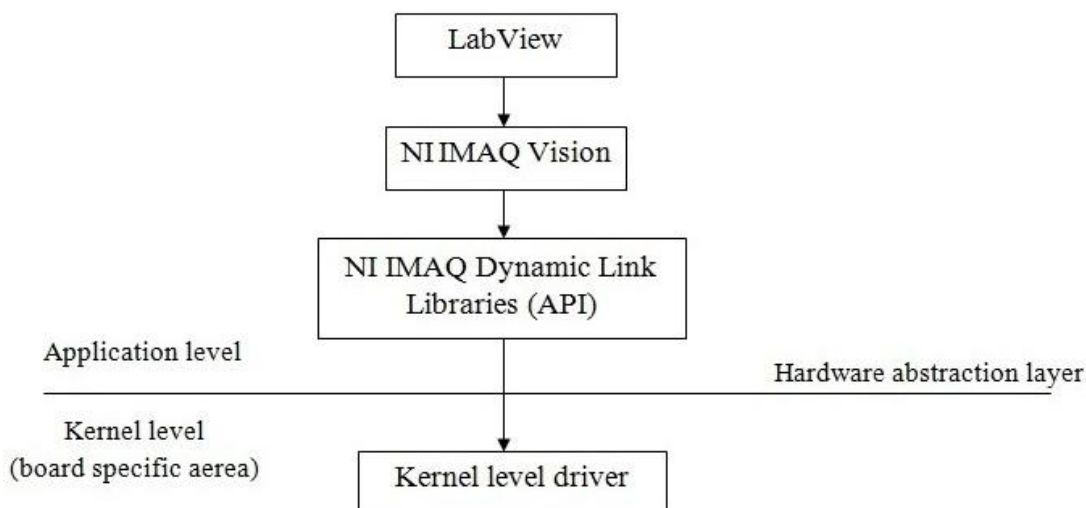


Figura 1.5. Architettura dell'NI IMAQ.

I driver NI IMAQdx sono invece necessari per permettere all'applicativo Labview l'accesso ai driver di webcam USB⁵ che rispettano le specifiche DirectShow⁶ sulla gestione del trasferimento dei dati [5]. In questo modo, oltre all'acquisizione di un flusso video, è anche possibile modificare dall'ambiente Labview alcune impostazioni di setup della videocamera, quali luminosità, contrasto, tempo di esposizione o zoom digitale.

⁵ Sono supportate anche videocamere basate su protocollo FireWire IEEE 1394 e GigaBit Ethernet.

⁶ DirectShow è un set di procedure realizzato da Microsoft per la gestione del flusso dati audio/video. L'interfaccia DirectShow è supportata dalla maggior parte dei driver di webcam attualmente in commercio.

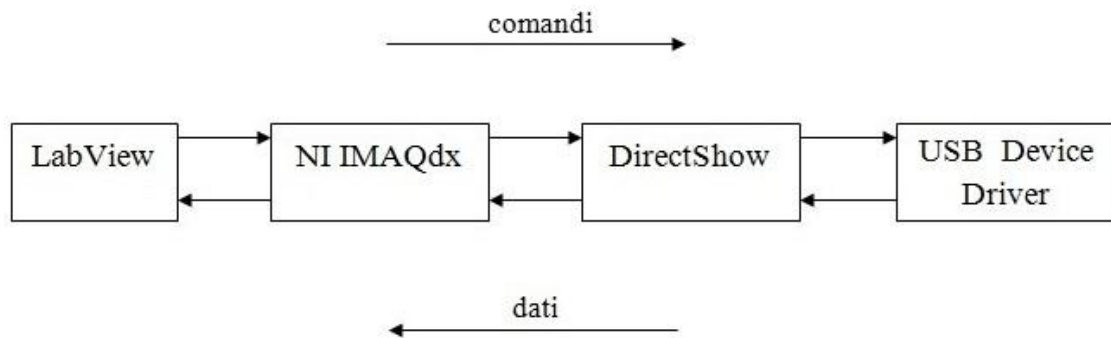


Figura 1.6. Sistemi software necessari per il collegamento con la videocamera USB.

Le funzioni di elaborazione delle immagini sono fornite da NI Vision [6], una libreria software che aggrega queste funzioni in tre sottogruppi:

- “Vision Utilities”: procedure per interagire con l’immagine acquisita, ad esempio ricavarne le dimensioni, mostrarla su video, definire regioni di interesse o sovrapporre all’immagine dei riferimenti.
- “Image Processing”: funzioni per l’analisi ed elaborazione dell’immagine. E’ possibile ad esempio eseguire un’operazione su ciascun pixel.
- “Machine Vision”: insieme degli algoritmi che caratterizzano un sistema di visione artificiale. Con queste funzioni viene effettivamente realizzata l’estrazione delle informazioni richieste dall’immagine. Esempi tipici sono l’identificazione e il conteggio di oggetti, o la localizzazione di bordi.

Nel capitolo successivo è descritta più approfonditamente l’elaborazione eseguita dalle funzioni di visione utilizzate in questa applicazione.

Capitolo 2

Sviluppo del codice

L'obiettivo del programma realizzato in ambiente Labview è quello di acquisire le immagini del meccanismo di controllo del trim tab, fornite dalla videocamera, ed estrarne l'informazione necessaria per la determinazione della deflessione dell'aletta compensatrice. Ciò avviene stimando la posizione, relativa rispetto a certi riferimenti fissi, del marker solidale al cavo che trasmette il movimento al trim tab. Sotto viene mostrata un'immagine analoga a quelle riprese dalla videocamera. I due triangoli bianchi vengono utilizzati come markers di riferimento in base ai quali si stima la posizione del marker mobile rettangolare.



Figura 2.1. Sistema di controllo manuale della deflessione del trim tab.

Il codice è strutturato in cinque moduli, collegati in cascata, cioè in cui l'elaborazione implementata da ciascuno è funzionale a quella del successivo:

1. Acquisizione delle immagini;
2. Pre-elaborazione delle immagini;
3. Estrazione dell'informazione, algoritmo di localizzazione dei marker;
4. Calcolo dello spostamento relativo del marker mobile;
5. Filtraggio dei dati.

Lo svolgimento sequenziale delle funzioni implementate dai sottosistemi sopraelencati avviene ad ogni immagine acquisita, dunque l'esecuzione del codice viene ripetuta con un loop infinito all'interno di un sistema software che permette di soddisfare i requisiti di determinismo richiesti per applicazioni aeronautiche. In particolare, tutto il codice è contenuto in un unico ciclo while, escluse le necessarie inizializzazioni dei dati. In seguito vengono discussi i dettagli implementativi di ciascun punto. La realizzazione del filtraggio dei dati, ultimo passo nella catena di elaborazione software, verrà approfondita nel capitolo 4.

2.1 Acquisizione dell'immagine

Una delle operazioni di inizializzazione effettuate prima del ciclo while consiste nell'allocazione della memoria necessaria per contenere un'immagine a livelli di grigio con codifica a 8bit. La funzione IMAQ Create, con l'opzione Grayscale U8, riserva un byte di memoria per ciascun pixel dell'immagine ricevuta. L'immagine riprodotta viene trattata come raster, cioè rappresentate in memoria con una matrice di valori, ciascuno dei quali rappresenta il livello di grigio del singolo pixel. Il numero di pixel contenuti in ogni immagine acquisita è imposto dalla risoluzione del sensore ottico e vale $800 \times 600 = 480'000$, anche se i bordi esterni dell'immagine non verranno considerati nelle successive elaborazioni software per evitare errori di misura dovuti alla distorsione introdotta dall'obiettivo nelle parti più esterne dell'immagine.

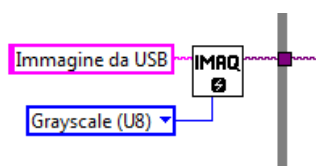


Figura 2.2. Icona IMAQ Create per l'allocazione in memoria di un'immagine in scala di grigi in cui il valore di ciascun pixel è codificato con un byte.

La scelta della codifica in scala di grigi a 8 bit di profondità garantisce un ridotto peso computazionale nelle successive procedure di equalizzazione dell'immagine e di localizzazione dei marker. Il blocco Vision Acquisition realizza l'interfaccia utente per l'utilizzo delle funzioni NI IMAQ per la definizione delle specifiche riguardo l'acquisizione delle immagini.

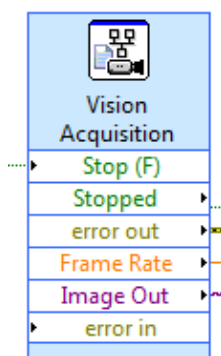


Figura 2.3. Icona della utility Vision Acquisition, inserita all'interno del ciclo while. Il valore di frame rate sarà un parametro necessario nelle successive elaborazioni.

Nel caso di una elaborazione troppo lenta delle immagini acquisite si può verificare un rallentamento del tempo di esecuzione del programma di acquisizione delle immagini. Una possibile soluzione prevede l'uso di buffer di memoria in cui immagazzinare le immagini in attesa di elaborazione. Per ovviare alla costruzione di buffer di memoria troppo grandi, National Instrument mette a disposizione la possibilità di considerare solo l'ultima immagine acquisita. In questo caso la frequenza di acquisizione delle immagini diminuisce quando il tempo di esecuzione dell'algorithmo supera il valore $1/f_s$.

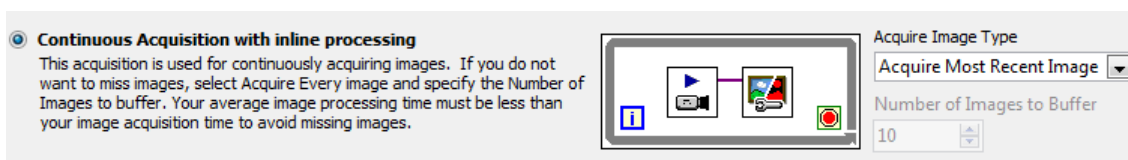


Figura 2.4. Impostazione del tipo di acquisizione tramite il menu del Vision Acquisition.

Con Vision Acquisition è inoltre possibile inviare comandi alla webcam in modo da modificare alcune caratteristiche dell'immagine ricevuta, come la luminosità, il contrasto o lo zoom digitale. Questo sarebbe un notevole vantaggio se le condizioni di illuminazione si mantenessero costanti durante il funzionamento del sistema, perché permetterebbe di migliorare la qualità dell'immagine acquisita, senza doverla ulteriormente elaborare. Purtroppo però i comandi possono essere trasmessi ai driver della webcam solo in fase di setup e non possono essere inviati durante il funzionamento della videocamera. Dato che le

condizioni di illuminazione variano durante le fasi di volo, è comunque necessaria una elaborazione delle immagini.

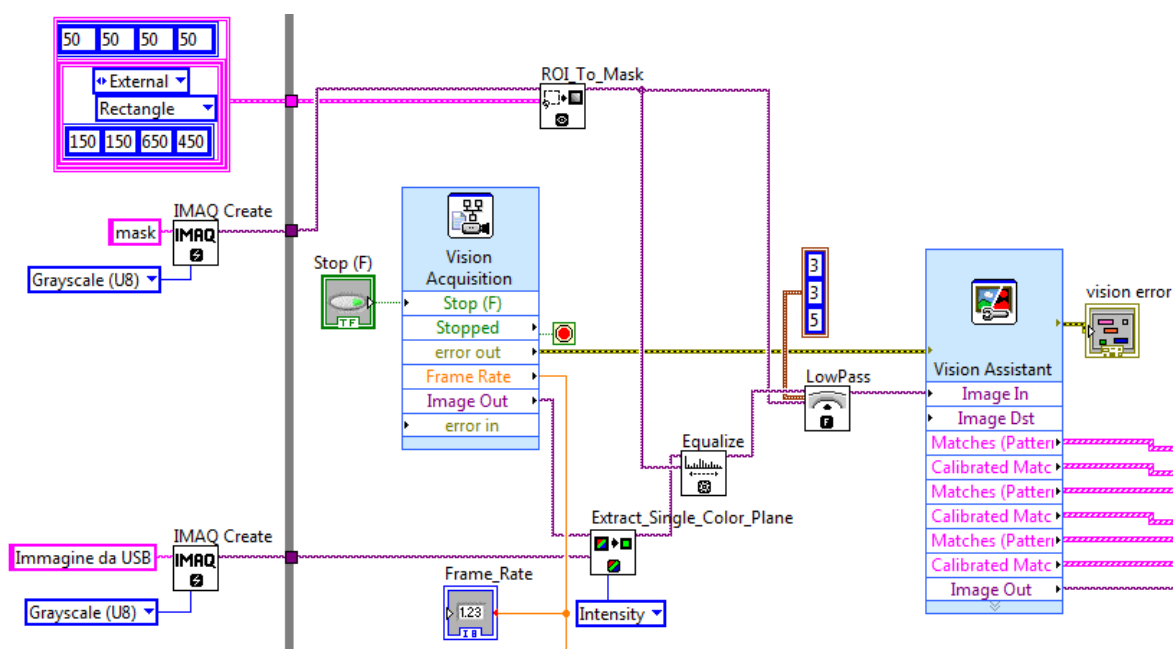


Figura 2.5. Porzione di codice in LabView che realizza il processo di acquisizione e elaborazione delle immagini. Le funzioni a destra del contorno verticale sono comprese all'interno del loop while, dunque sono eseguite per ogni immagine acquisita dal Vision Acquisition.

2.2 Pre-elaborazione

L'immagine acquisita viene elaborata per migliorarne la qualità con l'obiettivo di rendere più accurato il risultato fornito dall'algorithmo di localizzazione dei markers. In fase di elaborazione vengono eseguite tre operazioni in sequenza: applicazione di una maschera, equalizzazione, filtraggio.

- Applicazione della maschera

La porzione di immagine acquisita che viene successivamente processata è quella entro i limiti della regione di interesse (ROI), posta nella zona centrale dell'inquadratura. L'esclusione dei pixel prossimi al bordo dell'immagine permette una diminuzione della complessità computazionale degli algoritmi successivi, riducendo risorse di calcolo come tempo di esecuzione e memoria utilizzata. D'altra parte i pixel contenuti nel bordo dell'immagine sono quelli

soggetti a una maggiore distorsione ottica causata dalla non idealità dell'obiettivo della videocamera. Infatti, questo fenomeno, dovuto alla disuniformità di potenza di ingrandimento delle varie parti della lente, varia tipicamente in modo radiale rispetto l'asse ottico. I disturbi conseguenti sono la modifica delle forme degli oggetti inquadrati e il degrado della qualità del processo di associazione pixel-millimetri sul piano di calibrazione. Si preferisce escludere le parti dell'immagine affette da questo disturbo piuttosto che provare ad eliminare la distorsione con un algoritmo che aumenterebbe l'impiego di risorse di calcolo.

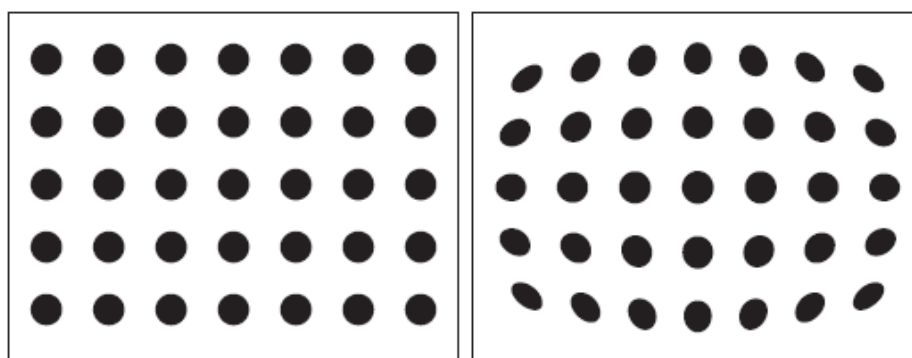


Figura 2.6. Distorsione dell'immagine causata dalla non idealità dell'obiettivo. La zona centrale è meno influenzata da questo disturbo.

La dimensione della ROI è stata scelta 500x500 pixel in modo tale da ottenere una superficie visibile sul piano osservato pari a circa 20x20 cm. La maschera ottenuta con la funzione "ROItoMASK" viene collegata all'ingresso "mask" delle funzioni di elaborazione successive affinché ignorino i pixel scartati.

