

POLITECNICO DI TORINO

**Corso di Laurea Magistrale
in Ingegneria Meccanica**

Tesi di Laurea Magistrale

**Rilievo del movimento umano tramite sensore Kinect
per applicazioni di robotica collaborativa**



Relatori

prof. Stefano Mauro

prof. Stefano Paolo Pastorelli

Candidato

Pierpaolo Palmieri

Dicembre 2018

Sintesi

Il crescente interesse verso la robotica collaborativa porta a ricercare soluzioni che permettano ai robot di avere indicazioni su ciò che accade intorno a loro. È interessante sviluppare metodi che permettano ai robot di riconoscere e rilevare il movimento umano, sia per questioni di sicurezza che di interazione.

I robot devono, dunque, essere equipaggiati con sensori che forniscano tutte le informazioni necessarie. Kinect è un sensore di profondità in grado di riconoscere la presenza di un umano e di fornire indicazioni sulla sua posizione mediante estrapolazione di uno scheletro. L'interesse verso il sensore Kinect in un ambiente di collaborazione fra uomo e robot è dovuto al basso costo e alla possibilità di ottenere direttamente informazioni sul movimento dell'uomo.

Il movimento fornito da Kinect è però poco accurato e nervoso, compromettendone l'effettivo utilizzo nelle applicazioni di interesse.

Nel presente lavoro si propone un modello che permetta di ottenere un movimento del braccio umano più accurato e fluido, a partire dai dati forniti dal sensore Kinect. La miglior stima del movimento del braccio viene ottenuta ricorrendo a un filtro di Kalman, mediante un modello cinematico, e attraverso l'imposizione di lunghezze fisse di braccio e avambraccio. Lo sviluppo del modello è sostenuto da prove sperimentali effettuate nei laboratori del Politecnico di Torino. I risultati ottenuti suggeriscono un'effettiva applicabilità degli algoritmi sviluppati.

Indice

Introduzione.....	1
Capitolo 1 Rilevamento ed elaborazione del movimento dell'uomo.....	5
1.1 Sensori.....	6
1.1.1 Kinect: hardware.....	8
1.2 Gesture Acquisition	10
1.2.1 Kinect: estrapolazione dello scheletro	11
1.3 Gesture Tracking	12
1.4 Gesture Classification.....	14
1.5 Riepilogo e Considerazioni.....	14
Capitolo 2 Teoria degli osservatori per un sistema stocastico.....	17
2.1 Filtri Bayesiani	18
2.2 Filtro di Kalman.....	21
2.2.1 Fase di aggiornamento	21
2.2.2 Fase di predizione	23
2.2.3 Filtro di Kalman esteso	23
Capitolo 3 Modelli	25
3.1 Modello complessivo del braccio	26
3.2 Modello multi-body del braccio.....	27
3.2.1 Set-up	27
3.2.2 Modello in Simscape	29
3.2.3 Output	33
3.2.4 Input	33
3.2.5 Modello del sensore	34

3.3	Cinematica diretta e inversa	34
3.3.1	Cinematica diretta	35
3.3.2	Cinematica inversa.....	36
3.4	Modello dinamico del sistema	37
3.4.1	Algoritmo UIEKF	39
3.5	Modello cinematico del sistema	43
3.5.1	Algoritmo KF	44
Capitolo 4	Sperimentazione e sviluppo degli algoritmi.....	47
4.1	Set-up sperimentale	48
4.1.1	Determinazione sistema di riferimento World.....	49
4.1.2	Elaborazione dei dati Kinect.....	51
4.1.3	Utilizzo di Kinect singola o doppia	53
4.1.4	Prove eseguite e acquisizione dati Kinect.....	53
4.2	Problematiche delle prove sperimentali	55
4.2.1	Problematica dell'intervallo temporale Δt di Kinect.....	55
4.2.2	Problematica delle lunghezze misurate.....	55
4.3	Confronto modello dinamico e cinematico	59
4.3.1	Confronto basato sui dati simulati	59
4.3.2	Confronto basato sui dati sperimentali	65
4.3.3	Conclusioni sul modello da adottare.....	68
4.4	Risposta dinamica e matrici di covarianza.....	68
4.4.1	Effetto matrice di covarianza di processo Q.....	71
4.4.2	Effetto matrice di covarianza di misura R	73
4.4.3	Risposta dinamica con dati sperimentali	74
4.5	Effetto inserimento filtro passa-basso.....	79
4.6	Analisi sull'imposizione delle lunghezze fisse.....	82
4.6.1	Problematica della taratura delle lunghezze fisse	86
4.7	Filtraggio della posizione del giunto spalla.....	88
4.8	Sistema di riferimento spalla rotante real-time	91
4.9	Confronto con dati Optitrack	94
4.9.1	Elaborazione dati Optitrack	95
4.9.2	Analisi della traiettoria del giunto polso.....	96

4.9.3	Analisi modulo delle velocità	97
4.9.4	Analisi direzione e verso delle velocità	100
4.10	Confronto con nuvola di punti Kinect	102
Capitolo 5	Modello completo	105
5.1	Input da fornire al modello	107
5.2	Tempi di calcolo	108
Conclusioni	111
Bibliografia	113

Introduzione

Attualmente, l'utilizzo di robot collaborativi è oggetto di largo interesse. La collaborazione uomo-robot combina i vantaggi dei robot industriali, caratterizzati da forza e accuratezza, con quella degli uomini, con la loro flessibilità [1]. Dovendo condividere lo stesso spazio, i robot devono essere dotati di un'intelligenza capace di fornire indicazioni su cosa accade intorno a loro, in particolar modo sui movimenti che compiono gli operatori, sia per questioni di sicurezza che di interazione per fini produttivi.

Si rende necessaria la presenza di opportuni sensori in grado di fornire informazioni al robot riguardo l'ambiente circostante. Lo sviluppo nel 2010 di Microsoft Kinect, un sensore di visione e di profondità (RGB-D) dal basso costo sviluppato inizialmente per la piattaforma di gioco di XBOX360, ha aperto nuove opportunità nella risoluzione di problemi di *computer vision*. Tale sensore, oltre a fornire le coordinate di una nuvola di punti che rappresentano lo spazio osservato (dati grezzi), attraverso il suo *Software Development Kit* (SDK) fornisce una serie di funzioni proprietarie tra cui il rilevamento della presenza di un essere umano con l'estrapolazione di uno scheletro [2]. Il segnale fornito è però affetto da scarsa accuratezza e lo scheletro presenta nel tempo un movimento nervoso e discontinuo con inattese vibrazioni, di difficile impiego come segnale di riferimento in varie applicazioni.

L'obiettivo del presente lavoro è lo sviluppo di algoritmi di rilevamento (*tracking*) e predizione del movimento dell'uomo mediante il sensore Kinect, in modo particolare mediante lo scheletro che esso è in grado di fornire. Si vuole

quindi filtrare e rilevare con un certo anticipo la posizione dell'uomo, nello specifico la posizione del braccio.

Tali algoritmi saranno utili per rendere disponibili le informazioni sul movimento dell'uomo in applicazioni di *hand following* e *collision avoidance* inseriti in un contesto di robotica collaborativa. Questo lavoro fa infatti parte di un progetto più ampio, con altri lavori di tesi paralleli, finalizzati al funzionamento e controllo in sicurezza di un robot collaborativo UR3 della Universal Robots per applicazioni avanzate.

Il tema della rilevazione e predizione o, più in generale, del riconoscimento dei movimenti dell'uomo, è stato affrontato in varie declinazioni in letteratura. Il crescente impiego di algoritmi di *machine learning* trova spazio nel riconoscimento di *gesture* a partire da immagini fornite da fotocamere digitali RGB [3], dallo scheletro fornito dal sensore Kinect [4] [5] oppure unicamente dalle informazioni di profondità dello stesso [6]. Queste applicazioni forniscono un riconoscimento e una classificazione di una *gesture* con una certa fiducia entro un set predefinito di *gesture* e richiedono un database di partenza non indifferente per permettere agli algoritmi di funzionare efficacemente. Tali algoritmi possono fornire al robot informazioni sull'attività compiuta dall'uomo o sull'attività da compiere in funzione della posa dell'uomo, ma non forniscono un'esaustiva informazione sulla posizione effettiva dell'uomo e su come prevederla. Per tale scopo in [7] viene utilizzato un approccio statistico, basato su un database, che fornisce la predizione del possibile spostamento dell'uomo. Anche questo approccio, oltre ad essere dispendioso, non si adatta alla problematica oggetto di esame.

Un approccio efficace di largo uso in robotica nell'ambito della visione è l'utilizzo del filtro di Kalman (*Kalman filter*, KF) [8]. Attraverso un modello del sistema oggetto di osservazione si ottiene una stima attendibile dello stato di moto. Si ha così un effetto filtrante, adatto ad essere utilizzato per migliorare il segnale di Kinect [9] e ad ottenere non solo un tracciamento della traiettoria percorsa dall'uomo più veritiera rispetto la misura, ma anche una possibile stima predittiva degli stati successivi.

Queste considerazioni sono alla base di un lavoro condotto precedentemente da G. Biondi, all'interno dello stesso progetto di robotica collaborativa, che vedeva

l'elaborazione, in ambiente Matlab, di un filtro di Kalman con un modello dinamico non lineare del sistema, ovvero del braccio. Non disponendo di dati sperimentali, era stato elaborato un modello *multi-body* su Simscape, con il quale effettuare delle prime analisi, che simulasse sia i movimenti di un braccio umano che il segnale registrato da Kinect con relativo rumore.

L'arto viene modellizzato con tre giunti (spalla, gomito, polso) uniti da due collegamenti (braccio e avambraccio) e avente 4 gradi di libertà. Il braccio umano viene quindi trattato come fosse un robot antropomorfo e descritto secondo le nozioni di robotica [10]. In diverse applicazioni il filtro di Kalman è utilizzato direttamente nello spazio operativo, magari utilizzando le informazioni relative a un solo giunto dello scheletro [11]. Nel modello che si tratterà, il filtro di Kalman è applicato alle variabili di giunto, nello spazio giunti, le quali rappresenteranno lo stato del sistema. Questa scelta permette al modello di integrare le informazioni sui possibili movimenti che un braccio umano può compiere, grazie anche alle limitazioni imposte alle variabili di giunto compatibilmente coi movimenti che un braccio reale è in grado di eseguire.

Nel presente lavoro si ottiene un'ulteriore indicazione sull'importanza di operare a livello giunti: il passaggio da spazio operativo a spazio giunti permette azione filtrante nell'imposizione di lunghezze fisse di braccio e avambraccio. Lo scheletro fornito da Kinect, infatti, essendo soggetto a disturbi, non conserva necessariamente la lunghezza dei collegamenti fra i giunti.

Con questo lavoro si ha lo sviluppo di un algoritmo scritto in Matlab, funzionante in real-time, che presenta un filtro di Kalman con un modello cinematico lineare del sistema. Il modello sviluppato semplifica e supera il modello dinamico non lineare di partenza, con un forte guadagno in termini di velocità di calcolo a fronte di un'accuratezza nella stima pressoché invariata. Si ottiene un modello complessivo in grado di tracciare indifferentemente il braccio destro e sinistro in diverse condizioni di orientazione del busto. Il contributo predittivo può essere fornito sia dai vettori di velocità istantanea che dalla predizione dello stato insita nel funzionamento del filtro di Kalman.