- Equalizzazione

Dato che la videocamera deve operare in un ambiente in cui le condizioni di luminosità possono variare, si rendono necessarie funzioni implementate a livello hardware e software che mantengano accettabili determinate caratteristiche dell'immagine acquisita. Ottenere un'immagine con un'illuminazione adeguata, evitando quindi la saturazione, con un buon contrasto è fondamentale affinché l'algoritmo di pattern matching possa funzionare in modo corretto, localizzando i markers in qualunque condizione di luminosità ambientale. La webcam possiede la capacità di variare il tempo di esposizione del sensore CMOS in funzione della

quantità di luce presente. Tale proprietà, seppur fondamentale, può non bastare da sola a garantire il funzionamento corretto dell'algoritmo di pattern matching. La funzione di equalizzazione dell'immagine permette di mantenere alcune proprietà visive, quali la luminosità e il contrasto, indipendenti dalle condizioni di luce ambientale e a valori predeterminati. Questo avviene in modo dinamico durante il funzionamento del sistema: ogni immagine acquisita è sottoposta a questa elaborazione. Tale processo richiede una prima fase di analisi dell'immagine ottenuta dalla videocamera. La funzione "IMAQ Equalize" utilizza l'istogramma dei livelli di grigio come strumento di analisi delle caratteristiche dell'immagine. Esso consiste in una funzione che lega ciascuno dei 256 livelli di grigio al numero di pixel con la relativa intensità.

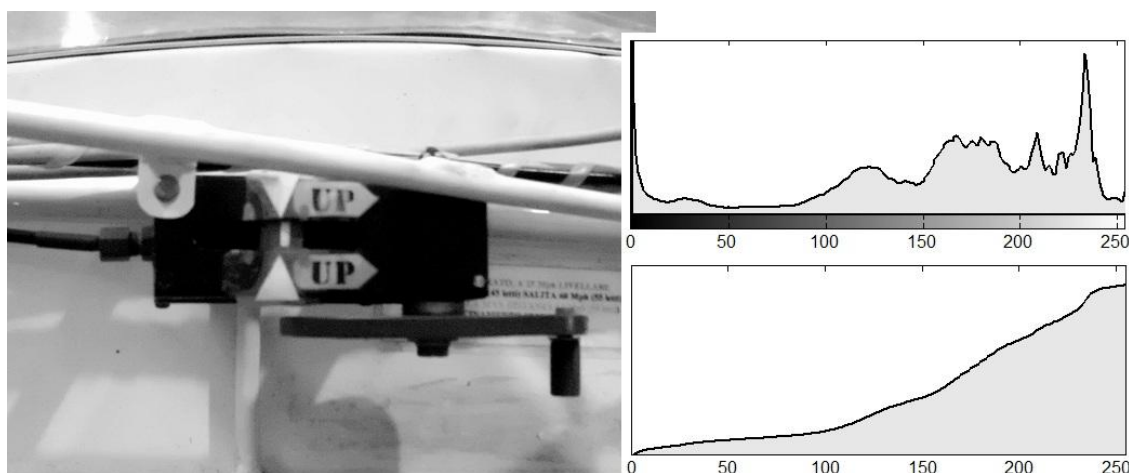


Figura 2.7. Immagine in scala di grigi con il corrispondente istogramma e istogramma cumulativo.

In questo modo, anche se non è fornita l'informazione sulla localizzazione dei valori di grigio all'interno dell'immagine, è comunque semplice individuare se l'immagine presenta saturazione, oppure se non è presente sufficiente contrasto. Infatti, un livello di illuminazione eccessivo o troppo basso, così come uno scarso contrasto generano un picco marcato nell'istogramma in corrispondenza del livello di grigio dominante. Il processo di equalizzazione si basa sulla linearizzazione dell'istogramma cumulativo. I livelli di grigio vengono uniformemente distribuiti in modo da risaltare le differenze di luminosità dei diversi dettagli riprodotti, e quindi rendere possibile una distinzione robusta dei

markers in fase di estrazione dell'informazione dall'immagine. Dopo il processo di equalizzazione, l'istogramma cumulativo dell'immagine processata possiede un profilo lineare.

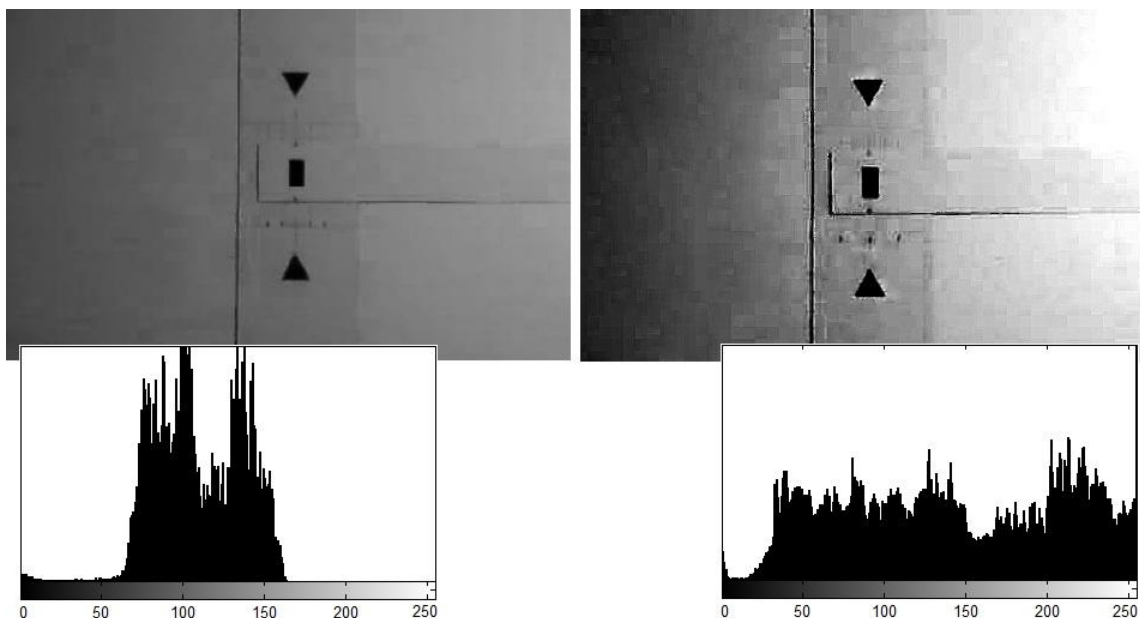


Figura 2.8. La scarsità di contrasto dell'immagine a sinistra si può notare anche dal picco nella parte centrale dell'istogramma. L'immagine equalizzata mostra invece un istogramma in cui i valori sono distribuiti in modo più uniforme (destra).

- Filtraggio dell'immagine

Una modesta qualità del sensore ottico, insieme agli effetti del processo di equalizzazione, causa la presenza di rumore, inteso come variazione aleatoria dell'intensità dei pixel nell'immagine. Questo disturbo deve essere ridotto perché contribuisce a generare un'incertezza nella determinazione della posizione del marker mobile. Per questo si introduce la funzione di filtraggio "IMAQ Low Pass" con lo scopo principale di ridurre il rumore e dunque l'errore nel processo di estrazione dall'immagine delle informazioni richieste. Un filtro di questo tipo altera il valore di ciascun pixel in modo da ridurre l'entità della variazione di intensità tra il pixel in esame e gli altri adiacenti, da qui il nome "LowPass".

L'immagine seguente illustra un esempio di filtraggio eseguito su un singolo pixel con codifica in scala di grigi ad 8bit.



Figura 2.9. Esempio di filtraggio di un'immagine a livelli di grigio. Il valore di intensità del pixel (in questo caso con valore 0) viene sostituito con la media pesata dei valori appartenenti alla matrice 3×3 in cui il pixel in esame è nella posizione centrale. La dimensione della matrice e i pesi utilizzati nel calcolo della media determinano il tipo di filtraggio eseguito. Nel caso in cui i pesi siano tutti positivi, il filtro ha un comportamento passa basso, nel senso che riduce le variazioni di intensità nell'immagine.

L'effetto di attenuazione del rumore è ovviamente accompagnato anche dalla sfocatura dei bordi dei markers, e in generale del contenuto dell'immagine. Questo però non è un problema dal momento che le immagini dei markers da localizzare vengono fornite all'algoritmo dedicato già filtrate, quindi con i bordi già sfuocati. Il vantaggio consiste nel ridurre le variazioni di intensità dei pixel dovute al rumore e quindi di rendere il risultato del processo di localizzazione dei markers più stabile, ottenendo un dato più affidabile. La routine "IMAQ Low Pass" richiede in ingresso il valore minimo di variazione percentuale tra il pixel in esame e quelli adiacenti affinché il valore del pixel in esame venga sostituito con la media delle intensità di tutti i pixel considerati. Tale valore è posto pari al 5% del valore del pixel in esame, un valore inferiore garantisce maggiore immunità al rumore ma causa un degrado notevole della qualità dell'immagine. Infatti, un filtraggio troppo intenso causerebbe una perdita di informazione nell'immagine che porterebbe a una probabile errata localizzazione dei markers.

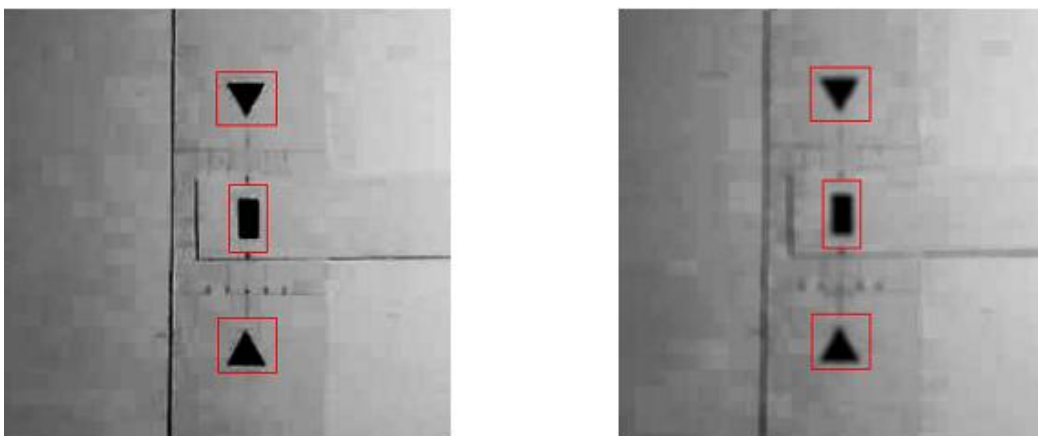


Figura 2.10. Confronto tra immagine originale (sinistra) e filtrata (destra). L'operazione di filtraggio causa una sfocatura in corrispondenza dei bordi dei markers, in cui però risulta fortemente attenuato il rumore. Ciò causa un errore minore nella determinazione della posizione dei marker.

2.3 Estrazione dell'informazione

L'informazione che si intende ricavare consiste nella posizione dei markers nell'immagine. La parte di software che implementa questa elaborazione viene eseguita utilizzando come supporto la utility Vision Assistant. Tramite questa elaborazione vengono determinate le coordinate dei markers in un sistema di riferimento in cui l'origine è posto nel vertice inferiore sinistro dell'immagine. Grazie al processo di calibrazione tali coordinate possono essere espresse in millimetri.

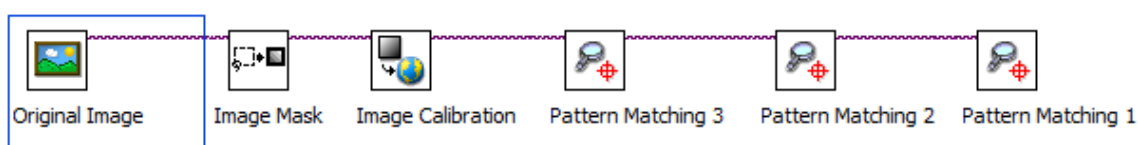


Figura 2.11. Cascata delle funzioni utilizzate per la determinazione della posizione dei markers.

Il primo passo consta nell'inserimento di una maschera che permette di considerare nel processo successivo solo la porzione dell'immagine precedentemente equalizzata e filtrata, riducendo il peso computazionale.

- Calibrazione

L'obiettivo della calibrazione del sistema in esame è garantire una corrispondenza esatta tra una distanza in pixel nell'immagine riprodotta e la relativa in millimetri nel piano in cui si trovano i markers reali anche nel caso in cui la posizione della videocamera rispetto a tale piano sia causa di un errore di prospettiva. Questo errore è dovuto alla non ortogonalità tra il piano osservato e la retta perpendicolare al sensore ottico della videocamera. Il procedimento di calibrazione è determinante per la precisione con cui l'algoritmo successivo localizza i markers.

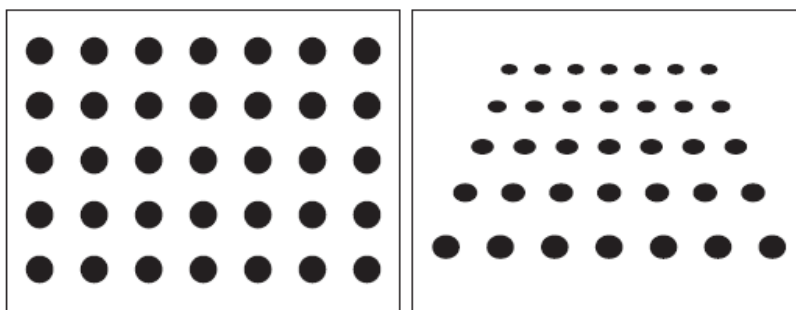


Figura 2.12. Errore di prospettiva dovuto alla posizione della videocamera rispetto all'oggetto in esame.

Dato che il funzionamento dell'intero sistema di stima della deflessione del trimtab si basa sulla determinazione di una distanza tra riferimenti nell'immagine acquisita, è di fondamentale importanza eliminare l'errore di prospettiva perché il dato ottenuto possa essere ritenuto utilizzabile. Il procedimento per ottenere la relazione tra pixel e millimetri per ogni punto dell'immagine viene eseguito preliminarmente, utilizzando come riferimento una griglia di punti con caratteristiche geometriche note. Questa griglia, denominata "calibration template", viene posta sullo stesso piano in cui si trovano i marker in modo che copra l'intera immagine ripresa dalla videocamera.

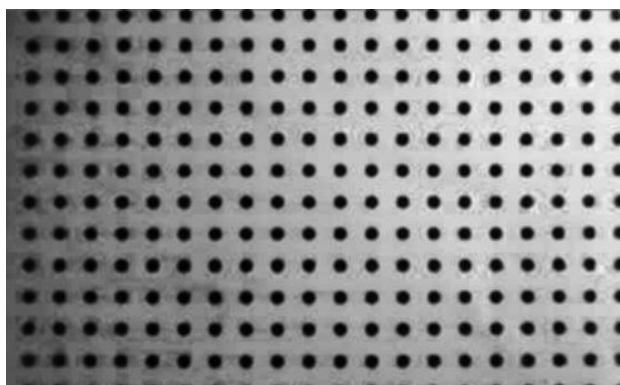


Figura 2.13. Immagine acquisita della griglia utilizzata per la calibrazione. I cerchi neri hanno diametro 0,5mm, e distanza centro-centro di 10mm sia in verticale che in orizzontale.

Date le caratteristiche geometriche note del riferimento di calibrazione, un algoritmo software, elabora l'immagine acquisita dalla webcam contenente la griglia, e fornisce un coefficiente che permette la trasformazione da pixel a millimetri. Durante l'esecuzione del programma, l'informazione di calibrazione ottenuta è applicata ad ogni nuova immagine acquisita. L'algoritmo di calibrazione viene eseguito una sola volta durante le fasi di inizializzazione del software. I dati da esso ottenuti verranno applicati come correzione di tutte le immagini acquisite. Affinché il successivo algoritmo di localizzazione possa ottenere le posizioni dei markers come coordinate espresse in millimetri è ovviamente necessario definire un sistema di riferimento. Si considera come origine il vertice inferiore sinistro dell'immagine.

- Algoritmo di Pattern Matching

L'algoritmo di pattern matching, o template matching, ha lo scopo di localizzare nell'immagine acquisita una sottoimmagine, o template, precedentemente memorizzata, fornendone le coordinate nel sistema di riferimento definito in fase di calibrazione. Da tali coordinate, espresse in millimetri, è possibile ricavare le informazioni richieste sul posizionamento relativo dei markers. L'accuratezza della stima della posizione dei riferimenti dipende da diversi fattori: risoluzione del sensore ottico, qualità dell'obiettivo, condizioni di illuminazione, vibrazioni meccaniche.