Si utilizza in tutto il lavoro la notazione formale presente in [10].

La presente dissertazione si sviluppa in cinque capitoli. Il primo capitolo fornisce una panoramica sul problema del rilevamento ed elaborazione del movimento dell'uomo. Vengono mostrate le varie tipologie di sensori comunemente usati per l'attività di misura di dati grezzi da utilizzare per l'identificazione del movimento che interessa per lo studio. Viene dunque approfondito il funzionamento del sensore di profondità utilizzato Kinect. Si trattano poi le fasi salienti che caratterizzano un classico processo di riconoscimento di una *gesture*. In particolare si fa chiarezza sulla diversa natura di processi di *tracking* e di classificazione di una *gesture*, mettendo in luce le differenze applicative nell'uso di algoritmi di *machine learning* e *model-based*.

Il secondo capitolo mostra come sia possibile stimare lo stato di sistemi stocastici mediante filtri Bayesiani, famiglia in cui rientrano i filtri di Kalman. Si stabiliscono dunque le basi teoriche necessarie al proseguo della trattazione.

Il terzo capitolo presenta nel dettaglio tutti i modelli e sotto-modelli utilizzati. Si mostra dunque come venga modellizzato il braccio umano e la sua relativa implementazione. Tale modello viene utilizzato in Matlab per completare il modello del sistema ed è ricostruito in Simscape, con un modello *multi-body*, per realizzare la simulazione del sistema osservato e il modello dello strumento di misura. Inoltre, si tratteranno i modelli cinematico e dinamico del sistema oggetto di indagine e come questi vengano impiegati dal filtro di Kalman.

Il quarto capitolo mostra le indagini sperimentali effettuate, le relative conclusioni tratte e le modifiche del modello che ne derivano. Per prima cosa si illustrano le modalità di esecuzione delle prove e il setup sperimentale. Sono poi ripercorsi cronologicamente i miglioramenti fatti a fronte dei risultati delle prove sperimentali. Vengono trattate le motivazioni alla base della scelta finale di un modello cinematico, la scelta di parametri specifici che influenzano le performance dell'algoritmo, scelte particolari sulla manipolazione dei dati e confronti dei risultati ottenuti con il sistema osservato misurato con diversi strumenti come il sensore Optitrack o la stessa nuvola di punti fornita da Kinect.

Il quinto capitolo presenta infine il modello completo con le performance in termini di velocità di calcolo ottenute.

Capitolo 1

Rilevamento ed elaborazione del movimento dell'uomo

In un ambiente di collaborazione uomo-robot (*Human-Robot Collaboration*, HRC), l'acquisizione e riconoscimento dei movimenti dell'uomo assume un ruolo fondamentale, sia in termini di sicurezza (*collision avoidance*) che in termini di iterazione uomo-robot (*target following*). Con riferimento a [12], si identificano le fasi caratterizzanti di un processo di riconoscimento delle *gesture* (*Gesture Recognition*), intese in generale sia come gesti statici che come movimenti complessi dell'intero corpo o parti di esso.

Nello specifico si individuano cinque fasi legate ad un processo di *gesture recognition* per HRC: *sensor data collection*, *gesture acquisition*, *gesture tracking*, *gesture classification* e *gesture mapping*, spiegate brevemente di seguito e mostrate in Figura 1.1.

- *Sensor data collection*: il sensore cattura i dati grezzi (*raw data*).
- *Gesture acquisition*: dai dati grezzi, per ogni istante, si ottengono le informazioni su ogni *gesture*.
- *Gesture tracking*: si rileva il movimento della *gesture* nel tempo. Per *gesture* statiche questa fase non è necessaria.

- *Gesture classification*: il movimento viene classificato in categorie di *gesture* predefinite.
- *Gesture mapping*: il risultato viene tradotto in un comando per il robot.

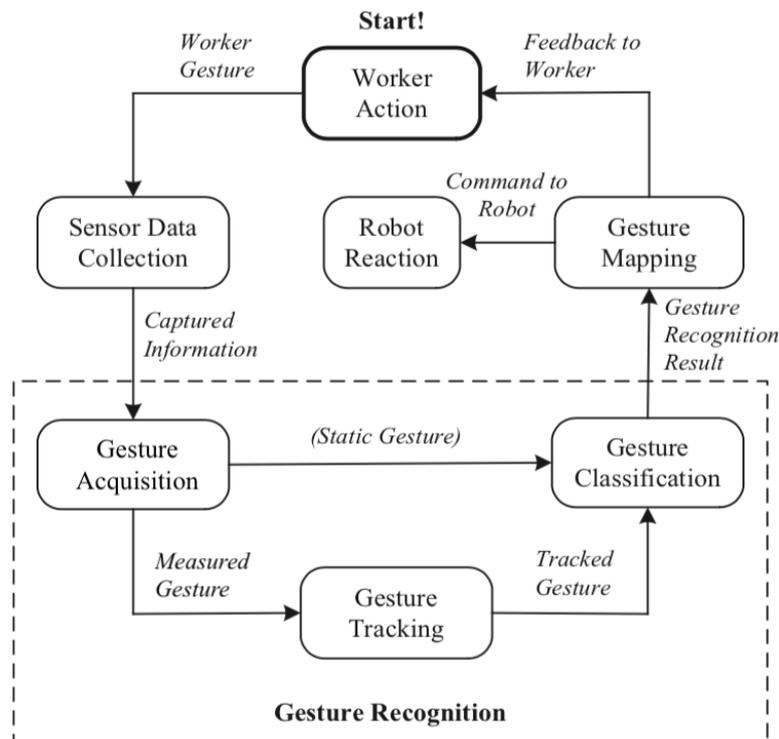


Figura 1.1 Processo di Gesture Recognition per HRC

Come mostrato, il primo punto su cui focalizzare l'attenzione è la scelta del sensore in grado di catturare le informazioni utili, ovvero il movimento dell'uomo. Saranno dunque presentati le principali tipologie di sensori utilizzati a tale fine, per trattare più nel dettaglio il sensore di profondità Kinect e le motivazioni che hanno spinto al suo utilizzo nel presente lavoro. Verrà trattato nello specifico il processo di *gesture recognition* con l'analisi delle fasi di cui è costituito, da cui trarre importanti considerazioni per lo sviluppo del lavoro.

1.1 Sensori

A monte del processo di *gesture recognition* vi è l'acquisizione di dati grezzi mediante sensori. Come presentato in Figura 1.2, è possibile individuare due macro-

categorie: sensori *image-based* e sensori *non-image based*, ovvero basati sull'acquisizione o meno di un'immagine attraverso un sistema ottico.

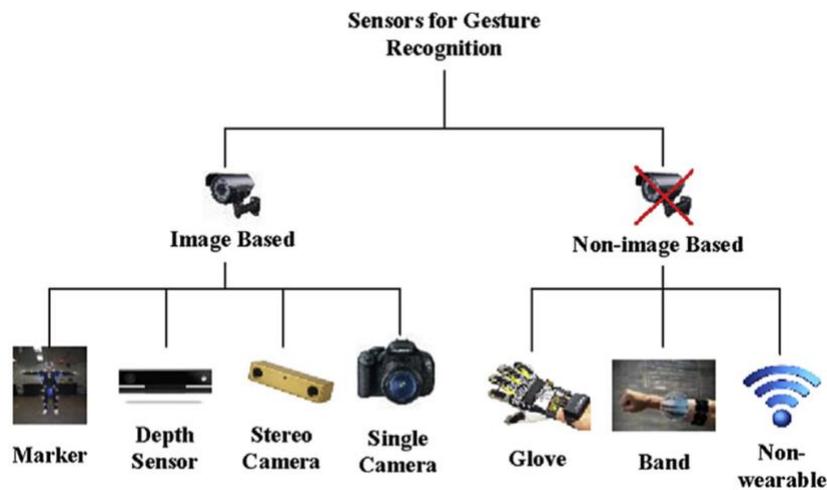


Figura 1.2 Tipologie di sensori per gesture recognition

I sensori *non-image-based*, tenuti in secondo piano per lungo tempo sono in crescita per merito dello sviluppo di sensori come i MEMS. Questa tipologia di sensori si divide a sua volta in due sotto-categorie: indossabili (mediante guanti o fasce) e non indossabili. I guanti richiedono generalmente una connessione con cavo, accelerometri e giroscopi e le procedure di calibrazione e setup sono complesse. Le fasce invece sono dotate di tecnologia wireless con sensori elettromiografici che permettono di avere la mano e dita libere essendo a contatto con il solo polso. Al contrario, i sensori non indossabili possono registrare un movimento senza venire a contatto con il corpo umano: si può trattare di sensori a segnali in radio frequenza con emettitore e ricevitore, adatti a rilevamenti a breve distanza. La precisione di questi sensori ad oggi non è accettabile per sistemi di HRC, ma è in grande crescita.

I sensori *image-based* usano fotocamere per “vedere” le *gesture*, come del resto fanno gli occhi umani. Si passano al vaglio le seguenti tecnologie: sensori che utilizzano *marker*, fotocamere singole, fotocamere stereoscopiche e sensori di profondità.

I sensori che utilizzano *marker* utilizzano più fotocamere sincronizzate. Ogni fotocamera cattura immagini 2D quindi, attraverso una procedura di triangolazione,

vengono calcolate le posizioni 3D di ogni *marker* collocato sul target. Un sensore di questo tipo molto diffuso è Optitrack, capace di garantire un'ottima accuratezza della misura [13] rispetto ad altre tecnologie *image-based*.

L'uso di una fotocamera singola permette il riconoscimento di *gesture* attraverso diversi algoritmi. Informazioni 3D possono essere raccolte anche mediante una fotocamera stereoscopica, ovvero una doppia fotocamera, attraverso triangolazione. Questo tipo di tecnologia ha migliorato nel tempo la sua robustezza, ma pecca ancora in fase di calibrazione e complessità computazionale.

I sensori di profondità rappresentano una facile ed economica soluzione per la *gesture recognition*. In generale si intende sensore di profondità un sensore che non sia stereografico. Difatti presenta vantaggi rispetto le fotocamere stereografiche come la sensibilità alle condizioni di illuminazione e le difficoltà di calibrazione. L'output fornito è una informazione 3D. Una delle tecnologie più utilizzate è la *Time-of-Flight* (ToF), basata sul principio fondamentale di identificare il tempo di viaggio della luce è stata impiegata anche in Kinect v2.

1.1.1 Kinect: hardware



Figura 1.3 Kinect v1 (a) e Kinect v2 (b)

Kinect è il sensore di profondità più utilizzato grazie il suo basso costo ed è presente in due versioni mostrate in Figura 1.3. Il sensore di profondità consiste in un proiettore laser a infrarossi e una fotocamera a infrarossi che insieme costruiscono una mappa di profondità. La tecnologia alla base di tale procedura è differente fra le due versioni: Kinect v1 utilizza una codifica di luce strutturata (*structured light*),

Kinect v2 utilizza la ToF consentendo migliori performance [14]. Come è mostrato in Figura 1.4, in realtà Kinect è uno strumento costituito da un insieme di sensori avendo, oltre quello di profondità con emettitore e fotocamera a infrarossi, una fotocamera RGB e un microfono. È presente, in aggiunta, un motore per l'inclinazione.

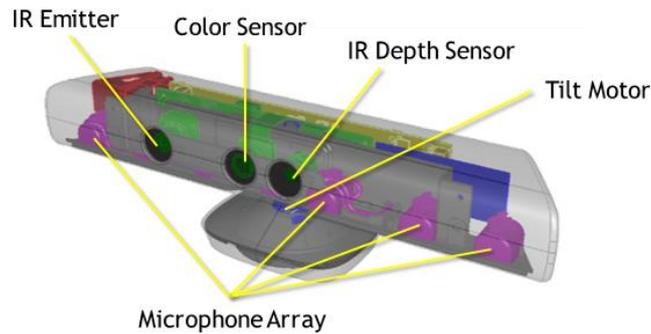


Figura 1.4 Componenti di Kinect v1

In Tabella 1.1 si evidenziano le differenze fra le due versioni di Kinect, con le migliori performance di Kinect v2 in risoluzione e ampiezza di campo visivo [14].

Tabella 1.1 Specifiche di Kinect v1 e Kinect v2

Caratteristiche	Kinect v1	Kinect v2
Tipologia sensore profondità	Structured light	Time of Flight (ToF)
Risoluzione fotocamera RGB	640 × 480, 30 fps	1920 × 1080, 30 fps
Risoluzione sensore infrarossi (IR)	320 × 240, 30 fps	512 × 424, 30 fps
Campo visivo immagine RGB	62° × 48,6°	84,1° × 53,8°
Campo visivo mappa di profondità	57° × 43°	60° × 70°

In Figura 1.5 viene mostrato un esempio di un frame di acquisizione di nuvola di punti, in cui ogni punto (o pixel) ha la sua informazione di profondità.

La qualità delle immagini di profondità soffrono di limitata accuratezza e stabilità dovuti a buchi e valori inconsistenti di profondità. I buchi di profondità possono verificarsi ad esempio su superfici riflettenti. Inoltre, la profondità di alcuni pixel spesso continua a cambiare anche se la scena è statica. In aggiunta, accuratezza e precisione risentono notevolmente della distanza dallo strumento: più il target è lontano, più la sua misura non è accurata [15].



Figura 1.5 Esempio di nuvola di punti fornita come output da Kinect

1.2 Gesture Acquisition

Dopo che i dati grezzi siano stati catturati dal sensore, il primo passo che porta alla *gesture recognition* è la *gesture acquisition*. In tale fase si rilevano e acquisiscono le informazioni sulle *gesture* a partire dai dati grezzi. Alcune diffuse tecnologie che risolvono questo problema sono basate su caratteristiche visive (*visual features*), algoritmi di apprendimento (*machine learning algorithms*) e modello dello scheletro (*skeleton model*).

Le caratteristiche visive su cui ci si può basare per rilevare una *gesture* sono i colori, caratteristiche locali oppure forme e contorni. Questo tipo di approccio può essere utilizzato sia per *gesture* statiche che in movimento attraverso la tecnologia *Optical Flow*, ha basso costo computazionale, ma scarsa qualità e affidabilità dei risultati essendo molto sensibile alle condizioni esterne di illuminazione.

Gli algoritmi di apprendimento basano il loro funzionamento sul rilevamento di varie caratteristiche visive e sono utilizzati maggiormente per *gesture* statiche. Per il loro funzionamento richiedono un alto costo computazionale in quanto è necessaria la preparazione di un ampio database su cui si possa basare il riconoscimento. Sia approcci basati su *machine learning* che su caratteristiche visive trovano generalmente impiego in accoppiata con fotocamere singole.

L'approccio che utilizza un modello di scheletro umano è spesso utilizzato per indagare pose e movimenti del corpo umano. Infatti, per identificare le *gesture*, un

modello dettagliato del corpo umano spesso è inutile. Lo scheletro conserva infatti solo le informazioni più importanti sulla posa o movimento del corpo umano aiutando a semplificare la problematica del riconoscimento delle *gesture*. Questa soluzione è diventata attrattiva per i sensori di profondità.

1.2.1 Kinect: estrapolazione dello scheletro

Kinect, con il suo SDK, è in grado di fornire in output già le informazioni sullo scheletro umano. Il modello dello scheletro è costituito da punti, rappresentanti i giunti che caratterizzano la posizione dell'uomo. Kinect v2 individua 25 giunti e può individuare contemporaneamente fino a sei scheletri. Nella Tabella 1.2 si mostrano le differenze fra le due versioni di Kinect in termini di riconoscimento dello scheletro. In Figura 1.6 si mostrano i giunti che Kinect v2 è in grado di individuare, mentre in Figura 1.7 viene mostrato un esempio di riconoscimento dello scheletro, in questo caso relativo solo alla sua parte superiore.

Tabella 1.2 Specifiche di Kinect v1 e Kinect v2

Caratteristiche	Kinect v1	Kinect v2
Giunti dello scheletro	20	25
Massimo numero di scheletri rilevabili	2	6

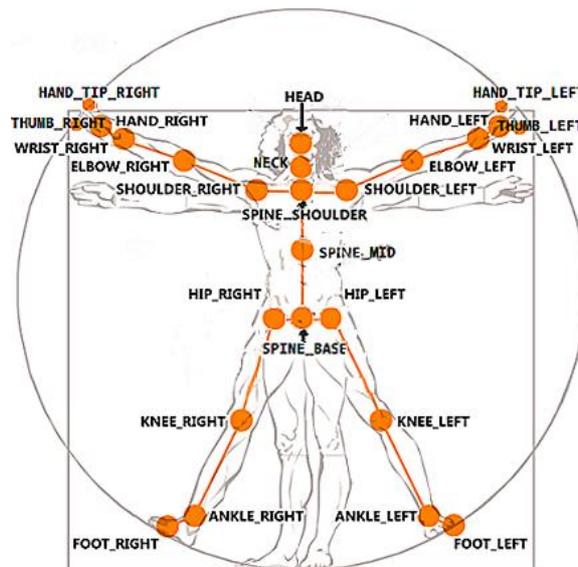


Figura 1.6 Giunti dello scheletro fornito da Kinect v2

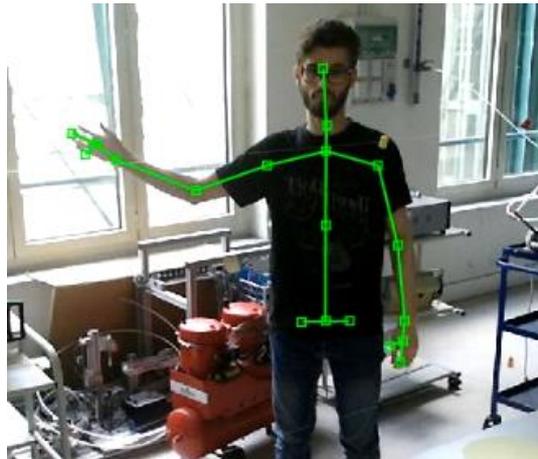


Figura 1.7 Esempio di parte superiore di scheletro, posizionato sull'immagine RGB, fornito come output da Kinect v2

Kinect è dunque quello strumento che ottempera alle fasi di *sensor data collection*, attraverso i dati grezzi collezionati con le informazioni di profondità e RGB, e *gesture acquisition*, attraverso lo scheletro che rappresenta le posizioni del corpo umano. Kinect rappresenta lo strumento ideale da inserire in un contesto di HRC, dal basso costo e che fornisca, attraverso algoritmi proprietari, informazioni riguardo punti particolari del corpo umano che possono essere utilizzati in diverse applicazioni. Lo scheletro viene individuato però sulla base di informazioni che, come già detto, sono affette da scarsa accuratezza. Gli algoritmi che permettono di identificare uno scheletro a loro volta introducono ulteriori errori producendo uno scheletro che nel tempo ha un comportamento poco stabile, con giunti che a volte assumono posizioni non naturali che non rispecchiano la realtà.

1.3 Gesture Tracking

Nella *gesture recognition* il concetto di rilevamento (*tracking*) è usato in modo diverso rispetto alla letteratura: è definito come il processo di trovare corrispondenze temporali fra i *frame*. Nello specifico ci si concentra sul problema del tracciamento di *gesture* continue che associa la *gesture* rilevata e acquisita al frame precedente con quella al frame corrente. Chiaramente per *gesture* statiche questa fase non è necessaria. In Figura 1.8 viene mostrato un esempio di *gesture tracking*: sulle basi delle informazioni sulle *gesture* ai frame precedenti viene effettuata una predizione per il frame successivo, una volta che è disponibile il

frame con la gesture acquisita viene effettuato un confronto che porta ad informazioni sulla gesture aggiornate. Si effettua dunque una miglior stima sulle posizioni delle *gesture*.

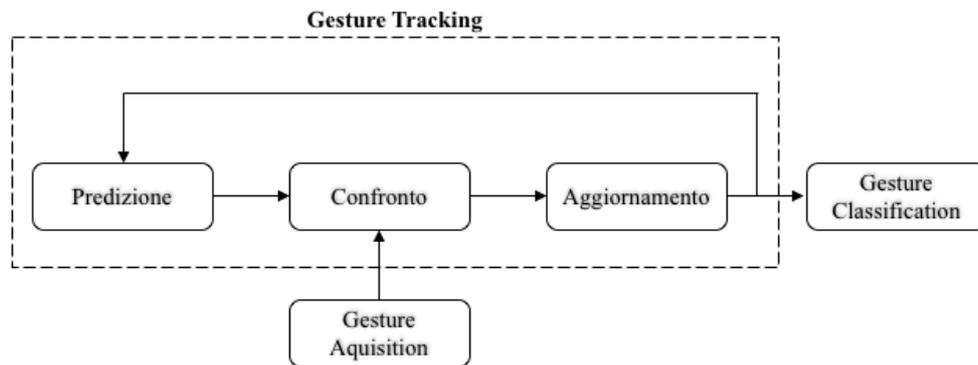


Figura 1.8 Esempio di processo di Gesture Tracking per ogni frame

Tra gli approcci più utilizzati vi sono lo scostamento medio (*mean shift*), il filtro di Kalman (*Kalman filter*, KF) e i filtri particellari (*particle filter*, PF).

Lo scostamento medio viene utilizzato prevalentemente per l'elaborazione di immagini RGB. Il suo funzionamento si basa sul confronto degli istogrammi di colore fra un frame e l'altro.

Il filtro di Kalman è un algoritmo ricorsivo in real-time usato per stimare in maniera ottimale lo stato di un sistema a partire da una misura nel tempo non accurata affetta da rumore. La Figura 1.8 ben rappresenta, in modo semplificato, il funzionamento di un filtro di Kalman in cui la predizione è basata su un modello fisico del sistema. Per maggiori approfondimenti si rimanda al Capitolo 2. Il filtro di Kalman risulta essere l'approccio più immediato da applicare nel caso di studio in cui si voglia tracciare la posizione dei giunti dello scheletro forniti da Kinect.