La funzione di pattern matching implementata da National Instrument è in grado di identificare il template in diverse condizioni ambientali, ad esempio tollera variazioni uniformi di illuminazione dell'immagine e piccole rotazioni del pattern da individuare. Ciò ovviamente non toglie che l'elaborazione eseguita per aumentare la qualità dell'immagine non sia in generale necessaria, ad esempio deve essere comunque sempre garantito un sufficiente livello di contrasto affinché la localizzazione avvenga con successo. L'algoritmo di pattern matching è basato sulla tecnica nota in teoria dei segnali come Cross Correlation, o Correlazione incrociata. In seguito vengono svolte alcune considerazioni sulle generalità di questo metodo di elaborazione che assume un ruolo fondamentale in questa e in molte altre applicazioni che fanno uso di algoritmi di visione artificiale.

La Cross Correlation è un'operazione effettuata sull'immagine che prevede la computazione di un gran numero di moltiplicazioni, per questo motivo richiede una grande quantità di risorse di calcolo e in particolare è la funzione che impiega il maggior tempo di esecuzione nell'intero processo di elaborazione di ogni singola immagine. Questo tempo viene diminuito riducendo l'area della parte dell'immagine da analizzare. Le caratteristiche del metodo di localizzazione adottato non permettono il riconoscimento di template ruotati di un angolo maggiore di 5-10 gradi o con dimensioni scalate rispetto la sottoimmagine originale, si parla quindi di template matching "rigido". Ciò non rappresenta un ostacolo per questa applicazione, dato che la posizione relativa della videocamera rispetto ai marker non varia. D'altronde l'aggiunta di un algoritmo di normalizzazione rispetto le rotazioni o di scaling causerebbe un'ulteriore significativo assorbimento di risorse, conseguenza inaccettabile in molte applicazioni real-time.

Nel seguito si considera $t(x, y)$ la sottoimmagine di dimensioni $K \times L$ ed $I(x, y)$ l'immagine di dimensioni $M \times N$, con $M > K, N > L$. La correlazione incrociata tra le due immagini è rappresentata dalla formula:

$$C(i, j) = \sum_{x=0}^{K-1} \sum_{y=0}^{L-1} [t(x, y) - \bar{t}] [I(x + i, y + j) - \bar{I}(i, j)] \quad (2.1)$$

Dove $i = 0, 1, \dots, M - 1$ e $j = 0, 1, \dots, N - 1$.

Si assume che l'origine di ciascuna delle due immagini t e I sia posta nel vertice superiore sinistro.

Il valore \bar{t} corrisponde al valore medio dell'intensità del template, mentre $\bar{I}(i, j)$ è il valore medio della regione dell'immagine a cui è sovrapposto t . Il processo di correlazione consiste nel traslare la sottoimmagine t , e calcolare il valore C per ciascuna posizione di t . Questo significa, per ogni coppia (i, j) , moltiplicare il valore di ciascun pixel di t sottratto del valore medio \bar{t} , per il relativo valore del pixel a cui è sovrapposto e a cui è sottratto $\bar{I}(i, j)$, ripetere questa operazione tante volte quanti sono i pixel contenuti nel template, infine sommare i prodotti ottenendo il valore di correlazione nel punto (i, j) . La coordinata in cui C è massima rappresenta la posizione in cui il template è più simile alla regione dell'immagine a cui è sovrapposto. Deve essere osservato che questo metodo non può fornire un risultato corretto nel caso in cui il template sia posizionato vicino al bordo dell'immagine e non ci sia sovrapposizione completa.

Come detto, il template t viene sovrapposto a I in tutte le possibili posizioni rispetto gli assi x e y , non vengono perciò effettuate rotazioni o cambi di scala sulla sottoimmagine. L'aggiunta di quest'ultima funzione aggiungerebbe ulteriore complessità computazionale, già resa elevata dal calcolo dell'espressione (2.1).

La figura 2.14 mostra la procedura per il calcolo del valore di correlazione tra l'immagine $I(x, y)$ e la sottoimmagine $t(x, y)$.

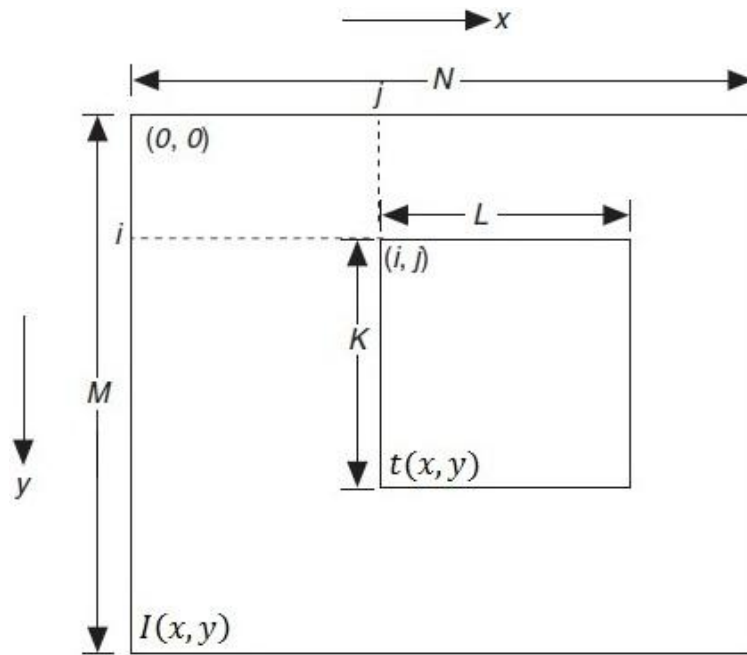


Figura 2.14. Procedura di correlazione. Per ogni posizione di t rispetto ad I viene calcolato il valore C . La coordinata per cui C è massimo è quella in cui la porzione di immagine è più somigliante al template.

Viene mostrato un esempio di questo metodo, eseguito con simulazione al calcolatore: nell'ambiente di calcolo numerico Matlab è stato realizzato il codice per la determinazione della correlazione incrociata tra le due immagini mostrate in seguito.

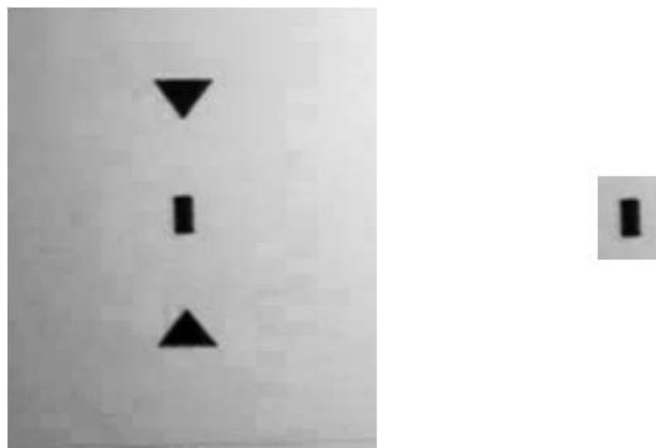


Figura 2.15. A sinistra: immagine in scala di grigi di dimensioni 239×287 pixel. Viene utilizzato come template (figura a destra) il dettaglio rettangolare al centro dell'immagine.

Ovviamente si prevede che l'algoritmo stimi una maggiore somiglianza, corrispondente a un valore massimo di $C(x,y)$, nella parte centrale dell'immagine, compresa tra i due triangoli.

Il codice, sviluppato utilizzando alcune delle funzioni di elaborazione di immagini, è riportato sotto.

```
%input dell'immagine e del template
ic=imread('immagine.jpg');
tc=imread('template.jpg');
%conversione da modello RGB a scala di grigi a 8bit
ib=rgb2gray(ic);
tb=rgb2gray(tc);
%cast dell'immagine i e del template t necessario per la
%successiva elaborazione
i = cast(ib, 'double');
t = cast(tb, 'double');
%lettura dimensioni immagini
[ih, iw] = size(i);
[th, tw] = size(t);
%dimensioni dell'immagine per cui è possibile effettuare
%l'elaborazione
M=iw-tw;
N=ih-th;
%calcolo valore medio vmi dell'intensità dell'immagine i
%considerata nella regione a cui è
%sovrapposto il template
for y=1:N
    for x=1:M
        vi(y,x)=0;
        for yt=1:th
            for xt=1:tw
                vi(y,x)=vi(y,x)+ i(y+yt,x+xt);
            end
        end
        vmi(y,x)=vi(y,x)/(th*tw);
    end
end
```

```

end
%calcolo del valore medio vmt dell'intensità del template t
vt=0;
for y=1:th
    for x=1:tw
        vt=vt+t(y,x);
    end
end
vmt=vt/(th*tw);
%calcolo della correlazione incrociata tra i e t
for y=1:N
    for x=1:M
        C(y,x)=0;
        for yt=1:th
            for xt=1:tw
                C(y,x)=C(y,x)+(t(yt,xt)-vmt)*
(i(y+yt,x+xt)-vmi(y,x));
            end
        end
    end
end
end
end
%creazione del grafico 3D
x=1:1:M;
y=1:1:N;
[X, Y] = meshgrid(x,y);
surf(X,Y,C);
shading flat;
view(30, 35);

```

La figura 2.16 mostra il risultato della simulazione eseguita. La funzione $C(x, y)$, il cui dominio è composto dai pixel dell'immagine più grande, possiede un picco centrale circa 1,6 volte superiore ai valori di C in corrispondenza dei markers triangolari. Questo risultato conferma le aspettative.

Ovviamente questo metodo funziona correttamente anche nel caso in cui i markers vengano scelti di colore chiaro e lo sfondo più scuro.

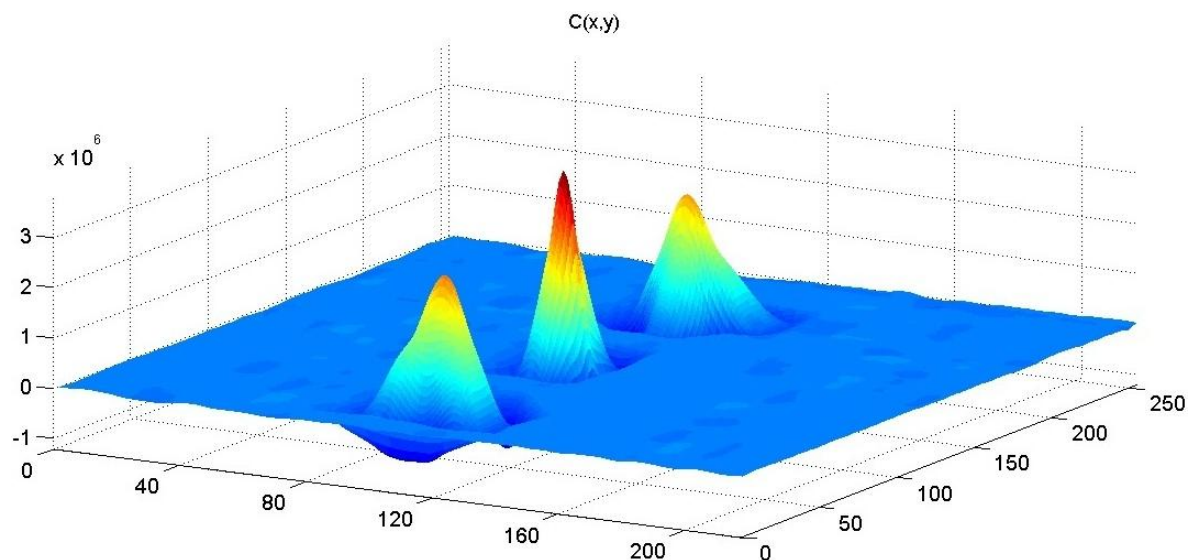


Figura 2.16. Grafico della funzione $C(x, y)$ calcolata per le due immagini di figura 2.15. Il picco centrale assume valore massimo di $3,7 \times 10^6$, mentre i due picchi laterali, a cui corrispondono i markers di forma triangolare, hanno valore $2,3 \times 10^6$.

L'esecuzione di questo algoritmo da parte del calcolatore richiede un tempo decisamente troppo elevato per poter elaborare immagini acquisite alle frequenze di funzionamento della videocamera. Per questo motivo, la funzione di ricerca del template deve essere completata con altre routine che rendano il processo di localizzazione più veloce. Un primo passo in questa direzione è la riduzione dell'area in cui calcolare il valore C espresso dalla (2.1): dall'analisi dell'immagine si possono identificare ed eliminare le zone con un contrasto minore di una determinata soglia, cioè in cui la differenza tra il valore minimo e massimo di intensità sia talmente basso da ipotizzare assenza di oggetti da identificare. Questo è solo uno dei metodi aggiunti al calcolo di C con cui viene effettivamente implementato l'algoritmo di template matching.

2.4 Calcolo del parametro di interesse

Una volta eseguito l'algoritmo di pattern matching e ottenute le coordinate in millimetri di ciascuno dei tre riferimenti nel sistema di riferimento fissato, il dato da ricavare è la distanza d tra il marker mobile (rettangolare) e la retta s definita dai due markers fissi (triangolari). La relazione che lega il valore d alla deflessione del trim tab è un'equazione lineare.

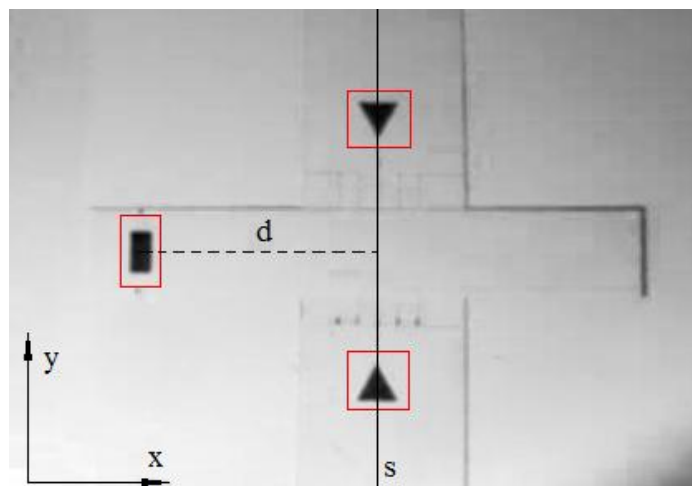


Figura 2.17. Immagine dei tre markers. Le coordinate x dei due triangoli non sono in generale identiche a causa di una rotazione relativa (e comunque di piccola entità) della videocamera rispetto i due riferimenti.

Il valore d è calcolabile sfruttando semplici considerazioni geometriche. Dalle coordinate x, y dei due triangoli sono determinati i parametri m ed n della retta s

$$s: y = mx + n$$

L'equazione che permette di ottenere la distanza di un punto da una retta è la seguente, dove x_r, y_r è la coppia di valori che definisce la coordinata del marker mobile rettangolare.

$$d = \frac{y_r - mx_r - n}{\sqrt{1 + m^2}}$$

Grazie all'operazione di calibrazione, che ha reso possibile l'espressione delle coordinate in millimetri, anche la distanza d è definita in millimetri.

Al valore d deve essere attribuito un segno, considerato positivo se il marker rettangolare risulta essere a destra di s . Per la determinazione di questa proprietà, viene calcolata la coordinata x_d del punto di intersezione tra il segmento d e la retta s . Se r è la retta perpendicolare a s , su cui giace il segmento d :

$$\begin{cases} s: y = mx + n \\ r: y = Mx + N \end{cases}$$

$$\text{In cui } M = -1/m, \quad N = y_r - Mx_r$$

$$Mx_d + N = mx_d + n$$

Da cui

$$x_d = \frac{n-N}{M-m}$$

L'informazione richiesta è dunque il segno della differenza tra x_r e x_d .

Considerando la struttura del meccanismo di controllo manuale del trim tab, è corretto ipotizzare che la relazione tra il valore d e la deflessione dell'ala sia di tipo lineare. I parametri di tale relazione sono ottenuti mediante calibrazione in laboratorio.

A questo punto è terminato il processo di acquisizione ed elaborazione software dell'immagine per la stima della deflessione del trim tab eseguita misurando la posizione relativa dei markers. Il sistema realizzato fornisce la successione x_n dei valori espressi in gradi di deflessione dell'aletta compensatrice. Ciascuno di questi elementi è ottenuto dall'elaborazione di un'immagine acquisita dalla videocamera.

Tale risultato è però affetto da una componente aleatoria dovuta al rumore inevitabilmente presente nella struttura meccanica ed elettronica del sistema. E' perciò necessaria un'ulteriore procedura che incrementi l'accuratezza del risultato ottenuto.