I filtri particellari possono essere visti come estensioni del filtro di Kalman. Se il funzionamento del filtro di Kalman si basa sul confronto di distribuzioni di probabilità gaussiane, nei filtri particellari è possibile utilizzarne di non gaussiane. Trovano applicazione, ad esempio, in ambienti affollati in cui è necessario tenere traccia di più lavoratori.

1.4 Gesture Classification

La *gesture classification* è l'ultima fase della *gesture recognition*. È un tipico problema di *machine learning*, in cui le *gesture* acquisite ed elaborate, vengono classificate in classi predefinite mediante l'utilizzo di un vasto database di riferimento. Può essere risolto con diversi algoritmi di *machine learning*, per citarne alcuni: *K-Nearest Neighbours*, *Hidden Markov Model*, *Support Vector Machine*, *Ensemble Method*, *Dynamic time warping*, *Artificial Neural Network*, *Deep learning*, come *Convolutional Neural Networks* e *Recurrent Neural Networks*.

Le classi, attraverso la fase di *gesture mapping*, vengono poi tradotte in comandi specifici del robot che si comporta diversamente a seconda dell'attività eseguita dall'operatore.

1.5 Riepilogo e Considerazioni

Nella ricostruzione di tutte le fasi che portano al riconoscimento di *gesture*, si è visto come Kinect costituisca un valido strumento. È possibile considerare Kinect come una black-box che, a partire dal sistema oggetto di osservazione, fornisca come output immagini RGB, nuvola di punti con informazioni di profondità e le posizioni dei giunti dello scheletro della persona individuata. Le fasi di *sensor data collection* e *gesture acquisition* vengono eseguite quindi dallo stesso strumento.

Da Kinect si è interessati maggiormente all'utilizzo delle informazioni relative allo scheletro per un'applicazione in robotica collaborativa. Si è detto come lo scheletro prodotto abbia un a comportamento nel tempo poco accurato e affetto da importante rumore.

Il filtro di Kalman sembra rappresentare una scelta efficace per effettuare una miglior stima delle posizioni dei giunti dello scheletro mediante dei modelli fisici del sistema. Attraverso il filtro di Kalman si esegue dunque una fase di *gesture tracking*. Nel seguito ci si concentrerà maggiormente sulle posizioni del braccio umano in quanto quelle di maggior interesse in ambiente collaborativo.

La fase di *gesture collection*, da eseguire mediante algoritmi di *machine learning*, e quindi *gesture mapping* non è prevista nel presente lavoro in quanto non si ha l'obiettivo di eseguire una classificazione della tipologia di *gesture*. A seconda

dell'applicazione, di *collision avoidance* o di *target following*, si andranno ad utilizzare gli algoritmi proposti in modo differente.

Si approfondisce dunque la fase di *gesture tracking*. Dal filtro di Kalman sarà poi possibile ottenere anche stime predittive, derivanti dal suo funzionamento, che possano essere utilizzate per anticipare i movimenti del robot a seconda dell'applicazione.

Capitolo 2

Teoria degli osservatori per un sistema stocastico

Il sensore di visione Kinect fornisce, come informazioni di output, le posizioni dei giunti dello scheletro. Tali misure di posizione non risultano essere accurate: si registrano nel tempo movimenti bruschi e salti che non aderiscono alla realtà da parte dello scheletro. Inoltre l'accuratezza della misura di profondità risulta essere tanto più bassa quanto più il target è lontano dalla Kinect.

Le osservazioni di Kinect possono essere corrette sulla base di previsioni prodotte da un modello fisico del sistema. Il filtro di Kalman, infatti, sulla base di un modello, effettua una predizione e, attraverso il confronto con la misura fornita, è in grado di ottenere una migliore stima della posizione dei giunti.

Il filtro di Kalman non è propriamente un filtro, ma piuttosto un osservatore di sistemi stocastici in grado di fornire delle stime sugli stati del sistema attraverso un algoritmo ricorsivo real-time basato su un modello fisico (*model-based*).

Questo capitolo fornisce i concetti fondamentali sulla teoria degli osservatori per sistemi stocastici per poi approfondire nel dettaglio il funzionamento del filtro di Kalman [16] [17].

2.1 Filtri Bayesiani

L'osservazione di qualsiasi sistema dinamico necessita della misura di quantità fisiche che, nella teoria dei sistemi, vengono spesso definite *output* e contenute in un vettore colonna \mathbf{y} . Chiaramente sono necessari uno o più sensori per avere una stima di tali quantità. Tutte le *misure* vengono collezionate in un vettore colonna \mathbf{z} . Sono valide le seguenti *equazioni di misura* per ogni istante k :

$$\mathbf{z}_k = \mathbf{y}_k + \mathbf{v}_k \quad (2.1)$$

L'*errore* della misura \mathbf{v}_k è l'incertezza dovuta al sensore utilizzato per misurare \mathbf{y}_k . Il sensore è uno stimatore del reale output del sistema e commette un errore \mathbf{v}_k , che è una variabile casuale la cui distribuzione dipende dalle caratteristiche del sensore stesso.

L'accuratezza del sensore è spesso non sufficiente e, per ottenere stime migliori, si rende necessario definire procedure matematiche basate su conoscenze a priori riguardo le dinamiche del sistema.

In aggiunta, le quantità direttamente misurabili spesso non sono sufficienti per una completa descrizione del comportamento del sistema. Le grandezze rappresentative di tutte le dinamiche del sistema sono collezionate in un vettore colonna \mathbf{x} , chiamato *vettore di stato*. Nel caso più generale in cui gli output sono collegati alla variabile di stato da una funzione h , le *equazioni di misura* diventano:

$$\mathbf{z}_k = h(\mathbf{x}_k) + \mathbf{v}_k \quad (2.2)$$

Un modo appropriato di rappresentare matematicamente le reali dinamiche del sistema, è attraverso *equazioni di stato*, che possono essere rappresentate in modo discreto mediante un sistema di equazioni dove lo stato all'istante successivo è funzione dello stato precedente attraverso il modello adottato:

$$\mathbf{x}_{k+1} = g(\mathbf{x}_k) + \mathbf{w}_k \quad (2.3)$$

dove \mathbf{w}_k è l'errore di processo. L'introduzione della variabile casuale \mathbf{w}_k rappresenta il fatto che le reali dinamiche del sistema non sono realmente conosciute e non sono integralmente spiegate dal modello. Inoltre si tiene conto di imprecisioni dovute alla rappresentazione discreta di un processo continuo.

Le equazioni di stato sono molto utili per identificare una stima del vettore di stato in quanto hanno in sé informazioni riguardo il comportamento del sistema atteso. Si fa presente che sia le equazioni di misura che quelle di stato non sono equazioni deterministiche a causa delle variabili casuali \mathbf{v}_k e \mathbf{w}_k . Per questa ragione le incertezze nella descrizione del sistema dovrebbero essere trattate mediante la teoria delle probabilità. Il comportamento nel tempo della misura riflette la sua funzione di probabilità congiunta e assumendo che non ci sia correlazione fra due differenti istanti del rumore di misura \mathbf{v}_k si ha:

$$p(\mathbf{v}^N) = p(\mathbf{v}_0, \mathbf{v}_1, \dots, \mathbf{v}_n) = p(\mathbf{v}_0)p(\mathbf{v}_1) \dots p(\mathbf{v}_n) \quad (2.4)$$

Dicasi lo stesso per il rumore di processo \mathbf{w}_k .

Attraverso l'Eq. (2.3) è possibile stimare il vettore di stato attraverso un modello matematico prelevando dalla distribuzione di probabilità congiunta del rumore di processo assunta come nota. Successivamente la misura acquisita produce ulteriori informazioni sullo stato del sistema mediante l'Eq. (2.2) in cui anche il rumore di misura è pescato dalla sua distribuzione di probabilità congiunta.

Formalmente la combinazione di queste informazioni viene espressa in termini di funzione densità di probabilità dalla formula di Bayes per probabilità condizionata:

$$p(\mathbf{x}_k | \mathbf{z}^k) = \frac{p(\mathbf{z}_k | \mathbf{x}_k)p(\mathbf{x}_k | \mathbf{z}^{k-1})}{p(\mathbf{z}_k | \mathbf{z}^{k-1})} \quad (2.5)$$

dove $p(\mathbf{x}_k | \mathbf{z}^k)$ è la probabilità condizionata di \mathbf{x}_k data l'intera storia di misure degli output del sistema, rappresentata dalla funzione di probabilità congiunta $p(\mathbf{z}^k)$. $p(\mathbf{z}_k | \mathbf{x}_k)$ è la probabilità condizionata della misura corrente \mathbf{z}_k dato il risultato corrente dello stato \mathbf{x}_k . È evidente come questa probabilità sia direttamente

influenzata mediante l'Eq. (2.2) dalla distribuzione del rumore di misura $p(\mathbf{v}_k)$. $p(\mathbf{x}_k|\mathbf{z}^{k-1})$ è la funzione densità di probabilità predetta di \mathbf{x}_k note tutte le misure passate eccetto quella corrente. Ricorrendo all'Eq. (2.3) e alla distribuzione di probabilità del rumore di processo \mathbf{w}_k , si ha:

$$p(\mathbf{x}_k|\mathbf{z}^{k-1}) = \int p(\mathbf{x}_k|\mathbf{x}_{k-1})p(\mathbf{x}_{k-1}|\mathbf{z}^{k-1})d\mathbf{x}_{k-1} \quad (2.6)$$

dove $p(\mathbf{x}_k|\mathbf{x}_{k-1})$ è la funzione densità di probabilità della transizione legata dunque al modello del sistema.

Infine la costante di normalizzazione $p(\mathbf{z}_k|\mathbf{z}^{k-1})$ può essere ottenuta dalla seguente convoluzione:

$$p(\mathbf{z}_k|\mathbf{z}^{k-1}) = \int p(\mathbf{z}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{z}^{k-1})d\mathbf{x}_k \quad (2.7)$$

Le equazioni (2.5) e (2.6) definiscono la formulazione generale di *filtro Bayesiano*. Si faccia presente che esso non è propriamente un filtro, ma piuttosto uno stimatore del comportamento nel tempo della distribuzione di probabilità delle variabili di stato. È dunque un osservatore dello stato del sistema trattato da variabile casuale.

La stima è composta da due operazioni. La prima è la *predizione*, rappresentata dall'Eq. (2.6), in cui il modello del sistema assunto produce una informazione a-priori riguardo la distribuzione di probabilità dello stato all'istante successivo, prima che sia acquisita la misura dell'output del sistema. In seguito, quando la distribuzione di probabilità relativa alla misura è disponibile, si ha l'*aggiornamento* della distribuzione dello stato stimata a-priori mediante l'Eq. (2.5).

È possibile ottenere un osservatore più semplice con qualche assunzione. Generalmente si assume il processo rappresentato dall'Eq. (2.3) sia una *catena di Markov*. Ciò implica che:

$$p(\mathbf{x}_k|\mathbf{x}^{k-1}) = p(\mathbf{x}_k|\mathbf{x}_{k-1}) \quad (2.8)$$

Ovvero che la funzione densità di probabilità dello stato all'istante k è influenzata solo dallo stato all'istante precedente $k - 1$.

Assumendo inoltre processo e misura come lineari e gaussiani, è possibile derivare la formula per uno degli osservatori più apprezzati: il *filtro di Kalman* (KF). Il motivo per cui è diffuso in molte applicazioni è da ricercare nella sua semplicità, efficacia e robustezza.

2.2 Filtro di Kalman

L'assunzione di linearità del sistema porta alle seguenti rappresentazioni:

$$\mathbf{x}_{k+1} = \mathbf{G}_k \mathbf{x}_k + \mathbf{w}_k \quad (2.9)$$

$$\mathbf{z}_k = \mathbf{H}_k \mathbf{x}_k + \mathbf{v}_k \quad (2.10)$$

dove \mathbf{F}_k è una matrice quadrata di dimensione n_x , e \mathbf{H}_k è una matrice ($n_z \times n_x$).

Inoltre sono necessarie una serie di assunzioni, trattate di seguito. I rumori di processo e misura sono rumori bianchi e all'istante k non hanno correlazione con loro stessi ad un altro istante. Si assume che i rumori siano gaussiani a media nulla. Il rumore di processo e di misura sono fra loro mutuamente indipendenti.

Si introducono le matrici di covarianza, riferite all'istante k , rispettivamente \mathbf{Q}_k del rumore di processo e \mathbf{R}_k del rumore di misura.

Infine le equazioni di stato rappresentano una catena di Markov e lo stato all'istante k è una variabile casuale con media $\bar{\mathbf{x}}_k$ e covarianza \mathbf{P}_k .

2.2.1 Fase di aggiornamento

La fase di aggiornamento di un filtro di Kalman viene derivata dalla formula di Bayes descritta dall'Eq. (2.5). Il calcolo parte dalla conoscenza della funzione densità di probabilità calcolata nella fase di predizione date le misure precedenti. Si ricorda che trattandosi di una catena di Markov, vale l'Eq. (2.8). Viene presa dunque a riferimento solo la misura all'istante immediatamente precedente.

$$p(\mathbf{x}_k | \mathbf{z}^{k-1}) = \mathcal{N}(\mathbf{x}_k; \bar{\mathbf{x}}_k^-, \mathbf{P}_k^-) \quad (2.11)$$

L'Eq. (2.11) indica che la distribuzione a priori della variabile casuale \mathbf{x}_k è una gaussiana, caratterizzata da media $\bar{\mathbf{x}}_k^-$ e covarianza \mathbf{P}_k^- . Il simbolo " - " posto ad apice indica che la stima è basata unicamente dalla predizione del modello e che non è stata aggiornata dal dato misurato. La notazione \mathcal{N} indica una gaussiana multivariata con una certa media e covarianza.

Con riferimento all'Eq. (2.10) si ha:

$$p(\mathbf{z}_k|\mathbf{x}_k) = \mathcal{N}(\mathbf{z}_k; \mathbf{H}_k\mathbf{x}_k, \mathbf{R}_k) \quad (2.12)$$

Infine dalle Eq. (2.11) e (2.12), ricordando l'Eq. (2.7), si ottiene la costante di normalizzazione:

$$p(\mathbf{z}_k|\mathbf{z}^{k-1}) = \mathcal{N}(\mathbf{z}_k; \mathbf{H}_k\bar{\mathbf{x}}_k^-, \mathbf{H}_k\mathbf{P}_k^-\mathbf{H}_k^T + \mathbf{R}_k) \quad (2.13)$$

La formula di Bayes come dalla Eq. (2.5) diventa quindi:

$$p(\mathbf{x}_k|\mathbf{z}^k) = \frac{\mathcal{N}(\mathbf{z}_k; \mathbf{H}_k\mathbf{x}_k, \mathbf{R}_k)\mathcal{N}(\mathbf{x}_k; \bar{\mathbf{x}}_k^-, \mathbf{P}_k^-)}{\mathcal{N}(\mathbf{z}_k; \mathbf{H}_k\bar{\mathbf{x}}_k^-, \mathbf{H}_k\mathbf{P}_k^-\mathbf{H}_k^T + \mathbf{R}_k)} \quad (2.14)$$

Anche la distribuzione densità di probabilità dello stato aggiornata è una gaussiana ed è possibile scriverla come:

$$p(\mathbf{x}_k|\mathbf{z}^k) = \mathcal{N}(\mathbf{x}_k; \bar{\mathbf{x}}_k^+, \mathbf{P}_k^+) \quad (2.15)$$

dove il simbolo " + " posto ad apice indica la stima dello stato aggiornata.

A seguito di passaggi algebrici che tengono conto della definizione di funzione di probabilità di una gaussiana multivariata [16] è possibile ricavare le espressioni finali utilizzate nella fase di aggiornamento di un filtro di Kalman:

$$\mathbf{K}_k = \mathbf{P}_k^-\mathbf{H}_k^T(\mathbf{H}_k\mathbf{P}_k^-\mathbf{H}_k^T + \mathbf{R}_k)^{-1} \quad (2.16)$$

$$\bar{\mathbf{x}}_k^+ = \bar{\mathbf{x}}_k^- + \mathbf{K}_k(\mathbf{z}_k - \mathbf{H}_k\bar{\mathbf{x}}_k^-) \quad (2.17)$$

$$\mathbf{P}_k^+ = \mathbf{P}_k^- + \mathbf{K}_k\mathbf{H}_k\mathbf{P}_k^- \quad (2.18)$$

dove la matrice \mathbf{K}_k è nota come *guadagno di Kalman*.

2.2.2 Fase di predizione

Per ricavare le espressioni di predizione si fa riferimento all'Eq. (2.6) ottenendo:

$$\mathcal{N}(\mathbf{x}_{k+1}; \bar{\mathbf{x}}_{k+1}^-, \mathbf{P}_{k+1}^-) = \int \mathcal{N}(\mathbf{x}_k; \bar{\mathbf{x}}_k^+, \mathbf{P}_k^+) \mathcal{N}(\mathbf{x}_{k+1}; \mathbf{G}_k \mathbf{x}_k, \mathbf{Q}_k) d\mathbf{x}_k \quad (2.19)$$

Come in precedenza, si saltano i passaggi algebrici che portano alla formulazione finale delle espressioni di interesse della fase di predizione di un filtro di Kalman, presentate di seguito.

$$\bar{\mathbf{x}}_{k+1}^- = \mathbf{G}_k \bar{\mathbf{x}}_k^+ \quad (2.20)$$

$$\mathbf{P}_{k+1}^- = \mathbf{G}_k \mathbf{P}_k^+ \mathbf{G}_k^T + \mathbf{Q}_k \quad (2.21)$$

2.2.3 Filtro di Kalman esteso

Se il sistema osservato presenta un comportamento non lineare, l'uso del filtro di Kalman non è appropriato. Ciononostante, dinamiche non lineari sono approssimabili da un modello lineare per un certo intervallo di tempo attraverso la derivata locale della funzione che descrive le non linearità del sistema. Nello specifico le Eq. (2.2) e (2.3) sono approssimabili con:

$$\mathbf{x}_{k+1} \simeq g(\bar{\mathbf{x}}_k^+) + \mathbf{G}_k (\mathbf{x}_k - \bar{\mathbf{x}}_k^+) \quad (2.22)$$

$$\mathbf{z}_k \simeq h(\bar{\mathbf{x}}_k^-) + \mathbf{H}_k (\mathbf{x}_k - \bar{\mathbf{x}}_k^-) \quad (2.23)$$

dove

$$\mathbf{G}_k = \left. \frac{\partial g}{\partial \mathbf{x}_k} \right|_{\mathbf{x}_k = \bar{\mathbf{x}}_k^+} \quad (2.24)$$

$$\mathbf{H}_k = \left. \frac{\partial h}{\partial \mathbf{x}_k} \right|_{\mathbf{x}_k = \bar{\mathbf{x}}_k^-} \quad (2.25)$$

Ciò porta, attraverso la formula di Bayes, ad ottenere equazioni formalmente identiche a quelle del filtro di Kalman. L'insieme delle formule così ottenute rappresentano il filtro di Kalman esteso (*extended Kalman filter*, EKF). Con esso il modello non lineare di un sistema è approssimato al primo ordine di uno sviluppo in serie di Taylor.

Capitolo 3

Modelli

Il filtro di Kalman necessita per il suo funzionamento di un *modello del sistema* oggetto di osservazione da cui ottenere lo stato predetto \bar{x}_k e di un *modello della misura* z_k . Nel caso di interesse si vuole studiare il movimento di un braccio umano osservato mediante sensore Kinect, il quale fornisce la posizione del centro giunti spalla, gomito e polso nello spazio operativo mediante l'extrapolazione dello scheletro dell'uomo. Il filtro di Kalman, invece, agisce sulle variabili di giunto q_i del braccio umano nello spazio giunti. È necessario dunque capire come modellizzare il braccio, quali vincoli introdurre e come eseguire le fasi di cinematica diretta e inversa.

In una prima fase di studio si cerca di indagare quale sia il modello del sistema più opportuno. Non avendo a disposizione dati sperimentali, si effettua una prima valutazione attraverso un modello multi-body del braccio costruito in Simscape. Si utilizza questo modello come modello dello strumento di misura, facendo muovere il braccio, acquisendo le informazioni sui centri giunti e introducendo rumore per simulare il comportamento del sensore. Inoltre tale modello costituisce il sistema osservato e quindi permette di avere i valori “veri” della misura consentendo un confronto.

I modelli del sistema presi in esame sono: un modello dinamico che utilizza un filtro di Kalman esteso ad input ignoti (*Unknown Inputs Extended Kalman Filter*,

disposizione altri ragionevoli da adottare. Si utilizza la notazione $\mathbf{p}_{e,1}^0, \mathbf{p}_{e,2}^0$ per indicare la posizione di gomito e spalla nel sistema di riferimento spalla. Le analisi vengono effettuate sul braccio sinistro.

3.2 Modello multi-body del braccio

Per poter effettuare delle simulazioni senza ricorrere a dati sperimentali, viene utilizzato un modello multi-body del braccio realizzato in ambiente Simscape. Si è deciso di semplificare la struttura del braccio umano utilizzando due cilindri collegati fra loro come illustrato in Figura 3.2.