Nel capitolo seguente viene trattato il procedimento di filtraggio dei valori forniti dal processo di elaborazione fin qui descritto.

Capitolo 4

Filtraggio adattativo dei dati

Il risultato fornito dal sistema di acquisizione ed elaborazione di immagini analizzato finora comprende una componente di rumore generata soprattutto dalle vibrazioni meccaniche della struttura dell'aeroplano e dalla rumorosità nel sensore ottico⁷. Questo disturbo, ipotizzato additivo, si riflette in un'incertezza di misura nei valori della successione x_n .

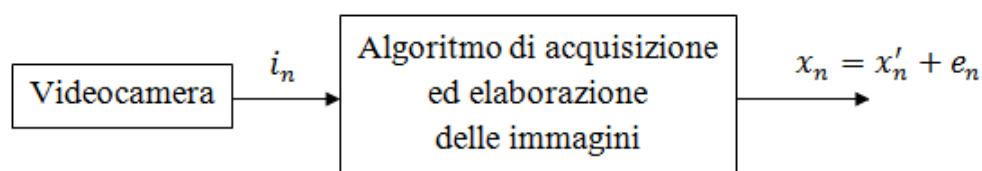


Figura 3.1. L'ingresso non manipolabile e_n rappresenta la componente di rumore nella successione x_n fornita dal sistema di determinazione della deflessione del trim tab. In questo modello x'_n rappresenta la parte di x_n depurata del rumore.

E' quindi necessario un filtraggio dei dati che aumenti il rapporto segnale-rumore di x_n in modo da poter considerare più accurata l'informazione sulla deflessione del trim tab. Il sistema realizzato per questo scopo possiede una struttura di filtraggio a media mobile, o FIR (Finite Impulse Response). La scelta di questo modello è motivata in particolare dalla possibilità di ottenere una caratteristica di fase lineare, oltre alla relativa facilità di implementazione.

Un filtro FIR è un sistema a tempo discreto caratterizzato da una risposta impulsiva h_n di durata temporale finita. Un sistema di questo tipo elabora gli ultimi N elementi del segnale tempo discreto in ingresso per realizzare una funzione di filtraggio con caratteristiche frequenziali determinate dalla funzione

⁷ Il rumore nel processo di conversione elettrica della luce si manifesta in variazioni aleatorie della luminosità dei pixel, e quindi rappresenta un disturbo per l'algoritmo di localizzazione dei markers. Il filtraggio passa basso dell'immagine contribuisce a ridurre l'entità di tale disturbo.

di trasferimento $H_S(f)$. Il legame tra risposta impulsiva e funzione di trasferimento è espresso dalla trasformata di Fourier DTFT.

$$H_S(f) = F_s[h_n]$$

con $h_n = h(nT)$, $T = \frac{1}{f_s}$ periodo di campionamento.

La relazione ingresso/uscita nel dominio del tempo è una semplice equazione alle differenze:

$$y_n = \sum_{i=0}^{N-1} b_i x_{n-i}$$

La successione dei pesi b_n coincide con la h_n .

$$b_n = h_n = F_s^{-1}[H_S(f)]$$

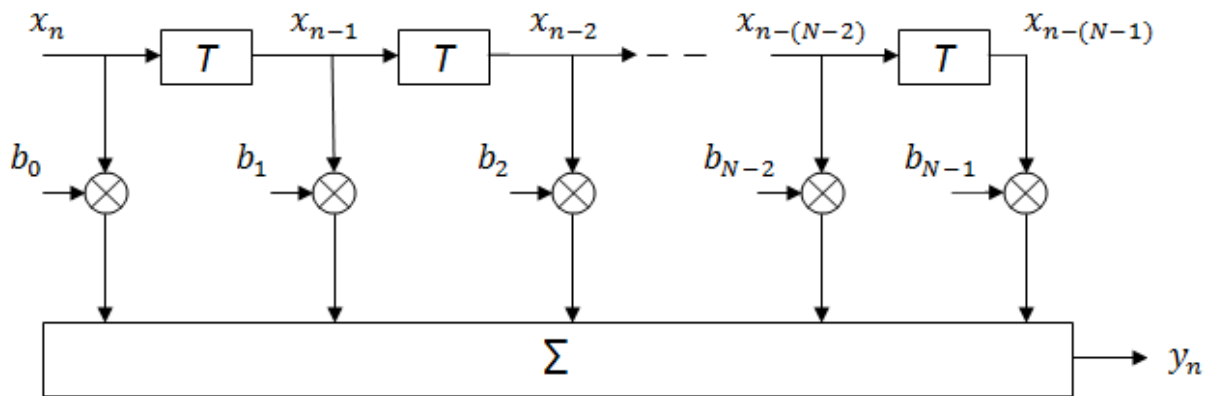


Figura 3.2. Schema a blocchi funzionali di un filtro FIR composto da N linee di ritardo, $N+1$ moltiplicatori, e un sommatore.

L'espressione della funzione di trasferimento $H_S(f)$ conferma che le caratteristiche frequenziali del filtro sono definite dalla successione b_n contenente gli N coefficienti moltiplicativi, oltre che dal valore T , inverso della frequenza di campionamento f_s .

$$H_s(f) = F_s[h_n] = \sum_{i=0}^{N-1} b_i e^{-j2\pi i f T}$$

Dato che il segnale x_n elaborato dal sistema si ipotizza in banda base, in tutte le successive considerazioni viene trattato il processo di sintesi di un filtro numerico passa basso. Per un filtro di questo tipo la banda passante B è per convenzione definita come l'intervallo di frequenze comprese tra la componente continua e la frequenza di taglio f_t per cui

$$|H_s(f_t)| = -3dB \cong 0,7$$

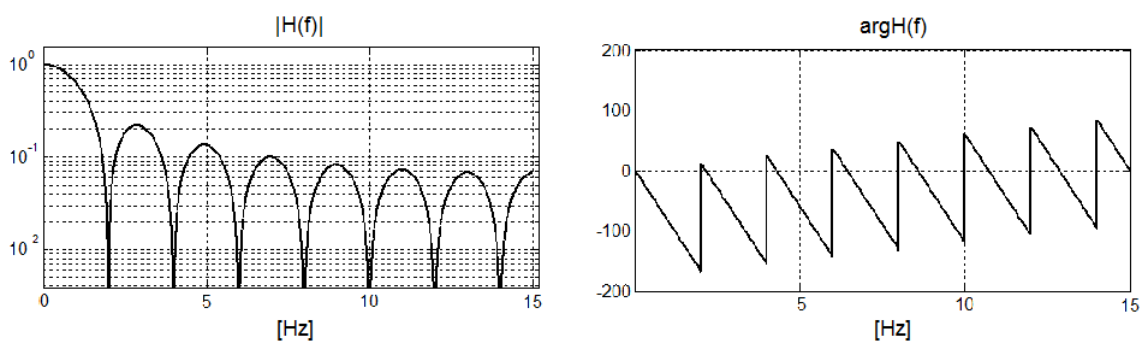


Figura 3.2. Grafico di ampiezza (in scala logaritmica) e fase della funzione di trasferimento di un sistema passa basso tempo discreto ($H_s(f)$ periodica), con frequenza di campionamento $f_s = 30Hz$. Questo sistema gode di caratteristica di fase $argH(f)$ lineare:

$$argH(f) = -j2\pi f t_0 \quad \text{con } t_0 \text{ ritardo di fase}$$

Risulta evidente che la realizzazione software di un sistema del genere richiede la memorizzazione di N valori numerici in una struttura dati di tipo First-In-First-Out. Tale struttura implementa la serie di ritardi temporali presente nel modello del filtro FIR (figura 3.2).

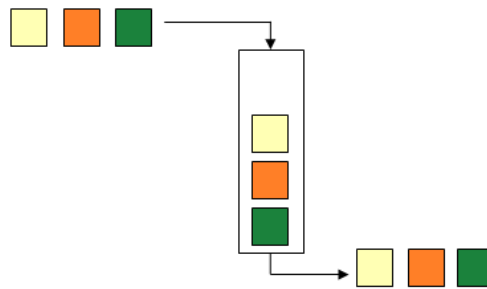


Figura 3.3. Schematizzazione di una struttura dati FIFO, denominata coda. Il comportamento richiesto è opposto a quello della struttura stack, di tipo Last In First Out.

La realizzazione nell'ambiente LabView della struttura FIFO sfrutta le funzioni per la gestione di array e la funzione di memorizzazione di un dato tra un ciclo e il successivo della struttura while. In particolare sono state utilizzate le funzioni descritte di seguito. "Array Size" per ottenere la lunghezza del vettore, fissata a 30 in fase di inizializzazione. "Array Subset" e "Replace Array Subset" sono le procedure con cui gli elementi dell'array FIFO vengono shiftati di una posizione. Con "Index Array" si accede alla posizione 0 del vettore per collocarvi l'elemento appena acquisito. "Feedback Node": con questo nome si indica la funzione che implementa un buffer con lo scopo di mantenere il valore di un dato da un ciclo al successivo all'interno del loop while. In questo caso il vettore ottenuto in un ciclo viene memorizzato per permettere la sua modifica al ciclo seguente con le funzioni sopra dette. Il comportamento di questo blocco è analogo a un ritardo temporale di un periodo T del segnale in ingresso.

In figura 3.4 viene illustrato il programma realizzato per la memorizzazione in una struttura FIFO di N elementi. Ad ogni ciclo del loop while il dato Deflessione Trim_tab, generato dal processo di elaborazione dell'immagine, viene posto nella posizione di indice 0 del vettore, dopo che tutti gli altri elementi vengono shiftati di una posizione. Array_inizializzato è un vettore di N elementi tutti nulli, necessario nel ciclo iniziale. L'icona Array_FIFO rappresenta il vettore degli ultimi N elementi acquisiti.

La dimensione N dell'array è costante e pari a 30 elementi. Il motivo della scelta di questo valore risulterà chiaro in seguito. Il numero di elementi considerati incide pesantemente sulle caratteristiche del filtraggio.

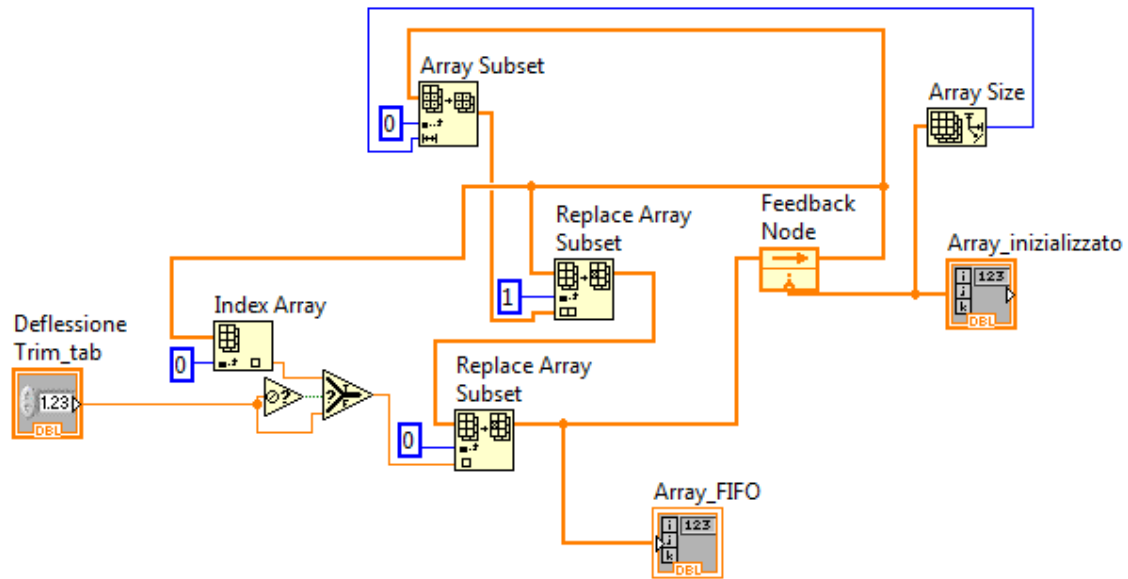


Figura 3.4. Codice G che implementa una struttura dati FIFO, denominata Array_FIFO.

Nel caso in cui, per uno o più cicli while, l'algoritmo di pattern matching non riesca ad identificare la posizione di almeno un marker, il valore Deflessione Trim_tab risulta NaN (Not a Number). E' decisamente inopportuno inserire un elemento di questo tipo nell'array FIFO perché, una volta entrato, potrà uscirne solo dopo $n = 30$ periodi di tempo $T = 1/f_s$. Dunque per il tempo nT seguente

l'algoritmo di filtraggio porterebbe in un uscita un dato senza alcun significato. Per evitare questo inconveniente viene controllata la natura del dato in ingresso: questo viene inserito nel vettore solo se è diverso da NaN.

3.1 Caratteristica adattativa del filtro

Da considerazioni legate all'utilizzo che il pilota del velivolo fa del trim tab, si può ipotizzare che il marker mobile resti nella stessa posizione per la maggior parte del tempo di volo, mentre venga spostato solo durante brevi periodi di tempo antecedenti a determinate fasi del volo, come l'atterraggio e il decollo. Per questo motivo è ragionevole concludere che il segnale x'_n (componente utile, cioè depurata del rumore, del segnale x_n) possieda una banda B di ampiezza variabile: quando il marker è fermo, l'unica componente frequenziale del segnale x'_n è la continua, mentre durante gli spostamenti del trim tab, il valore di B aumenta. In

questo senso risulta conveniente un sistema di filtraggio che adatti l'elaborazione eseguita in funzione delle caratteristiche del segnale in ingresso.

Il filtro adattativo realizzato mostra caratteristiche frequenziali variabili nel tempo e modificate in modo dinamico da un algoritmo che analizza alcune proprietà della successione x_n in ingresso. L'obiettivo di questo algoritmo è garantire il valore di ampiezza di banda passante del filtro FIR prossimo a quello della banda della componente utile del segnale x_n . In questo modo vengono ridotte le componenti di rumore e_n ad alta frequenza⁸, evitando di eliminare componenti armoniche del segnale utile, quindi introducendo un ritardo minimo in uscita quando il segnale da elaborare è soggetto a variazioni.

Il filtro implementato è un sistema tempo-variante: la funzione di trasferimento cambia dinamicamente. Le specifiche imposte dall'applicazione prevedono un filtraggio passa basso con valore di frequenza di taglio f_t variabile. Nel caso in cui il marker sia fermo le variazioni osservate del segnale sono dovute solo al rumore, dunque è necessaria una frequenza di taglio prossima a 0Hz. Mentre nel caso in cui il pilota agisca sul comando del trim tab, il valore di banda passante deve aumentare. Per il sistema implementato i valori assunti da f_t appartengono all'intervallo $[0.5, 2]Hz$.

La struttura del sistema è composta da un filtro FIR per cui i parametri della funzione di trasferimento variano in dipendenza all'elaborazione effettuata parallelamente da un algoritmo che analizza il segnale x_n nel dominio del tempo. Come detto, l'obiettivo è modulare il valore della frequenza di taglio in modo da ottenere in modo dinamico una banda passante prossima a quella del segnale utile in ingresso, garantendo così la massima esclusione di rumore ad alte frequenze e introducendo un ritardo minimo nel segnale in uscita, dato che si ipotizza che non vengano mai eliminate componenti armoniche del segnale x'_n .

Nella figura sottostante è illustrato il modello del filtro realizzato. L'uscita y_n risulta solo l'approssimazione della componente utile di x_n . Questo è dovuto al fatto che il filtro implementato, per la sua natura passa basso, non è in grado di

⁸ In questo caso “alta frequenza” indica tutte le componenti di rumore a frequenze maggiori della massima componente del segnale x'_n .

eliminare il rumore sovrapposto al segnale utile nel dominio della frequenza. In più l'intervallo di variazione di f_t è stato scelto in base a stime empiriche, quindi soggette ad errore, del segnale da filtrare.