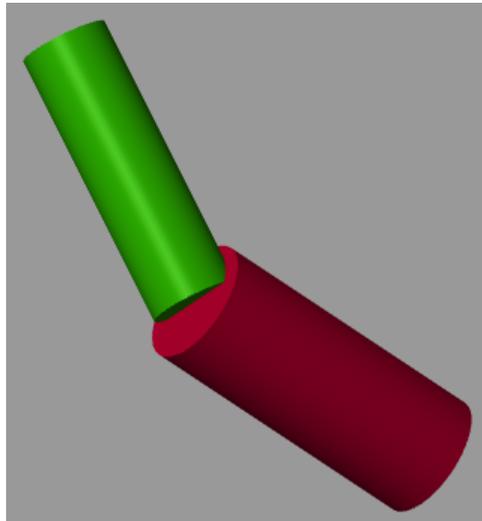


Figura 3.2 Visualizzazione grafica del modello 3D del braccio umano realizzato in Simscape

Al modello si forniscono come input i valori delle variabili di giunto q_i . Il modello fornisce come output le coordinate del gomito e polso nel tempo $\mathbf{p}_{e,1}^0, \mathbf{p}_{e,2}^0$. Successivamente si sporcheranno tali dati di output con del rumore per simulare una realistica situazione di acquisizione dati da parte del sensore e si confronteranno i risultati ottenuti con i valori veri non affetti da rumore.

3.2.1 Set-up

Vengono definite in Matlab le caratteristiche fisico-geometriche del braccio da consegnare al modello Simscape. Vengono indicati: l (lunghezza del braccio/avambraccio), r (raggio), m (massa). Si utilizzano rispettivamente i pedici

1 e 2 per indicare il braccio e l'avanbraccio. È necessario introdurre i momenti di inerzia:

$$J_r = J_{xx} = J_{yy} = \frac{1}{12} m \cdot l^2 \quad (3.1)$$

$$J_{ax} = J_{zz} = \frac{1}{2} m \cdot r^2 \quad (3.2)$$

calcolati rispetto una terna levogira centrata nel baricentro del cilindro, con l'asse z nella direzione dell'asse del cilindro, come riportato in Figura 3.3. J_r risulta dunque il momento di inerzia calcolato rispetto il baricentro di un'asta sottile con massa uniforme. J_{ax} risulta il momento di inerzia di un cilindro pieno rispetto il suo asse.

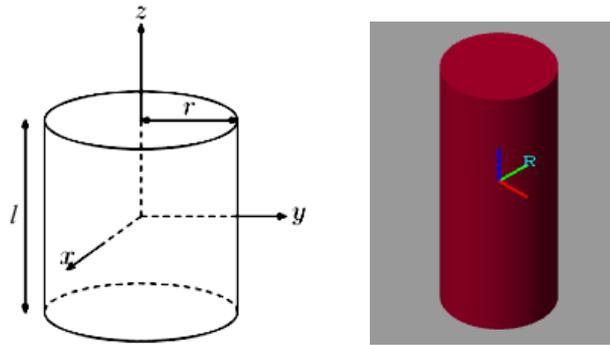


Figura 3.3 Cilindro e sistema di riferimento a esso relativo con centro nel suo baricentro schematizzato e rappresentato in Simscape

I valori relativi al raggio r e massa m dei cilindri sono assunti ragionevolmente pari a $r_1 = 50$ mm, $r_2 = 35$ mm, $m_1 = 2,2$ kg, $m_2 = 1,7$ kg. Le lunghezze del braccio sono poste a $l_1 = 241$ mm, $l_2 = 229$ mm.

Si fornisce dunque al modello *Simscape* il tensore di inerzia rispetto il sistema di riferimento solidale avente origine nel baricentro del solido:

$$\mathbf{J} = \begin{bmatrix} J_{xx} & 0 & 0 \\ 0 & J_{yy} & 0 \\ 0 & 0 & J_{zz} \end{bmatrix} \quad (3.3)$$

Si definisce inoltre la frequenza dell'elaborazione, pari a 30 Hz ovvero la stessa frequenza di acquisizione di Kinect.

3.2.2 Modello in Simscape

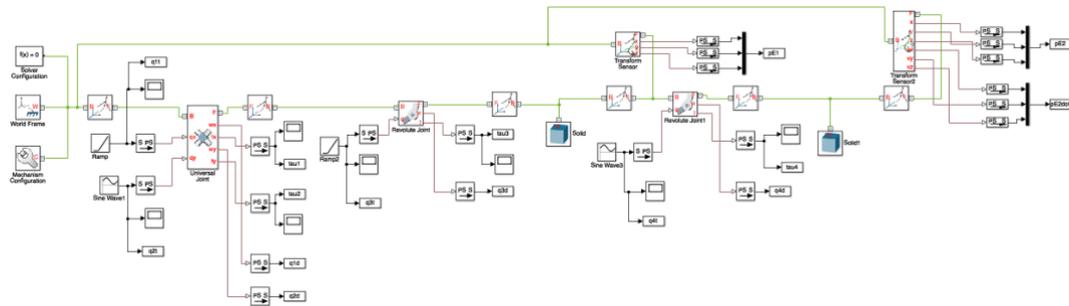


Figura 3.4 Modello Simscape complessivo

Si presenta la costruzione del modello dalla base, rappresentata dalla spalla, fino al polso. In generale di seguito si indicheranno matrici di sola rotazione con \mathbf{R} , vettori di traslazione con \mathbf{T} e matrici omogenee di rototraslazione con \mathbf{A} .

Per prima cosa viene definito come sistema di riferimento “*World*”, il sistema di riferimento spalla O , avente asse z verso l’alto. Per $q_1 = q_2 = q_3 = q_4 = 0$, il braccio si presenta in una configurazione distesa con asse lungo l’asse z del sistema di riferimento O . Viene impostata una gravità uniforme.

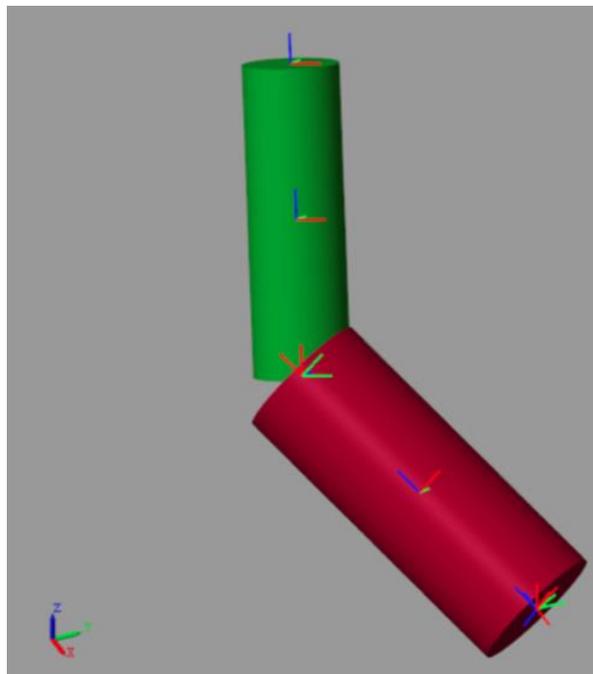


Figura 3.5 Sistemi di riferimento presenti nel modello multi-body, rappresentati in Simscape

Inizialmente, attraverso il primo modulo “*Rigid Transform*”, si definisce la matrice $\mathbf{R}_{O'I}^O$ che consente una rotazione del sistema di riferimento O di -90° rispetto l'asse y :

$$\mathbf{R}_{O'I}^W = \begin{bmatrix} 0 & 0 & -1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \quad (3.4)$$

Si indica con $\mathbf{A}_{O'I}^O$ la corrispondente matrice di trasformazione omogenea avente rotazione $\mathbf{R}_{O'I}^O$ e nessuna traslazione. La notazione è valida per l'intera trattazione.

$$\mathbf{A}_{O'I}^O = \begin{bmatrix} \mathbf{R}_{O'I}^O & \mathbf{0} \\ \mathbf{0} & 1 \end{bmatrix} \quad (3.5)$$

Tale rotazione è necessaria per posizionare l'asse x verso l'alto in modo da poter utilizzare il modulo seguente “*Universal Joint*”. Il sistema di riferimento seguente (*follower frame*) ruota prima intorno all'asse x , dopo intorno all'asse y . Si definiscono le seguenti matrici di rotazione in cui compaiono le variabili di giunto q_1 (rotazione intorno all'asse x) e q_2 (rotazione intorno all'asse y):

$$\mathbf{R}_{O_1'I}^{O_1'}(q_1) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos q_1 & -\sin q_1 \\ 0 & \sin q_1 & \cos q_1 \end{bmatrix} \quad (3.6)$$

$$\mathbf{R}_{O_2'I}^{O_2'}(q_2) = \begin{bmatrix} \cos q_2 & 0 & \sin q_2 \\ 0 & 1 & 0 \\ -\sin q_2 & 0 & \cos q_2 \end{bmatrix} \quad (3.7)$$

A questo punto si effettua una rotazione rigida intorno all'asse y di $+90^\circ$ attraverso la matrice:

$$\mathbf{R}_{O''I}^{O_2'} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{bmatrix} \quad (3.8)$$

A seguito di tale trasformazione si applica il modulo “*Revolute Joint*” il quale consente la rotazione intorno all’asse z mediante la variabile di giunto q_3 . Si presenta la matrice di rotazione risultante:

$$\mathbf{R}_{O_3^{II}}^{O^{II}}(q_3) = \begin{bmatrix} \cos q_3 & -\sin q_3 & 0 \\ \sin q_3 & \cos q_3 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3.9)$$

Si sono disposti i tre gradi di libertà relativi alla spalla. Per collocare il solido rappresentate il braccio, è necessario inserirne il sistema di riferimento baricentrico, con asse z lungo l’asse del cilindro. Per fare questo è necessario traslare il sistema di riferimento O_3^{II} di $l_1/2$ mediante il blocco “*Rigid Transform*” impostando la sola traslazione:

$$\mathbf{T}_{O_3^{III}}^{O_3^{II}} = \begin{bmatrix} 0 \\ 0 \\ l_1/2 \end{bmatrix} \quad (3.10)$$

Si utilizza il blocco “*Solid*” per costruire il modello del braccio. Il braccio viene approssimato come un cilindro pieno avente come tensore di inerzia \mathbf{J}_1 , massa m_1 , lunghezza l_1 e raggio r_1 .

Successivamente viene applicata una rototraslazione tramite “*Revolute Joint*”. Una rotazione di -90° intorno all’asse y e di -90° intorno all’asse x e un’ulteriore traslazione di $l_1/2$ lungo l’asse z. La rototraslazione è esprimibile mediante la seguente matrice di trasformazione omogenea $\mathbf{A}_{O^{IV}}^{O^{III}}$:

$$\mathbf{A}_{O^{IV}}^{O^{III}} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & l_1/2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.11)$$

È possibile passare direttamente dal sistema di riferimento O_3^{II} a O^{IV} mediante la trasformazione:

$$\mathbf{A}_{O^{IV}}^{O^{II}} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & l_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.12)$$

Attraverso il modulo “*Revolute Joint*” si ha la rotazione dell’asse z mediante la variabile di giunto q_4 . La rotazione precedente era necessaria per posizionare l’asse z in modo opportuno per poter poi disporre una rotazione intorno all’asse z, con il presente modulo, al fine di simulare il movimento dell’avambraccio rispetto al braccio. Tale rotazione è espressa da:

$$\mathbf{R}_{O_4^{IV}}^{O_4^{IV}}(q_4) = \begin{bmatrix} \cos q_4 & -\sin q_4 & 0 \\ \sin q_4 & \cos q_4 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3.13)$$

Per collocare il solido rappresentate l’avambraccio, è necessario inserirne il sistema di riferimento baricentrico, con asse z lungo l’asse del cilindro. Per fare questo è necessaria una rototraslazione del sistema di riferimento O_4^{IV} con una traslazione di $l_2/2$ lungo l’asse x, una rotazione di $+90^\circ$ lungo l’asse x e di $+90^\circ$ lungo l’asse y mediante il blocco “*Rigid Transform*”. Si ottiene la matrice omogenea di rototraslazione:

$$\mathbf{A}_{O^V}^{O_4^{IV}} = \begin{bmatrix} 0 & 0 & 1 & l_2/2 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.14)$$

Si utilizza il blocco “*Solid*” per costruire il modello dell’avambraccio, approssimato come un cilindro pieno avente come tensore di inerzia \mathbf{J}_2 , massa m_2 , lunghezza l_2 e raggio r_2 .

Infine, attraverso il blocco “*Rigid Transform*”, si dispone una traslazione di $l_2/2$ lungo l’asse z al fine di individuare il sistema di riferimento O^{VI} di centro polso:

$$\mathbf{T}_{O^{VI}}^{O^V} = \begin{bmatrix} 0 \\ 0 \\ l_2/2 \end{bmatrix} \quad (3.15)$$

3.2.3 Output

Si vogliono acquisire le coordinate del gomito $\mathbf{p}_{e,1}^O$ e del polso $\mathbf{p}_{e,2}^O$ nel sistema di riferimento spalla O . Per farlo si utilizza il modulo “*Trasform Sensor*” in grado di misurare la variazione, dipendente dal tempo, tra due sistemi di riferimento. È possibile specificare quali relazioni di vogliono acquisire mediante le proprietà del modulo (per esempio rotazioni, traslazioni, velocità e accelerazioni). Nel caso specifico sono di interesse le sole coordinate dei punti rappresentanti gomito e polso che vengono acquisite nella variabile $PE1$ per il gomito e $PE2$ per il polso. Esse sono delle matrici ($t \times 3$), con t la lunghezza del vettore tempo ($time$).

Nel caso del polso, si predispone anche l’acquisizione delle velocità del punto lungo gli assi cartesiani (registrandolo nella variabile $PE2dot$) nel caso in cui sia utile una sua valutazione.

Per quanto riguarda il gomito il *Trasform Sensor* viene inserito fra i sistemi di riferimento $O - O^{IV}$, mentre per il polso tra i sistemi di riferimento $O - O^{VI}$. Quello che di fatto fa il modulo è la cinematica diretta del sistema a cui si rimanda al paragrafo 3.3 per ulteriori dettagli.

In aggiunta si acquisiscono i valori “veri” delle coppie ai giunti $\tau_1, \tau_2, \tau_3, \tau_4$ e delle velocità $\dot{q}_1, \dot{q}_2, \dot{q}_3, \dot{q}_4$ in modo da poterli confrontare con i valori stimati dal filtro di Kalman. I dati vengono registrati nelle variabili $tau1, tau2, tau3, tau4$ per le coppie e $q1d, q2d, q3d, q4d$ per le velocità ai giunti.

3.2.4 Input

Utilizzando funzioni rampa e seno, si dispongono gli input ai blocchi giunto, ovvero il valore delle variabili di giunto q nel tempo. I dati vengono registrati nelle variabili $q1t, q2t, q3t, q4t$.

Inizialmente si utilizzano funzioni tentativo per una corretta simulazione dei movimenti del braccio. Tenendo conto dei valori più plausibili per le variabili di giunto \mathbf{q} si utilizza il seguente sistema scelto arbitrariamente:

$$\begin{cases} q_1 = 0.01t + \frac{\pi}{6} \\ q_2 = \frac{\pi}{4} \sin(0.02\pi t) - \frac{\pi}{3} \\ q_3 = 0.01t + \frac{\pi}{6} \\ q_4 = \frac{\pi}{7} \sin(0.08\pi t) + \frac{\pi}{4} \end{cases} \quad (3.16)$$

dove t è il tempo.

3.2.5 Modello del sensore

Ottenute le posizioni del gomito $\mathbf{p}_{e,1}^O$ e del polso $\mathbf{p}_{e,2}^O$ nel sistema di riferimento O attraverso il modello Simscape, si vuole che tali posizioni siano rappresentative dei dati acquisiti realmente dal sensore di profondità. Per farlo è necessario aggiungere del rumore alle posizioni simulate “vere”.

Il rumore viene assimilato all’aggiunta di una distribuzione gaussiana a media nulla con data deviazione standard σ_{sensor} .

A tal fine viene utilizzata la funzione “*randn*” di Matlab, dove $randn(M, N)$ restituisce una matrice ($M \times N$). Tale funzione viene preceduta dalla funzione “*rng*” la quale permette di avere sempre la stessa sequenza di numeri random per ogni esecuzione del codice, rendendo confrontabile il risultato di volta in volta e su più dispositivi.

$$\begin{cases} \mathbf{p}_{e,1noise}^O = \mathbf{p}_{e,1}^O + \sigma_{sensor} \mathbf{M}_{randn(3,t)} \\ \mathbf{p}_{e,2noise}^O = \mathbf{p}_{e,2}^O + \sigma_{sensor} \mathbf{M}_{randn(3,t)} \end{cases} \quad (3.17)$$

dove t è la dimensione del vettore tempo e $\mathbf{M}_{randn(3,t)}$ la matrice generata dalla funzione $randn(3, t)$. Viene impostata una $\sigma_{sensor} = 5 \cdot 10^{-3} m$.

3.3 Cinematica diretta e inversa

Di seguito si illustra la procedura di cinematica diretta e inversa adottata nel modello del braccio. Tale procedura e la definizione delle quattro variabili di giunto, discussa nel paragrafo 3.2, rimangono invariate per tutta la trattazione per ogni modello adottato.

3.3.1 Cinematica diretta

Noti i valori di q_1, q_2, q_3, q_4 , si cercano le coordinate del gomito $\mathbf{p}_{e,1}^O$ e del polso $\mathbf{p}_{e,2}^O$ attraverso la cinematica diretta (*forward kinematics*).

Per quanto precedentemente descritto, il punto $\mathbf{p}_{e,1}^O$, ovvero la posizione del centro gomito, nel sistema di riferimento O è ottenuto come segue:

$$\tilde{\mathbf{p}}_{e,1}^O = \mathbf{A}_{O^I}^O \mathbf{A}_{O_1^I}^{O^I} \mathbf{A}_{O_2^I}^{O_1^I} \mathbf{A}_{O_3^I}^{O_2^I} \mathbf{A}_{O_3^{II}}^{O_3^I} \tilde{\mathbf{p}}_{e,1}^{O_3^{II}} \quad (3.18)$$

dove

$$\tilde{\mathbf{p}}_{e,1}^{O_3^{II}} = \begin{bmatrix} 0 \\ 0 \\ l_1 \\ 1 \end{bmatrix} \quad (3.19)$$

La notazione $\tilde{\mathbf{p}}$ indica la rappresentazione omogenea del vettore \mathbf{p} . Il punto $\mathbf{p}_{e,2}^O$, ovvero la posizione del centro polso, nel sistema di riferimento O è ottenuto come segue:

$$\tilde{\mathbf{p}}_{e,2}^O = \mathbf{A}_{O^I}^O \mathbf{A}_{O_1^I}^{O^I} \mathbf{A}_{O_2^I}^{O_1^I} \mathbf{A}_{O_3^I}^{O_2^I} \mathbf{A}_{O_3^{II}}^{O_3^I} \mathbf{A}_{O_3^{IV}}^{O_3^{II}} \mathbf{A}_{O_4^{IV}}^{O_3^{IV}} \tilde{\mathbf{p}}_{e,2}^{O_4^{IV}} \quad (3.20)$$

dove

$$\tilde{\mathbf{p}}_{e,2}^{O_4^{IV}} = \begin{bmatrix} l_2 \\ 0 \\ 0 \\ 1 \end{bmatrix} \quad (3.21)$$

Per la cinematica diretta si realizza la funzione Matlab “*fkine_humanArm*” che, a partire dalle variabili di giunto e dai valori di lunghezza del braccio e avambraccio, fornisce in output le posizioni $\mathbf{p}_{e,1}^O$ e $\mathbf{p}_{e,2}^O$.

3.3.2 Cinematica inversa

Siano $\mathbf{p}_{e,1}^O = (x_{e,1}, y_{e,1}, z_{e,1})$ e $\mathbf{p}_{e,2}^O = (x_{e,2}, y_{e,2}, z_{e,2})$ le coordinate del centro gomito e del centro polso nel sistema di riferimento O . Note le posizioni del gomito $\mathbf{p}_{e,1}^O$ e del polso $\mathbf{p}_{e,2}^O$ fornite dal modello del sensore è di interesse il calcolo delle variabili di giunto q_1, q_2, q_3, q_4 , possibile mediante cinematica inversa (*inverse kinematics*).

Data la sola posizione del polso (3 coordinate) la soluzione al problema cinematico inverso non sarebbe unica poiché si hanno 4 gradi di libertà (ridondanza). Note invece anche le coordinate del centro gomito la risoluzione del problema è possibile. Valgono i vincoli:

$$\begin{cases} \|\mathbf{p}_{e,1}^W\| = l_1 \\ \|\mathbf{p}_{e,2}^W\| \leq l_1 + l_2 \end{cases} \quad (3.22)$$

Si riporta la soluzione analitica del problema cinematico inverso.

$$\begin{cases} q_1 = \text{atan2}\left(\frac{y_{E1}}{l_1 \sin q_1}, \frac{x_{E1}}{l_1 \sin q_1}\right) \\ q_2 = \text{acos}\left(\frac{z_{E1}}{l_1}\right), \text{ con } -\pi \leq q_2 \leq 0 \\ q_3 = \text{atan2}\left(\frac{y_{E2} - l_1 \sin q_1 \sin q_2 + a_1 - \cos q_4 (l_2 \sin q_1 \sin q_2 + a_2)}{l_1 \cos q_1 \sin q_4}, \frac{l_1 \cos q_2 + l_2 \cos q_2 \cos q_4 - z_{E2}}{l_1 \sin q_1}\right) \\ q_4 = \text{acos}\left(\frac{x_{E1} - l_1 \cos q_1 \sin q_2 + \tan q_1 (y_{E2} - l_1 \sin q_1 \sin q_2 + a_1) + a_1 \cot q_1}{l_2 \cos q_1 \sin q_2 + a_2 \cot q_1 + \tan q_1 (l_2 \sin q_1 \sin q_2 + a_2)}\right), \text{ con } 0 \leq q_4 \leq \pi \\ a_1 = (z_{E1} - l_1 \cos q_2) \sin q_1 \cot q_2; a_2 = l_2 \cot q_2 \sin q_1 \cos q_2 \end{cases} \quad (3.23)$$

Si ricorda che $-\pi \leq \text{atan2}(Y, X) \leq \pi$. I limiti sulle variabili di giunto, compatibilmente con i movimenti del braccio umano, risultano pertanto:

$$\begin{cases} -\pi \leq q_1 \leq \pi \\ -\pi \leq q_2 \leq 0 \\ -\pi \leq q_3 \leq \pi \\ 0 \leq q_4 \leq \pi \end{cases} \quad (3.24)$$

3.4 Modello dinamico del sistema

Si presenta un modello dinamico del sistema precedentemente elaborato. Si eviterà di entrare nei dettagli teorici, illustrando le logiche alla base con i dovuti riferimenti.