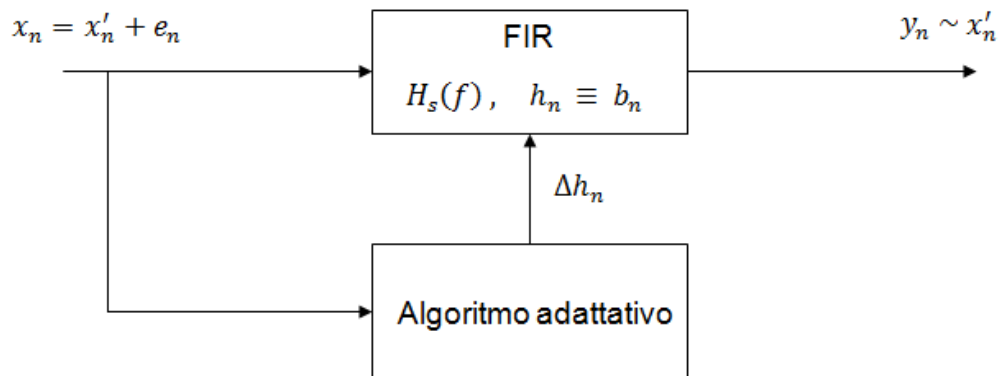


Figura 3.5. Schema a blocchi funzionali del filtro adattativo. Il segnale tempo discreto in uscita viene idealmente posto uguale alla componente utile di segnale in ingresso, cioè al segnale in ingresso depurato del rumore ad alta frequenza. L'algoritmo modifica la risposta impulsiva del filtro FIR ogni T secondi, processando il segnale x_n .

Un tipo di algoritmo adattativo è stato ideato dal teorico di matematica finanziaria Perry Kaufman, che negli anni '90 studiava metodi per l'ottimizzazione dell'analisi di mercati finanziari⁹.

La soluzione proposta [7], denominata media mobile adattativa, o AMA (Adaptive Moving Average), ha l'obiettivo di ridurre la maggior quantità di rumore presente nel segnale da trattare, introducendo il minimo ritardo nel segnale elaborato. Il processo di adattamento avviene tramite il calcolo di un coefficiente ER (Efficiency Ratio) ricalcolato ad ogni nuovo elemento di x_n ricevuto. A partire dal valore di ER, tramite una legge che nel caso in esame è stata scelta in modo empirico, si ottiene il valore del parametro dinamico contenuto nella funzione di trasferimento del filtro.

Il parametro ER_n viene calcolato in corrispondenza dell'istante di tempo nT processando gli ultimi N valori elaborati dal sistema di acquisizione descritto nel

⁹ L'algoritmo adattativo a cui si fa riferimento in questo testo è descritto nel libro "Smarter Trading. Improving performance in changing markets", pubblicato nel 1994.

capitolo 2, cioè i valori di x_n appartenenti all'intervallo $\{x_0; x_{N-1}\}$ ¹⁰. La quantità ottenuta è definita dal rapporto tra due indicatori denominati Direzionalità e Volatilità. Questi nomi appartengono al campo dell'analisi tecnica dei mercati finanziari, ma rendono chiaro il significato delle entità che esprimono.

Il parametro D_n (Direction) è definito come il modulo della differenza tra il valore x_0 , ultimo dato ricevuto, e x_{N-1} acquisito in anticipo di $N-1$ periodi di tempo.

$$D_n = |x_0 - x_{N-1}|$$

Il valore V_n (Volatility) è posto uguale alla somma dei moduli delle differenze di ciascun elemento con il precedente:

$$V_n = \sum_{i=0}^{N-2} |x_i - x_{i+1}|$$

Si ottiene

$$ER_n = \frac{D_n}{V_n}$$

È evidente che l'intervallo di variazione dei valori di questo coefficiente sia $[0,1]$, dal momento che può valere solo $D_n < V_n$. Si ha che $ER_n \approx 0$ nel caso in cui le variazioni degli ultimi N campioni del segnale d'ingresso abbiano entità inferiore rispetto a quelle causate dal solo rumore. Mentre $ER_n \approx 1$ se le variazioni del segnale acquistano un valore maggiore rispetto al rumore.

¹⁰ Questi valori vengono memorizzati nell'array FIFO descritto in precedenza. Nella trattazione seguente si osserverà che il numero di elementi considerati nel calcolo di ER_n può essere inferiore alla lunghezza del vettore, posta a 30 elementi.

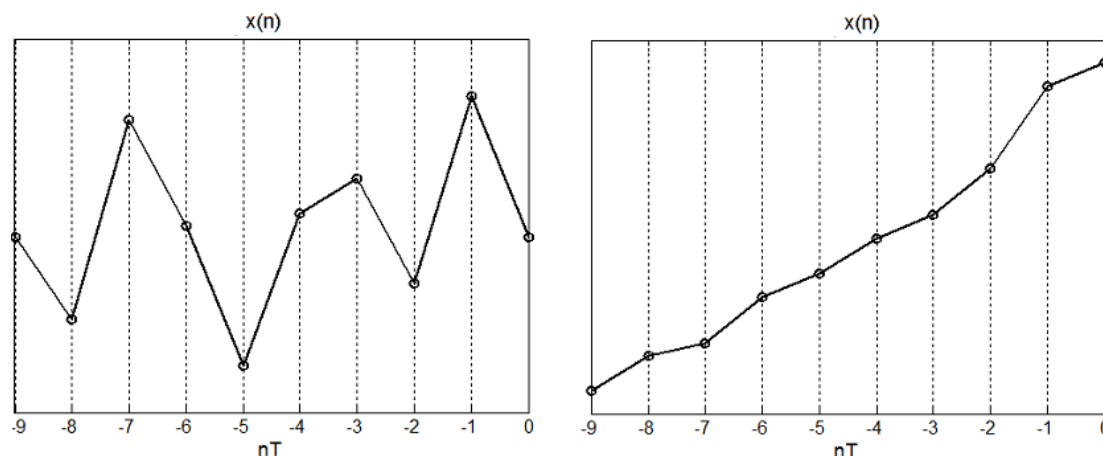


Figura 3.6. I due grafici sopra riportati mostrano due casi opposti in cui il parametro ER assume valori prossimi a 0 e 1 rispettivamente. Il numero di campioni N utilizzati è posto uguale a 10.

Di seguito è mostrata la realizzazione in ambiente LabView del software per la determinazione di ER_n ¹¹. Il valore N_{max} è il numero di elementi della successione x_n utilizzati nel calcolo. N_{max} risulta dipendere dalla frequenza di campionamento f_s ; questo aspetto viene approfondito nel paragrafo 3.3 “Considerazioni sulla frequenza di campionamento”.

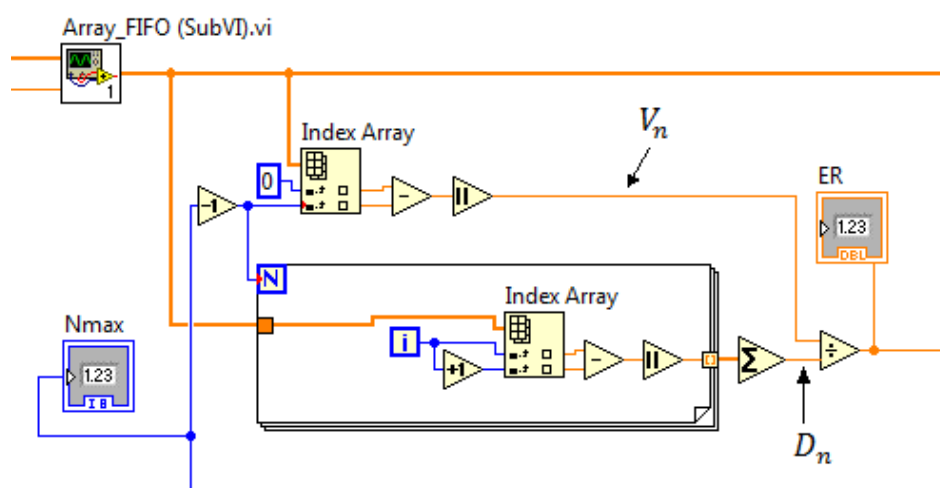


Figura 3.7. Codice G che realizza il calcolo del parametro ER secondo la definizione.

¹¹ Il pedice n indica che questo parametro viene calcolato per ciascun periodo di tempo nT , cioè ad ogni nuovo elemento acquisito. Nel seguito si evita l'utilizzo di questo pedice.

Il parametro che effettivamente modifica in modo dinamico il valore della frequenza di taglio del filtro è il numero N di campioni utilizzati per il calcolo della media mobile. In seguito verrà esposto il metodo utilizzato per implementare un filtro in cui esiste una relazione di proporzionalità inversa tra N e f_t .

La caratteristica di tempo varianza del sistema non permette di sintetizzare il filtro utilizzando come riferimento determinate caratteristiche frequenziali, cioè un'unica funzione di trasferimento $H_s(f)$ che ne descriva completamente il comportamento. E' però possibile considerare le proprietà della $H_s(f)$ nei due casi progettuali estremi di ampiezza di banda passante prossima a 0,5 Hz e 2Hz. I due modi di funzionamento del sistema corrispondono a due valori di N estremi: N_{max} ed N_{min} rispettivamente.

Si è ritenuto opportuno scegliere la relazione tra ER e N in modo tale da rendere il parametro N più stabile nei casi opposti in cui si abbia ER prossimo agli estremi 0 e 1. Viene così data più importanza ai livelli di reattività minima e massima del filtro. L'espressione di $N(ER)$ è la seguente:

$$N_n = \left[\frac{1}{2} (N_{max} - N_{min}) (\cos(\pi ER_n) + 1) \right] + N_{min}$$

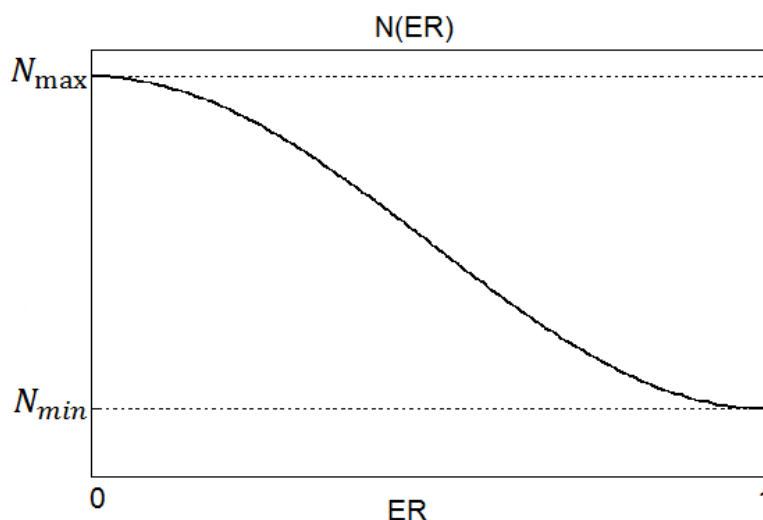


Figura 3.8. Dipendenza tra ER e N scelta di tipo cosinusoidale per garantire maggiore stabilità del parametro N nei due casi opposti illustrati in figura 3.6.

3.2 Processo di sintesi

La scelta della successione b_n dei pesi utilizzata influisce profondamente sulle caratteristiche del filtraggio. In seguito verranno condotte alcune considerazioni sulla modalità di determinazione dei pesi, basata sul metodo delle finestre [8].

L'obiettivo del processo di sintesi dei pesi è l'ottenimento di una risposta impulsiva h_n , e quindi di una funzione di trasferimento $H_s(f)$, con proprietà che soddisfino le specifiche imposte. In questo modo il comportamento del filtro può avvicinarsi a quello richiesto per l'applicazione.

Dalle specifiche progettuali di filtraggio si può formalizzare un'espressione per la risposta in frequenza, $H_{id}(f)$, che il sistema di filtraggio dovrebbe avere per garantire il funzionamento desiderato:

$$H_{id}(f) = \begin{cases} 1 e^{-j2\pi fT\alpha} & |f| \leq f_t \\ 0 & f_t \leq |f| \leq f_s/2 \end{cases}$$

dove αT rappresenta il ritardo di fase.

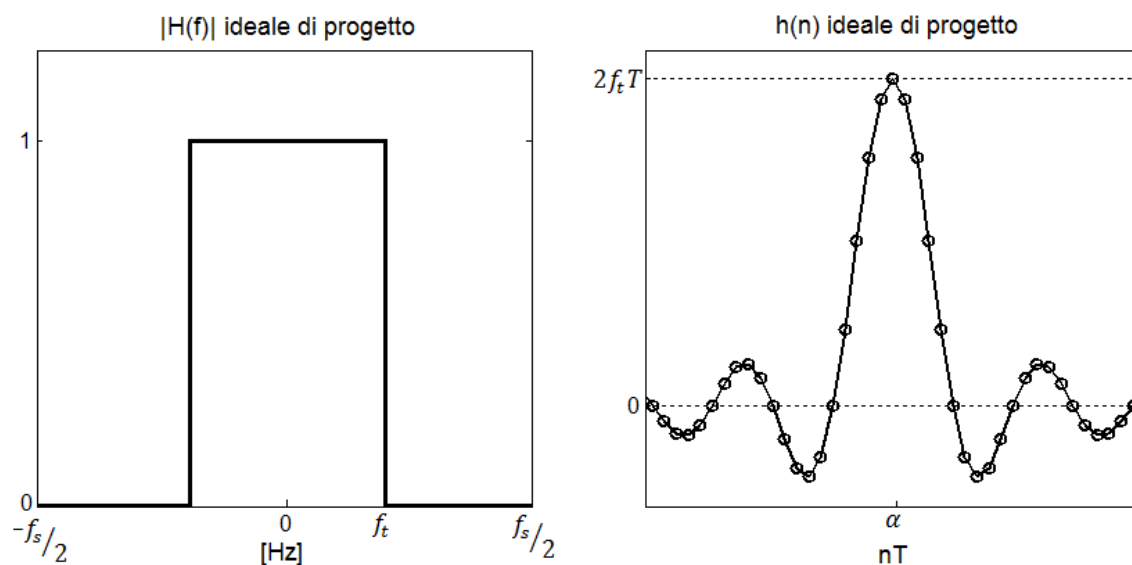


Figura 3.9. A sinistra il grafico bilatero del modulo della funzione di trasferimento $|H_{id}(f)|$, con discontinuità in corrispondenza delle transizioni tra banda passante e banda attenuata. A destra la risposta ideale all'impulso, traslata di α periodi.

La risposta all'impulso, $h_{id,n}$, ottenuta con antitrasformata di Fourier di $H_{id}(f)$ è un segnale con supporto illimitato, e ritardato αT .

$$h_{id,n} = F_s^{-1}[H_{id}(f)] = 2 f_t T \operatorname{sinc}(2f_t(n - \alpha)T).$$

Per tale funzione¹², il lobo principale ha ampiezza $nT = \frac{1}{f_t}$.

Le seguenti considerazioni fanno intuire che un filtro con pesi ottenuti da questa risposta impulsiva non sia effettivamente realizzabile perché sarebbe necessario un numero infinito di pesi, mentre per la definizione di filtro FIR, la successione b_n deve essere finita. Ai fini della causalità del sistema, i valori della risposta impulsiva, quindi i pesi, devono essere nulli per $n < 0$.

Per tali ragioni ci si deve accontentare di ottenere gli elementi b_n considerando una versione di $h_{id,n}$ traslata e troncata, come si vedrà a breve. L'approccio più immediato al progetto del filtro FIR consiste nell'ottenere una risposta impulsiva di lunghezza finita troncando la risposta infinita $h_{id,n}$ traslata di αT in modo da garantire valori simmetrici nell'intervallo considerato. Affinché questo accada deve valere:

$$\alpha = \frac{N - 1}{2}$$

La figura seguente illustra un esempio di troncamento della funzione *sinc* che rappresenta la $h_{id,n}$. In particolare viene considerato il solo lobo principale.

¹² La funzione *sinc* che compare nella $h_{id,n}$ è definita come:

$$\operatorname{sinc}(x) = \begin{cases} \frac{\operatorname{sen}(\pi x)}{\pi x} & \text{per } x \neq 0 \\ 1 & \text{per } x = 0 \end{cases}$$

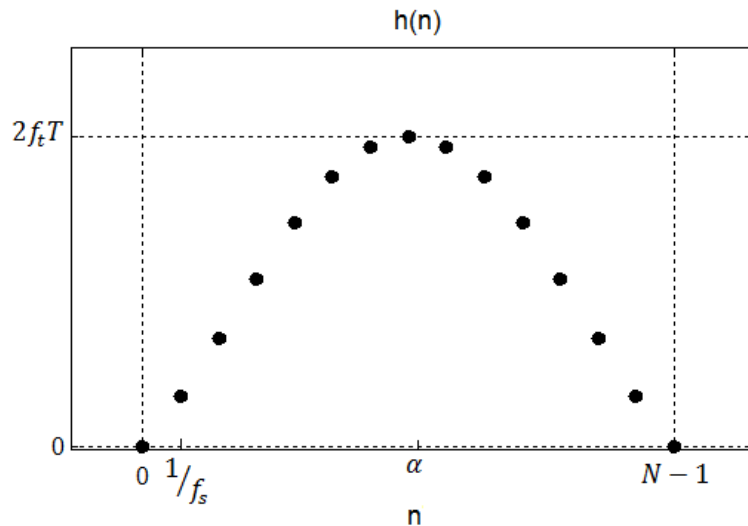


Figura 3.10. Successione h_n ottenuta dal troncamento della risposta impulsiva ideale. Il ritardo di fase scelto garantisce la simmetria dei pesi, quindi la simmetria della caratteristica di fase. La distanza temporale tra i campioni è $T = 1/f_s$.

L'espressione generale della risposta impulsiva è la seguente:

$$h_n = \begin{cases} h_{id}(nT) & 0 \leq n \leq N-1 \\ 0 & \text{altrove} \end{cases}$$

$$h_n = 2 f_t T \operatorname{sinc} \left(2 f_t \left(n - \frac{N-1}{2} \right) T \right) \quad n = \{0, 1, 2, \dots, N-1\}$$

L'operazione di troncamento, o finestatura, sopra riportata corrisponde al prodotto della successione h_{id} per una successione w_n , chiamata finestra. Nel caso di finestra rettangolare si ha:

$$h_n = h_{id} \cdot w_n$$

$$\text{dove } w_n = \begin{cases} 1 & 0 \leq n \leq N-1 \\ 0 & \text{altrove} \end{cases}$$

Nel dominio della frequenza vale la seguente generica relazione, in cui w_n può avere una forma qualunque:

$$H_s(f) = H_{id}(f) * W_s(f) \quad , \quad W_s(f) = F_s[w_n]$$

In cui il prodotto di convoluzione è definito come segue:

$$H_s(f) = \frac{1}{f_s} \int_{-f_s/2}^{f_s/2} H_{id}(\varphi) W_s(f - \varphi) d\varphi$$

La $H_s(f)$ è il prodotto di convoluzione della risposta in frequenza desiderata con la trasformata di Fourier della finestra, per questo motivo $|H_s(f)|$ è una versione “smussata” della funzione di trasferimento desiderata, in cui cioè non è presente una netta discontinuità tra banda passante e banda attenuata e il guadagno $|H_s(f)|$ non assume valori costanti.

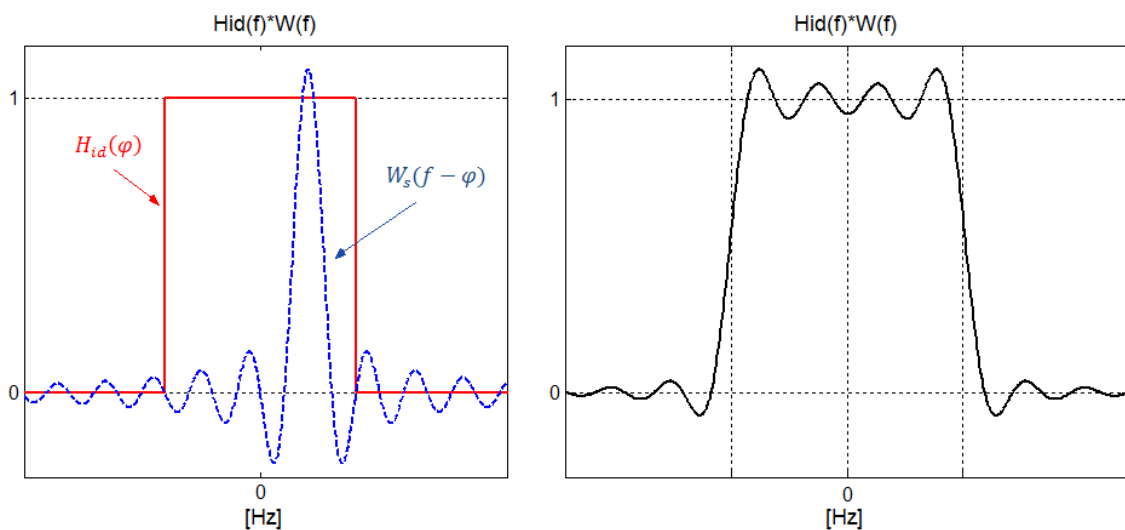


Figura 3.11. Operazione di convoluzione corrispondente all'applicazione della finestra alla risposta all'impulso desiderata.

In particolare, la scelta della funzione finestra $w_n = w(nT)$ utilizzata per il calcolo di h_n è un aspetto da cui dipendono alcune proprietà della risposta in frequenza del sistema FIR. Considerando gli effetti del prodotto di convoluzione $H_s(f) = H_{id}(f) * F_s[w_n]$, si deduce che da $F_s[w_n]$ dipendono due importanti caratteristiche di $H_s(f)$: l'ampiezza di banda di transizione, B_t , e il guadagno in banda attenuata, G_a . In seguito verranno confrontate due tipi di finestre scelte per le caratteristiche frequenziali diametralmente opposte. Le considerazioni fatte presuppongono uguali le quantità f_s, f_t, N , nei due diversi casi di analisi.

La funzione finestra rettangolare, già utilizzata in precedenza, garantisce una B_t ridotta, ma allo stesso tempo un valore massimo di G_a decisamente elevato. Viceversa la finestra di Hamming, di tipo coseno rialzato, garantisce un guadagno G_a in banda attenuata, ed una banda di transizione B_t minori rispetto al caso della finestra rettangolare.

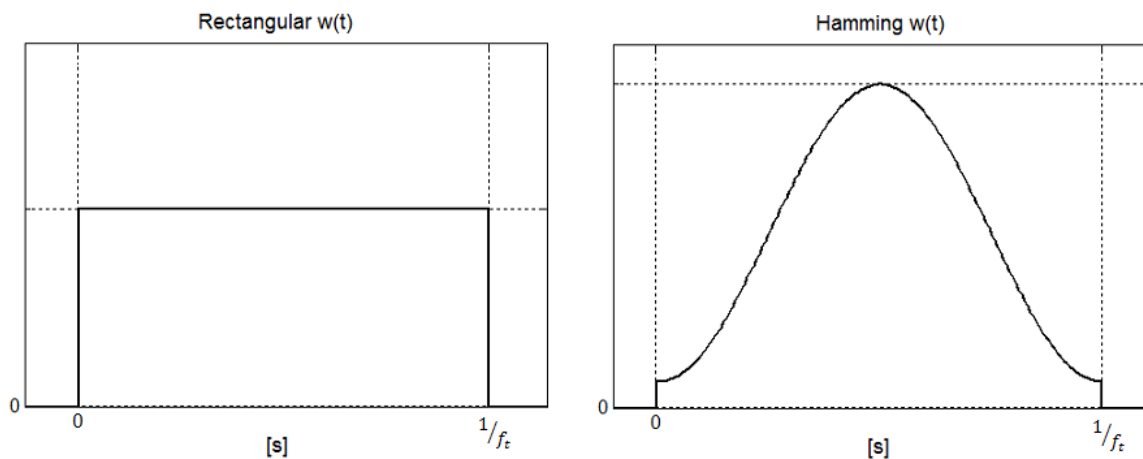


Figura 3.12. Queste finestre sono definite dalle seguenti relazioni, in cui $T = 1/f_s$:

Finestra rettangolare: $w_n = w(nT) = 1$, $0 \leq n \leq N - 1$.

Finestra di Hamming: $w_n = w(nT) = 0,54 - 0,46 \cos\left(\frac{2\pi n}{N-1}\right)$, $0 \leq n \leq N - 1$.

Nel grafico sottostante è riportato il modulo delle $F[w_n]$.

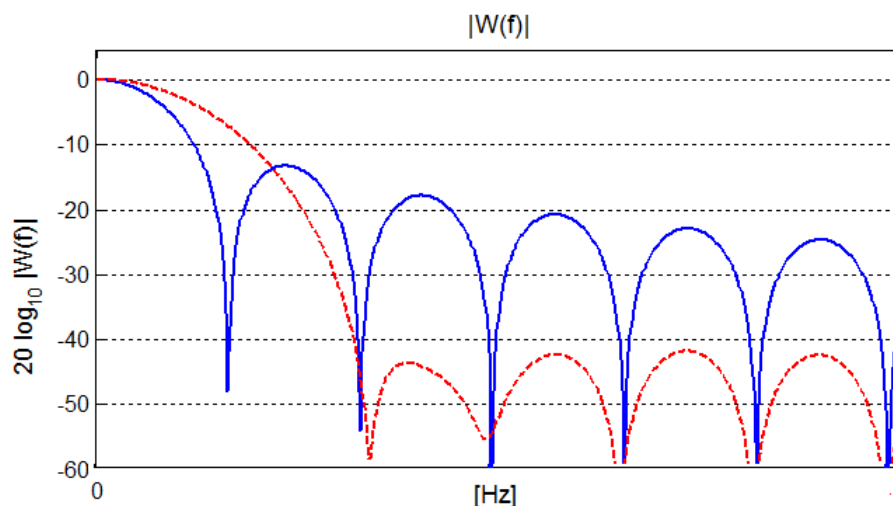


Figura 3.13. DTFT della finestra rettangolare (curva blu continua) e della finestra di Hamming (curva rossa tratteggiata).

La finestra rettangolare ha una DTFT con lobo principale visibilmente più stretto, e perciò, a parità di lunghezza N , è quella che dà luogo a transizioni più ripide nella $H_s(f)$ in corrispondenza di una discontinuità di $H_{id}(f)$. Tuttavia per tale finestra il primo lobo laterale ha un valore massimo solo di circa -13dB .

La finestra di Hamming, garantisce un guadagno dei lobi laterali inferiore a -40dB grazie al fatto che è definita in modo da raccordarsi in modo più dolce verso lo zero. Il prezzo che si paga è però una larghezza del lobo principale molto maggiore.

Nella tabella sotto riportata sono elencati i valori approssimati dei parametri di interesse nella scelta della funzione finestra.

Finestra	Altezza massima dei lobi laterali [dB]	Larghezza del lobo principale
Rettangolare	-13 dB	$4\pi / N$
Hamming	-41 dB	$8\pi / N$

Per le ipotesi fatte sul segnale in ingresso, durante la maggior parte del tempo di funzionamento del sistema l'obiettivo del filtraggio è unicamente quello di eliminare il rumore, senza restrizioni sul ritardo di fase, portando in uscita un segnale contenente componenti armoniche prossime alla continua. Proprio per questo motivo un lobo principale di ampiezza elevata risulta essere una caratteristica degradante delle qualità del filtraggio implementato per questa applicazione.

E' opportuno notare che all'aumentare del numero N di campioni la larghezza del lobo principale si riduce contribuendo a rendere il filtro meno sensibile alle frequenze più alte. Dunque per garantire un filtraggio ottimale si sarebbe portati a scegliere la finestra di Hamming aumentando N_{max} , in modo da eliminare completamente il rumore in banda attenuata e al tempo stesso ridurre B_t .

Purtroppo però, per come è stato definito ER_n , il numero N_{max} di campioni utilizzati nel calcolo di tale parametro incide sulla prontezza del filtro, e quindi sulla proprietà di adattamento alle variazioni di banda del segnale.

In particolare, un valore N_{max} troppo elevato ha l'effetto di rendere l'algoritmo adattativo meno sensibile all'evoluzione delle caratteristiche del segnale in ingresso. Infatti con l'aumentare di N_{max} il parametro ER_n tende ad assumere valori prossimi allo zero, causando una scarsa capacità di adattamento del filtro.

In definitiva, da un lato si richiede un elevato numero di campioni, N , per utilizzare la finestra di Hamming mentre, dall'altro, si richiede un numero N limitato per assicurare il giusto livello di adattamento del filtro. L'impossibilità di raggiungere un trade-off pratico rende il metodo di Hamming, per questa applicazione, meno preferibile rispetto alla finestra rettangolare.

La sintesi del filtro, per i motivi sopra esposti, impone di accettare un compromesso sul numero massimo di pesi utilizzati, N_{max} . Dalle prove sperimentali eseguite con diversi valori di questo parametro e con diverse funzioni finestra si deduce che un valore prossimo a $N_{max} = 30$ campioni ottenuti con finestra rettangolare permette, da un lato un filtraggio sufficientemente selettivo, dall'altro un buon tempo di risposta dell'algoritmo adattativo.

Nella figura seguente è mostrato il confronto tra filtraggio eseguito con pesi ottenuti dalla finestra rettangolare, e finestra di Hamming, a parità del numero di pesi.

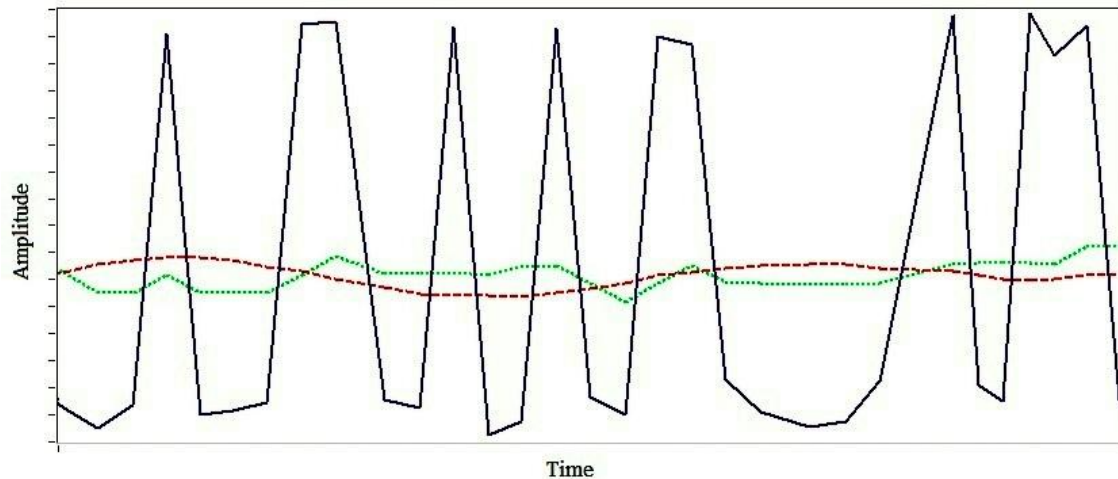


Figura 3.14. Segnali filtrati ottenuti con finestra rettangolare (linea verde a punti) e finestra di Hamming (linea rossa tratteggiata).

Dalla figura risulta evidente che, nel segnale filtrato con finestra di Hamming le componenti armoniche nella banda attenuata vengono quasi totalmente eliminate. Tuttavia il fatto che la banda di transizione abbia ampiezza maggiore rispetto alla risposta in frequenza con finestra rettangolare, fa sì che i due segnali abbiano una dinamica comparabile. Dall'analisi nel dominio del tempo in condizioni stazionarie, cioè con D prossimo a 0, quindi con variazioni del segnale dovute solo al rumore, risulta dunque che, a parità del parametro N , i due tipi di filtraggio abbiano prestazioni non molto differenti, riuscendo ad ottenere un segnale praticamente continuo con un limitato contributo di rumore superstite. Per questi motivi l'utilizzo di una finestra a coseno rialzato, quindi di un algoritmo più complesso per il calcolo dei pesi al variare di N , non è giustificato.

La successione dei pesi ottenuta con finestra rettangolare viene dunque a corrispondere ai valori della successione

$$b_n = h_n = 2f_t T \operatorname{sinc} \left[2f_t T \left(n - \frac{N-1}{2} \right) \right] \quad n \in \{0, 1, 2, \dots, N-1\} \quad (3.1)$$

Dall'ultima relazione segue che, isolando il solo lobo principale della $h_{id,n}$, come mostrato in figura 3.9, valgono le seguenti uguaglianze:

$$(N-1)T = \frac{1}{f_t}$$

da cui

$$f_t = \frac{f_s}{N - 1} \quad (3.2)$$

Dalla (3.1) risulta evidente che nel caso in cui si imponga la risposta impulsiva uguale al lobo principale della h_{id} , si crea una dipendenza di proporzionalità inversa tra il numero di elementi considerati nel filtraggio e la frequenza di taglio del filtro. Più avanti sarà motivata la scelta di una risposta impulsiva con questa forma e verranno svolte considerazioni sul parametro f_s .

E quindi:

$$b_n = \frac{2}{N - 1} \text{sinc} \left(\frac{2n}{N - 1} - 1 \right) \quad n \in \{0, 1, 2, \dots, N - 1\} \quad (3.3)$$

in cui N è il parametro dinamico da cui dipende la frequenza di taglio del filtro. Si ottiene quindi una espressione semplificata per il calcolo dei pesi in cui appare il solo parametro N .

Vengono discussi in seguito i motivi della scelta di questo procedimento di sintesi della successione b_n .

Ipotizzando di utilizzare direttamente la frequenza di taglio come parametro dinamico nella espressione:

$$b_n = h_n = 2f_t T \text{sinc} \left[2f_t T \left(n - \frac{N - 1}{2} \right) \right] \quad n \in \{0, 1, 2, \dots, N - 1\}$$

dato che N ha un limite superiore (30 in questa applicazione), imporre un valore di $f_t < \frac{f_{s,max}}{(N_{max} - 1)}$ genera una risposta all'impulso che approssima la

funzione rect al diminuire di f_t , come mostrato in figura 3.15.

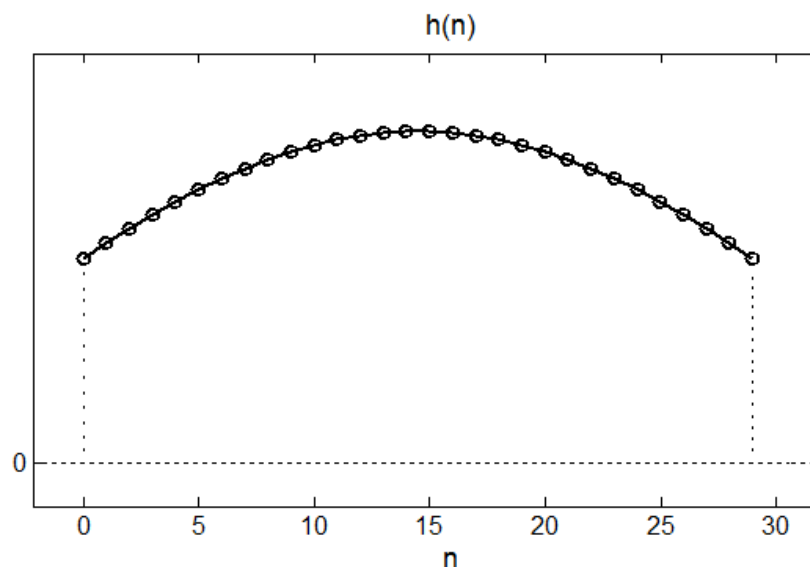


Figura 3.15. Risposta all'impulso con $f_t = 0,5\text{Hz}$; $f_s = 30\text{Hz}$; $N = N_{max} = 30$.

In questo caso la risposta in frequenza del filtro risulterebbe la convoluzione tra una funzione sinc $F_s^{-1}[w_n^{rect}]$ e una sinc approssimata $F_s^{-1}[h_n]$. Tale funzione avrebbe ancora le sembianze di una sinc presentando inadeguate prestazioni del filtro, come una scarsa attenuazione in banda attenuata. Questo difetto è riscontrabile osservando che un segnale filtrato nel modo appena esposto, presenta una quantità notevole di oscillazioni ad alta frequenza associabili a rumore debolmente attenuato.

Tale difetto potrebbe essere rimediato utilizzando la finestra di Hamming, ma rimarrebbe il problema associato alla larghezza del lobo centrale. In oltre diventerebbe necessario aggiungere al programma una funzione che renda la finestra utilizzata un ulteriore parametro aggiornato dinamicamente. L'utilizzo di una successione b_n che si raccorda in modo blando allo zero, garantisce quindi una maggiore riduzione di rumore a frequenze al di fuori della banda passante. Questo, unito a una stretta banda di transizione garantita dalla finestra rettangolare, permette di ottenere un buon compromesso sulle caratteristiche del filtro.

Utilizzando N variabile, si crea una dipendenza tra la caratteristica di fase del filtro e il parametro ER. In questo modo nel momento in cui la banda passante si allarga e il filtro deve essere più reattivo nell'inseguimento del segnale, viene

ridotto in modo dinamico anche il ritardo di fase t_0 . Dall'espressione del ritardo introdotto dal filtro si nota che questo dipende solo da N .

$$\arg H_s(f) = -j2\pi f t_0$$

$$t_0 = \alpha T = \left(\frac{N-1}{2}\right) T$$

Imporre come parametro variabile la sola f_t , mantenendo N costante, causerebbe un ritardo di fase indipendente dalle caratteristiche del filtraggio richieste. Verrebbe quindi a mancare un'importante proprietà di adattamento del sistema.

3.3 Considerazioni sulla frequenza di campionamento

Un aspetto importante nel processo di progettazione di un filtro che elabora valori acquisiti da una videocamera è la variazione della frequenza di campionamento f_s . Infatti, la webcam utilizzata possiede un frame rate di valore dipendente dal livello di luminosità presente nell'ambiente in cui opera. Il numero di frame per secondo è infatti inversamente proporzionale al tempo di esposizione del sensore ottico per ogni immagine acquisita, e appartiene all'intervallo [7,30]fps.

Nasce quindi l'esigenza di mantenere le caratteristiche frequenziali del filtro indipendenti dal valore f_s .

Nel seguito si suppone che il frame rate della videocamera corrisponda alla frequenza di campionamento f_s del segnale, ovvero alla frequenza con cui il filtro acquisisce i valori processati dell'angolo di deflessione del trim tab.

Nella formula utilizzata per il calcolo dei pesi:

$$b_n = \frac{2}{N-1} \operatorname{sinc}\left(\frac{2n}{N-1} - 1\right) \quad n \in \{0,1,2, \dots, N-1\} \quad (3.3)$$

la variabile f_s non è presa in considerazione. Il fatto che la frequenza di campionamento non sia costante nel sistema implica la necessità di un

miglioramento dell'algoritmo di filtraggio. Infatti da una prima analisi risulta evidente che un inconveniente nell'utilizzo dell'espressione (3.3) di b_n è che, essendo il range di N ammissibile fissato tra $[N_{min}, N_{max}]$, la frequenza di taglio, f_t , non può essere variata a piacimento a causa di una variazione di f_s ; ad esempio, dal legame imposto tra i parametri f_s, f_t e N , espresso dall'equazione (3.2) risulta evidente che la frequenza di taglio, nel caso in cui $f_s = 30$, $N_{min}=X$, $N_{max}=30$, abbia un limite minimo di circa $f_{t,min} = 1Hz$.

Anche se questo valore non corrisponde a quello suggerito dalle caratteristiche del segnale da trattare, per cui $f_{t,min} = 0.5Hz$, risulta evidente dalle successive considerazioni che valori di f_s che causano $f_{t,min} > 0.5Hz$ sono comunque accettabili ai fini della soppressione del rumore.

Ricordando che la banda del segnale campionato ha un valore massimo stimato $f_m = 2Hz$, il minimo valore di f_s accettabile è $f_{s,nyq} = 2f_m$, imposto dal teorema di Nyquist. Per l'applicazione in esame è ragionevole pensare che il livello di illuminazione sia tale da garantire una f_s di valore prossimo a 30 almeno per la maggior parte del tempo di funzionamento, dal momento che il sistema di acquisizione è posto nella parte superiore dell'abitacolo, separato dall'esterno da una lastra trasparente di plexiglass. E' necessario però tenere in considerazione anche le conseguenze legate a variazioni di f_s .

Nel paragrafo precedente è stata ricavata la relazione che lega i principali parametri del filtro.

$$f_t = \frac{f_s}{N - 1} \quad (3.2)$$

$$\text{con } N \in \{N_{min}, N_{max}\}$$

Per le proprietà della tecnica di sovracampionamento valori di f_s maggiori del minimo imposto dal teorema di Nyquist permettono di incrementare il rapporto segnale-rumore all'uscita del filtro.

Considerando il contributo di rumore come un processo aleatorio con densità spettrale p costante nella banda $\left[-f_s/2, f_s/2\right]$, la potenza del rumore ha espressione:

$$P_N = \int_{-f_s/2}^{f_s/2} p df$$

Tale valore, caratterizzando dal punto di vista statistico il rumore, può certamente essere ipotizzato costante.

La potenza di rumore in banda passante si riduce se, all'aumentare della frequenza di campionamento f_s , la frequenza di taglio f_t rimane invariata.

In particolare, se $f_s > f_{s,nyq}$ e $f_t = cost.$, la nuova potenza di rumore in banda passante è pari a $\frac{P_N}{f_s/f_{s,nyquist}}$.

Un'ulteriore obiettivo dell'algoritmo di adattamento è quindi rendere indipendente l'ampiezza della banda passante da variazioni del periodo di campionamento. Ciò viene realizzato facendo variare in modo dinamico e dipendente da f_s i valori N_{max} e N_{min} con cui viene calcolato N_n .

Esaminando gli estremi dell'intervallo di variazione di f_s , le frequenze di taglio del filtro che soddisfano le specifiche di progetto $f_{t,min} = 0.5Hz$ e $f_{t,max} = 2Hz$, si ottengono, utilizzato l'equazione (3.2) i valori N_{max} e N_{min} in tabella

	$f_s = 7Hz$	$f_s = 30Hz$
$f_{t,max} = 2Hz$	$N_{min} = 3$	$N_{min} = 15$
$f_{t,min} = 0.5Hz$	$N_{max} = 14$	$N_{max} = 60$

Per le conclusioni a cui si era arrivati nel paragrafo precedente, il valore $N_{max} = 60$ non è accettabile, ma deve essere dimezzato. La conseguenza di ciò è che, a frame rate sufficientemente elevata, la frequenza di taglio minima assume un

valore maggiore rispetto a quello che si voleva ottenere. Nel caso peggiore sia f_s che $f_{t,min}$ raggiungono valori massimi: $f_s = 30\text{Hz}$ e $f_{t,min} \approx 1\text{Hz}$, per cui la specifica di $f_{t,min} = 0,5\text{Hz}$ non è rispettata.

In ogni caso a un campionamento a 30Hz corrisponde una potenza di rumore in banda passante circa 4 volte minore rispetto al caso in cui $f_s = 7\text{Hz}$. Quindi, anche se alla massima frame rate, con $N_{max} = 30$, si ha una frequenza di taglio doppia rispetto quella imposta dalle specifiche, la potenza di rumore in banda passante risulta comunque dimezzata. Per questo motivo è giustificato lo sfruttamento di una frame rate variabile tramite l'aggiunta di una funzione software che renda il più possibile indipendente la banda passante del filtro dalla frequenza di campionamento.

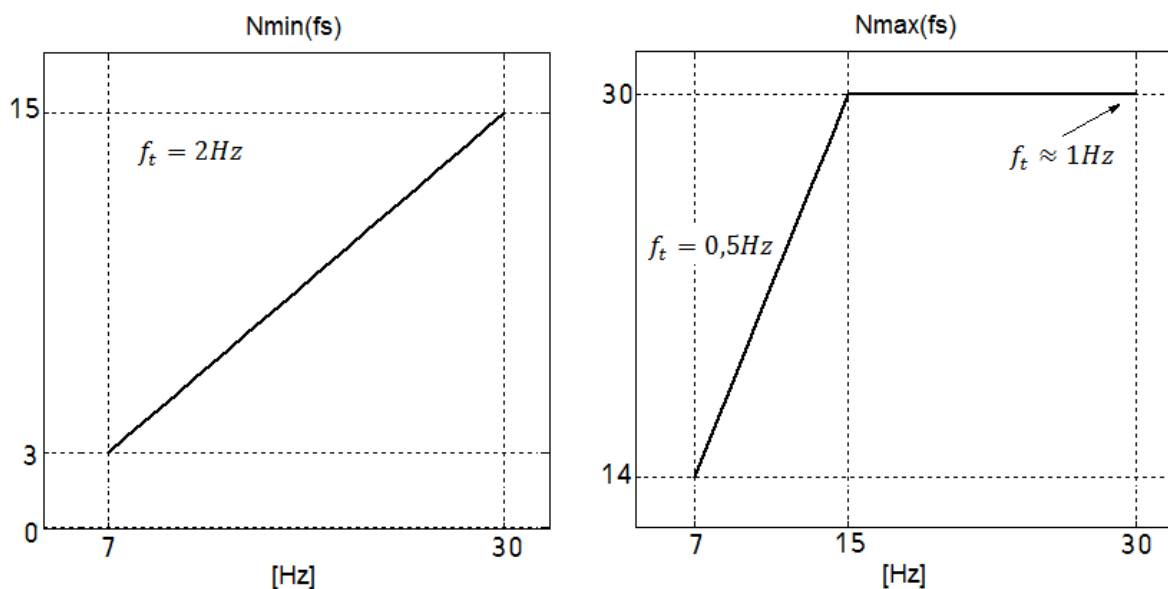


Figura 3.16. Relazione imposta ai valori N_{min} e N_{max} in relazione alla frequenza di campionamento. In questo modo la frequenza di taglio risulta essere, almeno per la maggior parte dei valori f_s , indipendente dal frame rate della videocamera. Nel grafico a destra si nota la saturazione di N_{max} a 30 e il relativo valore massimo raggiunto da f_t .

Di seguito è riportata la porzione di codice che implementa la relazione mostrata in figura 3.16 tra il valore f_s e la coppia N_{min}, N_{max} .

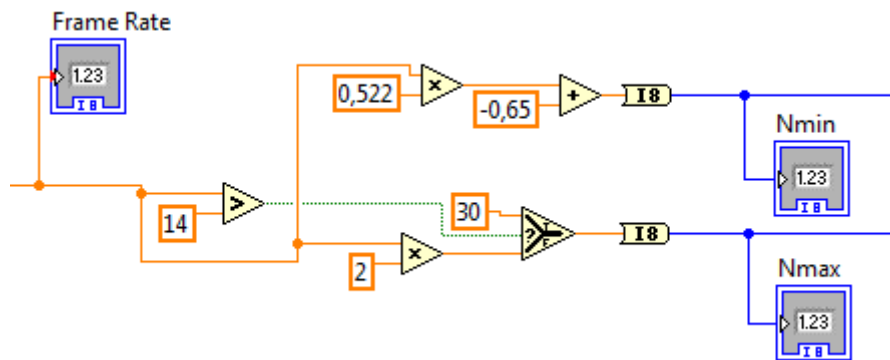


Figura 3.17. I valori N_{min} , N_{max} vengono calcolati in modo da mantenere le caratteristiche del filtraggio indipendenti dalla frequenza di campionamento nel caso in cui $N \leq 30$.

Capitolo 4

Risultati ottenuti

Vengono commentati i risultati ottenuti mediante analisi del sistema in due differenti condizioni di funzionamento: nel primo test di laboratorio vengono studiate le caratteristiche statiche del filtro imponendo il marker fermo in una posizione (test statico). Nel secondo caso invece si studia la risposta del filtro durante le fasi di moto del marker (test dinamico).

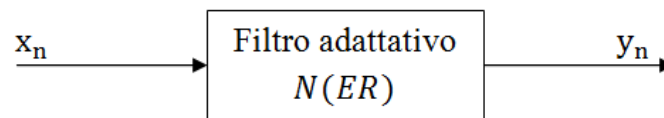


Figura 4.1. Il parametro dinamico, ricalcolato ad ogni valore x_i acquisito, è costituito dal numero N di elementi della successione x_n utilizzati nel filtraggio.

Nel primo caso il filtro presenta un valore del parametro ER prossimo a 0 , nel secondo, le caratteristiche del segnale d'ingresso impongono variazioni del valore assunto da ER nell'intervallo $[0,1]$. La relazione tra ER ed N , mostrata in figura 3.8, è stata imposta di tipo cosinusoidale

$$N = \left[\frac{1}{2} (N_{max} - N_{min}) (\cos(\pi ER) + 1) \right] + N_{min}$$

Il legame tra il parametro N e la frequenza di taglio del filtro è invece:

$$f_t = \frac{f_s}{N - 1}$$

In cui il valore di f_s è stato posto costante e pari a $30Hz$ durante i test eseguiti.

4.1 Analisi delle caratteristiche statiche

In assenza di spostamenti imposti dal pilota al marker mobile, le variazioni osservate di x_n sono unicamente dovute al rumore. Per questo motivo il filtro

adattativo modifica la funzione di trasferimento aumentando N in modo da ridurre la frequenza di taglio al valore minimo. L'incremento di N provoca l'incremento del ritardo t_0 introdotto dal filtro, questo non rappresenta un problema dato che in questo caso la componente utile del segnale filtrato è solo quella a frequenza nulla.

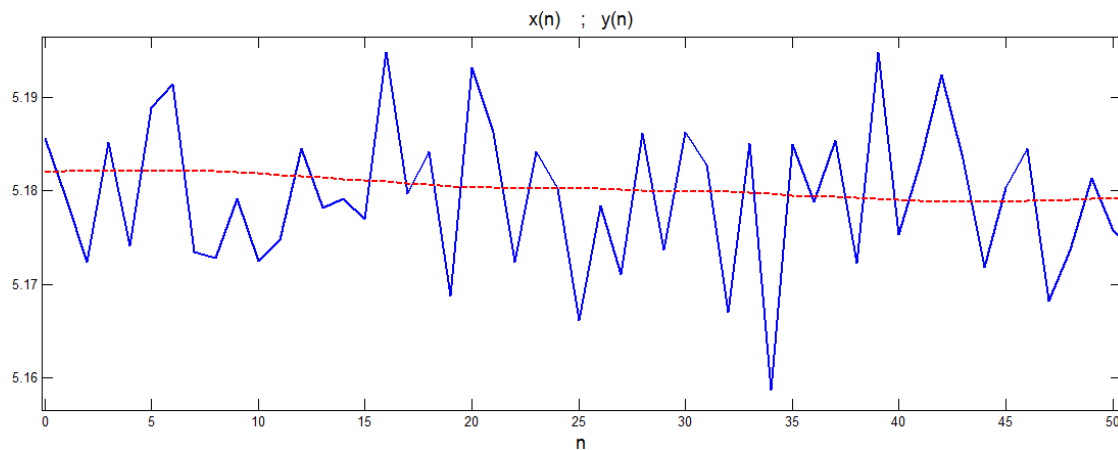


Figura 4.2. Dal segnale in ingresso (linea continua blu) viene eliminato il rumore ad alta frequenza. Il segnale filtrato (linea tratteggiata rossa) presenta solo lente variazioni la cui ampiezza è ridotta rispetto a quella del segnale non filtrato. Nel grafico sono mostrati 50 campioni di ciascuna delle due successioni.

La qualità delle caratteristiche statiche del filtraggio può essere quantificata considerando la differenza tra la deviazione standard¹³ σ_y della successione y_n rispetto la σ_x della successione non filtrata. Per il calcolo delle due quantità sono stati presi in considerazione 1500 campioni (tempo di acquisizione di circa 1 minuto) di cui i primi 50 sono mostrati in figura 4.2.

$$\bar{x}_n = 5,179 \quad \sigma_x = 0,223$$

$$\bar{y}_n = 5,179 \quad \sigma_y = 0,0447$$

¹³ La deviazione standard σ indica la variabilità di una popolazione di dati. Per una successione di N elementi con valore medio \bar{x}_n è pari a

$$\sigma = \sqrt{\frac{1}{N} \sum_{i=0}^{N-1} (x_i - \bar{x}_n)^2}$$

Per la natura passa basso del filtro, il valor medio del segnale in ingresso si mantiene in uscita. La deviazione standard del segnale elaborato risulta circa un quinto di quella della successione x_n , questo è intuibile osservando la drastica riduzione della dinamica del segnale generato dal sistema.

4.2 Analisi dinamica

Le caratteristiche dinamiche del filtro sono state analizzate studiando alcuni parametri del segnale tempo discreto $y(nT)$ considerando una variazione dell'ingresso a "gradino". Questo metodo risulta essere utile per confrontare il comportamento dinamico di sistemi. In seguito verrà confrontato il filtro implementato con uno "standard" non adattativo.

L'andamento della risposta di un sistema a un ingresso a gradino è descritta da parametri standard [9]: la figura sottostante, in cui si suppone che la variazione dell'ingresso avvenga a $t = 0$, illustra i principali.

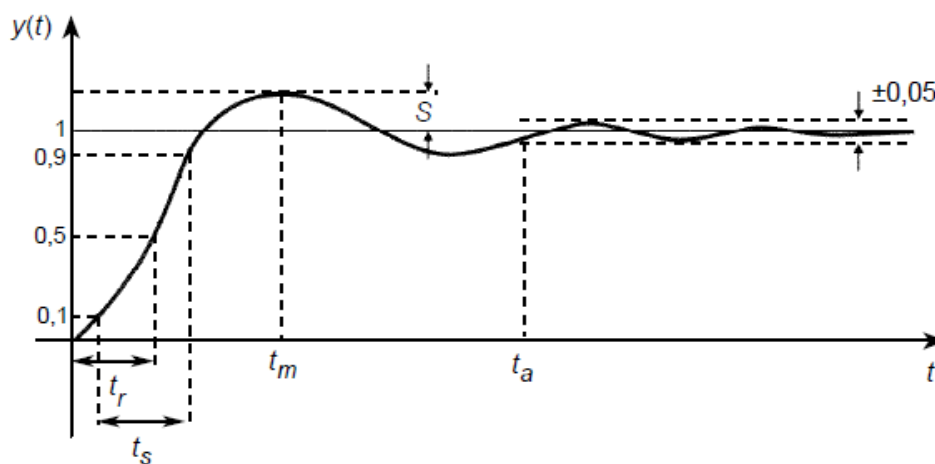


Figura 4.3. Principali caratteristiche della risposta al gradino.

- Tempo di ritardo t_r : intervallo di tempo in cui l'uscita raggiunge il 50% dell'escursione.
- Tempo di salita t_s : periodo di tempo in cui il segnale d'uscita passa dal 10% al 90% dell'escursione totale.
- Tempo di assestamento t_a : tempo in cui l'uscita raggiunge il 95% dell'escursione totale.

- Ampiezza della massima sovraelongazione S : differenza tra il valore massimo e il valore finale raggiunto dall'uscita.
- Istante di massima sovraelongazione t_m : istante in cui l'uscita assume valore massimo.

Il test è stato effettuato a frequenza di campionamento $f_s = f_{s,max} = 30\text{Hz}$. La peculiarità del filtro in esame è quella di adattarsi alle caratteristiche del segnale da elaborare. Rispetto al caso precedente, in cui la banda passante era resa minima con $N = N_{max}$, ora il sistema modifica automaticamente la frequenza di taglio, portandola a $f_{t,min}$, riducendo N fino al valore N_{min} , pari a 15 nel caso in esame, in cui $f_s = f_{s,max}$.

In seguito si confronta l'uscita del sistema implementato, in cui $ER \in [0,1]$, con la risposta di un filtro i cui parametri sono costanti e uguali a quelli assunti dal filtro in condizioni statiche, per cui $ER \approx 0$ e quindi $N = 30$.

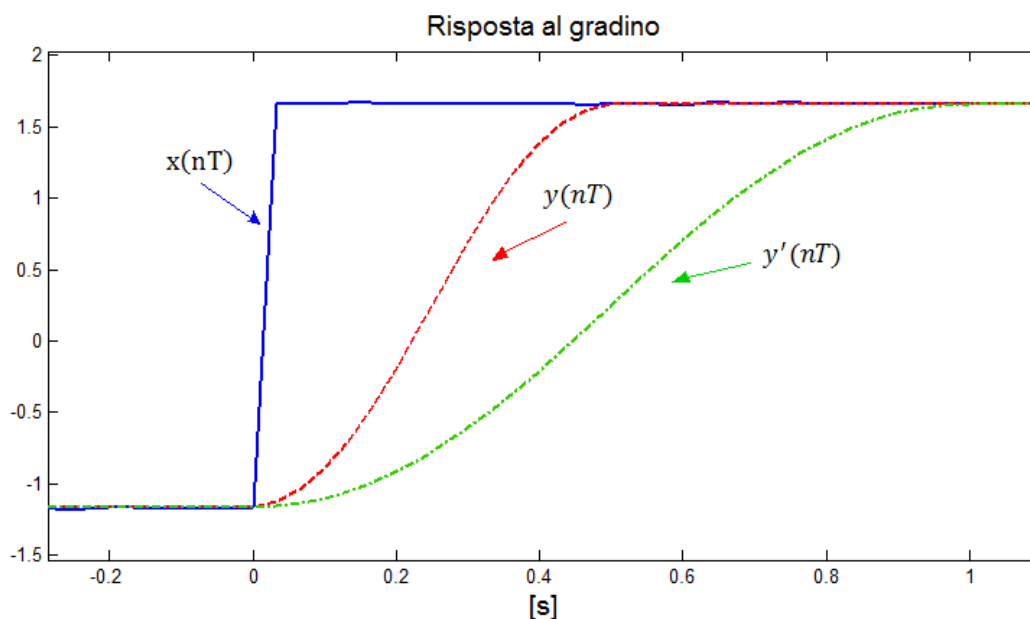


Figura 4.4 Risposta al segnale x_n “gradino”. La linea rossa tratteggiata y_n rappresenta l'uscita del filtro adattativo. y'_n è la risposta allo stesso ingresso, di un filtro numerico non adattativo, con frequenza di taglio costante pari a 1Hz, ed $N = 30$ pesi.

Risulta evidente la capacità di adattamento del filtro. L'algoritmo adattativo permette di variare dinamicamente secondo l'esigenza il numero dei pesi utilizzati nel filtraggio, dunque la frequenza di taglio. Oltre all'aumento della f_t ,

la riduzione dinamica del numero dei pesi permette di diminuire la pendenza della caratteristica di fase del filtro (imposta lineare con la scelta di pesi simmetrici), in cui l'argomento è proporzionale al ritardo t_0 introdotto dal sistema. Risulta infatti

$$\arg H_s(f) = -j2\pi f t_0$$

In cui

$$t_0 = \left(\frac{N-1}{2}\right)T \quad \text{dove } T = 1/f_s$$

Nella tabella sottostante sono elencati i risultati ottenuti nel test dinamico. I simboli con l'apice sono riferiti alla risposta y'_n del filtro non adattativo.

Tempo di salita	Tempo di ritardo	Tempo di assestamento
$t_s = 0,299 \text{ s}$	$t_r = 0,260 \text{ s}$	$t_a = 0,466 \text{ s}$
$t'_s = 0,599 \text{ s}$	$t'_r = 0,532 \text{ s}$	$t'_a = 0,899 \text{ s}$

Confrontando ciascun parametro, si osserva che il tempo di risposta del filtro adattativo risulta essere circa la metà di quello del filtro statico. Questo tempo è strettamente legato al numero di pesi utilizzati, infatti durante il tempo di risposta, nel primo caso N si riduce fino a $N_{min} = 15$, nel secondo rimane fisso a 30. In nessuno dei due casi è presente sovraelongazione dal momento che i filtri implementati sono sistemi di ordine 1.

Capitolo 5

Conclusioni

In questa tesi è stato illustrato il processo di realizzazione di un sistema di visione artificiale per la stima della deflessione di una superficie mobile. Tale sistema, alternativo ai sensori di posizione tradizionali, garantisce il massimo livello di sicurezza grazie all'assenza di contatti meccanici con parti strutturali del velivolo.

Il sistema di misura si compone di tre parti. Il sottosistema di acquisizione delle immagini costituisce la parte fisica del progetto, mentre l'elaborazione delle immagini per l'estrazione dell'informazione e il filtraggio dei dati ottenuti sono processi implementati via software.

E' stata utilizzata una videocamera webcam USB collegata con la piattaforma di elaborazione presente nell'aeroplano.

Dato che le condizioni di illuminazione variano durante il volo è necessario processare le immagini acquisite con algoritmi di elaborazione che mantengano costanti le proprietà visive (luminosità, contrasto) necessarie per l'individuazione dei markers.

L'obiettivo principale del software è la localizzazione di tre markers posti sul meccanismo di controllo manuale del trim tab. Uno di questi marker è solidale con il cavo di comando dell'aletta, gli altri due fungono da riferimenti fissi.

Determinata la distanza del marker mobile rispetto i riferimenti, il valore di questa distanza viene trasformato in gradi di deflessione del trim tab utilizzando una relazione lineare determinata in modo empirico.

La successione dei dati ottenuta è affetta da rumore. Quest'ultimo aspetto è stato trattato considerando che la deflessione del trim tab subisce variazioni imposte dal pilota solo durante brevi fasi del volo. Da questo punto di vista si può considerare che il segnale che rappresenta tale deflessione abbia un'ampiezza di

banda variabile nel tempo. E' corretto dunque utilizzare un filtraggio di tipo adattativo che modifichi il valore della frequenza di taglio in modo dinamico e dipendente dal segnale in ingresso.

La struttura del sistema di filtraggio realizzato è composta da due sottosistemi: un filtro numerico di tipo FIR e un algoritmo che, analizzando il segnale in ingresso, modifica i parametri della funzione di trasferimento del FIR, in modo da variarne la banda passante.

Per le caratteristiche di funzionamento della videocamera, la frequenza di acquisizione delle immagini risulta proporzionale al livello di illuminazione ambientale. Si è rivelato vantaggioso introdurre un'ulteriore algoritmo che mantenesse le proprietà frequenziali del filtro indipendenti dal frame rate della videocamera. In questo modo è stato possibile sfruttare le proprietà del sovracampionamento e si è evitato l'utilizzo di un sistema di illuminazione dei markers da localizzare che per mantenere il frame rate costante.

Si è riusciti ad ottenere un sistema che attenua la componente di rumore garantendo comunque la minima reattività. L'algoritmo adattativo, utilizzato soprattutto in matematica finanziaria, analizza le caratteristiche del segnale in ingresso nel dominio del tempo e permette al sistema di stimare se le variazioni sono dovute al rumore o dal controllo del pilota.

- Sviluppi futuri

Come è stato detto, l'utilizzo di un sistema di sensing che utilizza una videocamera è motivato soprattutto da ragioni di sicurezza. Ci sono però alcune applicazioni a bordo dell'aereo per cui l'utilizzo di una videocamera diventa una soluzione necessaria.

Una di queste è la determinazione della deflessione delle ali dovuta al carico aerodinamico. In questo caso la videocamera, posta sul lato del velivolo, acquisisce le immagini dell'ala e ne stima la deformazione.



Figura 5.1. Le frecce arancioni indicano il movimento di flessione delle ali sottoposte al carico aerodinamico durante il volo.

In questo caso diventa necessario realizzare un sistema di acquisizione di immagini che possieda alcune caratteristiche particolari: ad esempio è richiesto un obiettivo con profondità di campo abbastanza elevata da poter mantenere a fuoco almeno la maggior parte dell'ala. Per quanto riguarda il sensore ottico, si richiede un valore di frame rate tale da garantire una frequenza di campionamento sufficiente per poter osservare le più rapide variazioni del segnale rappresentato, ad esempio, dalla posizione di marker posti sulla superficie dell'ala.

Indice

0. Introduzione.....	pag. 3
1. Layout di sistema.....	pag. 7
1.1. Webcam USB.....	
1.2. Setup del sistema di acquisizione delle immagini.....	
1.3. Supporto hardware/software.....	
2. Sviluppo del codice.....	pag. 19
2.1. Acquisizione dell'immagine.....	
2.2. Pre-elaborazione.....	
2.3. Estrazione dell'informazione.....	
2.4. Calcolo del parametro di interesse.....	
3. Filtraggio adattativo dei dati.....	pag. 38
3.1. Caratteristica adattativa.....	
3.2. Processo di sintesi.....	
3.3. Considerazioni sulla frequenza di campionamento.....	
4. Risultati ottenuti.....	pag. 63
4.1. Analisi delle caratteristiche statiche.....	
4.2. Analisi dinamica.....	
5. Conclusioni.....	pag. 68

Riferimenti

- [1] Nicola Mimmo, LA.U.R.A. Pilot Operative Handbook, P/N 001, I-PTFB, 2012.
- [2] R.Pallas-Areny, J.Webster, “Sensors and Signal Conditioning”, Wiley, 2001.
- [3] National Instruments “IMAQ Vision Concepts Manual”.
- [4] National Instruments “NI-IMAQ User Manual”.
- [5] National Instruments “NI-IMAQdx User Manual, Image acquisition software”.
- [6] National Instruments “NI Vision for LabView User Manual”.
- [7] P.J.Kaufman, “Smarter trading. Improving performance in changing markets”, Mc-Graw Hill, 1994.
- [8] A.V.Oppenheim, R.W.Schafer, “Elaborazione numerica dei segnali”, Franco Angeli, 2010.
- [9] G.Marro, “Controlli automatici”, Zanichelli, 2004.