Il modello dinamico di sistemi manipolatori open-loop può essere descritto dalla seguente formulazione [18] [19]:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{u} + \mathbf{F}_{ext}(\mathbf{q}) \quad (3.25)$$

dove \mathbf{q} sono le variabili di giunto, $\mathbf{M} \in \mathbb{R}^{n \times n}$ è la matrice di massa, $\mathbf{C} \in \mathbb{R}^n$ è un vettore comprensivo di diversi termini [18], $\mathbf{u} \in \mathbb{R}^n$ è il vettore delle coppie ai giunti, $\mathbf{F}_{ext} \in \mathbb{R}^n$ sono le forze esterne ai giunti, con n i gradi di libertà. Nel caso specifico $n = 4$.

Questa espressione è non lineare. Volendo dunque applicare un filtro di Kalman esteso (EKF) si deve effettuare una linearizzazione del modello proposto. Tramite [19] si linearizza il problema giungendo a:

$$\mathbf{M}\ddot{\mathbf{q}} + \mathbf{A}_D\dot{\mathbf{q}} + \mathbf{B}_D\mathbf{q} = \mathbf{u} \quad (3.26)$$

dove \mathbf{A}_D e \mathbf{B}_D sono le matrici di linearizzazione. Si noti come gli input \mathbf{u} dell'equazione, rappresentanti le coppie ai giunti, non siano noti. Ciò richiede lo sviluppo di un filtro di Kalman esteso a input ignoti (UIEKF). In [20] si presenta una metodologia efficace per la stima degli input ignoti \mathbf{u} che viene presa come riferimento.

Tale filtro di Kalman utilizza come stato del sistema la variabile $\mathbf{x} \in \mathbb{R}^{2n}$ comprensiva delle variabili di giunto $\mathbf{q} \in \mathbb{R}^n$ e la loro derivata $\dot{\mathbf{q}} \in \mathbb{R}^n$, ovvero:

$$\mathbf{x} = \begin{bmatrix} \mathbf{q} \\ \dot{\mathbf{q}} \end{bmatrix} \quad (3.27)$$

Le informazioni sulle accelerazioni $\ddot{\mathbf{q}} \in \mathbb{R}^n$ vengono poi ottenute, istante per istante, mediante la risoluzione dell'Eq. (3.25).

Per il suo corretto funzionamento l'UIEKF richiede che la misura sia $\mathbf{z} \in \mathbb{R}^{2n}$, ovvero della stessa dimensione dello stato \mathbf{x} e che contenga quindi sia le informazioni sulle variabili di giunto \mathbf{q} che sulle loro velocità $\dot{\mathbf{q}}$. Come precedentemente messo in luce, il sensore Kinect fornisce solo la posizione dei giunti nello spazio operativo che, attraverso cinematica inversa, forniscono informazioni sui valori delle variabili di giunto \mathbf{q} . Vi è dunque l'ulteriore problematica di dover stimare le velocità $\dot{\mathbf{q}}$ misurate attraverso un modello del sensore.

Si decide di effettuare questa operazione adottando un ulteriore filtro di Kalman con un modello cinematico ad accelerazione costante (si rimanda al paragrafo 3.5 per approfondimenti sul modello). In questo caso lo stato del sistema è espresso dal vettore $\mathbf{q}_{dot} \in \mathbb{R}^{3n}$ comprensivo delle variabili di giunto \mathbf{q} e le loro derivate $\dot{\mathbf{q}}$ e $\ddot{\mathbf{q}}$, ovvero:

$$\mathbf{q}_{dot} = \begin{bmatrix} \mathbf{q} \\ \dot{\mathbf{q}} \\ \ddot{\mathbf{q}} \end{bmatrix} \quad (3.28)$$

La misura è rappresentata da $\mathbf{q} \in \mathbb{R}^n$, quindi di diversa dimensione rispetto lo stato \mathbf{q}_{dot} .

Riassumendo, la descrizione del sistema mediante un modello dinamico non lineare richiede l'utilizzo di un UIEKF per la stima dello stato del sistema. Tale osservatore si compone delle fasi di linearizzazione, predizione e aggiornamento. Quest'ultima fase necessita dei valori misurati di \mathbf{q} e $\dot{\mathbf{q}}$. Le velocità $\dot{\mathbf{q}}$ non sono però fornite dal sensore, perciò è necessario stimarle con un modello lineare mediante un filtro di Kalman semplice a sua volta composto dalle fasi di predizione e aggiornamento.

Di seguito si illustra il funzionamento dell'algoritmo, realizzato in ambiente Matlab, facendo ricorso a una rappresentazione a blocchi.

3.4.1 Algoritmo UIEKF

Dallo stato aggiornato \mathbf{x}_k^+ all'istante k , è possibile calcolare gli elementi dell'Eq. (3.25). Conoscendo gli input \mathbf{u}_k stimati, è possibile conoscere le accelerazioni $\ddot{\mathbf{q}}_k$. A questo punto è possibile linearizzare il problema e arrivare all'Eq. (3.26) mediante il calcolo di \mathbf{A}_D e \mathbf{B}_D . La procedura è illustrata in Figura 3.6. La notazione $\mathbf{x}_k^+[\mathbf{q}]$ è da intendersi: “ \mathbf{q} presenti in \mathbf{x}_k^+ ”.

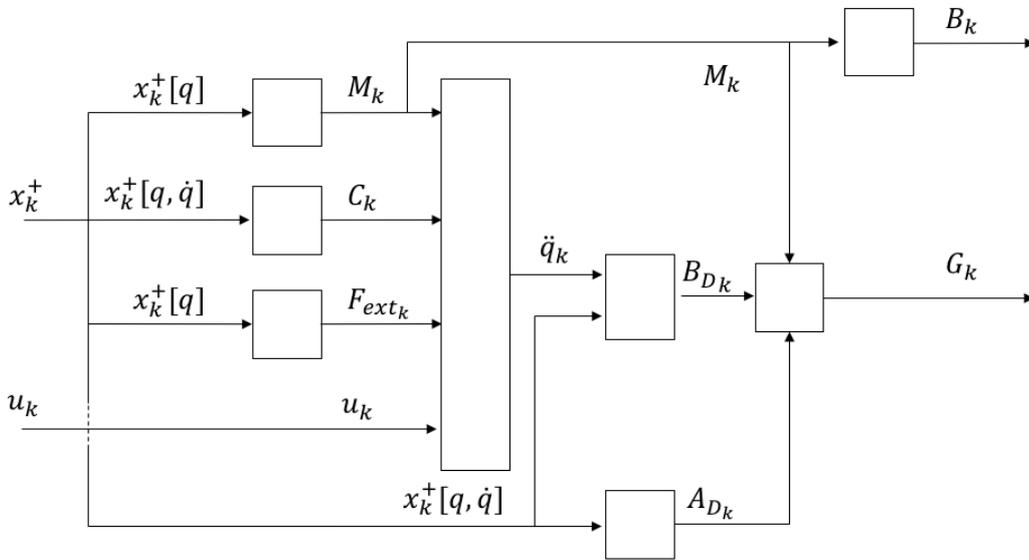


Figura 3.6 Schema a blocchi della fase di linearizzazione di UIEKF e calcolo delle matrici \mathbf{G} e \mathbf{B}

A questo punto è possibile il calcolo delle matrici \mathbf{G}_k e \mathbf{B}_k necessarie per la fase di predizione:

$$\mathbf{x}_{k+1}^- = \mathbf{G}_k \cdot \mathbf{x}_k^+ + \mathbf{B}_k \cdot \mathbf{u}_k \quad (3.29)$$

$$\mathbf{P}_{k+1}^- = \mathbf{G}_k \mathbf{P}_k^+ \mathbf{G}_k^T + \mathbf{Q}_k \quad (3.30)$$

L'Eq. (3.29) rappresenta l'estensione dell'Eq. (2.20) nel caso di presenza di input. È possibile ottenere la predizione dello stato anche attraverso la risoluzione differenziale dell'Eq. (3.25). \mathbf{G}_k e \mathbf{P}_{k+1}^- sono matrici (8×8) , ovvero $(n_x \times n_x)$, mentre \mathbf{B}_k è una matrice (8×4) , ovvero $(n_x \times n_u)$. La fase di predizione è mostrata in Figura 3.7. La covarianza dell'errore di processo \mathbf{Q}_k viene legata alla matrice \mathbf{G}_k attraverso un fattore scelto arbitrariamente.

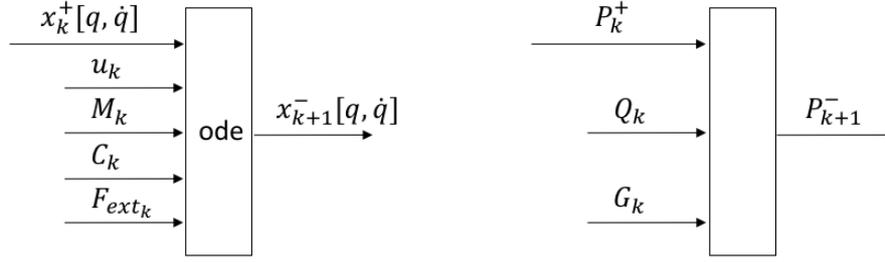


Figura 3.7 Schema a blocchi della fase di predizione di UIKF

Per la fase di aggiornamento è necessario calcolare il guadagno di Kalman \mathbf{K}_{k+1} all'istante $k + 1$, confrontando la matrice di covarianza dello stato predetto \mathbf{P}_{k+1}^- e la matrice di covarianza della misura \mathbf{R}_{k+1} . La matrice del guadagno di Kalman \mathbf{K}_{k+1} ha dimensione (8×8) , ovvero $(n_x \times n_z)$. La matrice di covarianza della misura \mathbf{R}_{k+1} viene assunta arbitrariamente ed è presa invariante nel tempo, per cui verrà indicata semplicemente con \mathbf{R} . Essa ha dimensione (8×8) , ovvero $(n_z \times n_z)$. La matrice del modello di misura \mathbf{H} è una matrice identica tempo invariante di dimensione (8×8) , ovvero $(n_z \times n_x)$. Mediante l'Eq. (2.16) si ha:

$$\mathbf{K}_{k+1} = \mathbf{P}_{k+1}^- \mathbf{H}^T (\mathbf{H} \mathbf{P}_{k+1}^- \mathbf{H}^T + \mathbf{R})^{-1} \quad (3.31)$$

Per poter ottenere la stima dell'input \mathbf{u}_{k+1} , è necessario il calcolo della matrice di covarianza dell'input \mathbf{S}_{k+1} [20]:

$$\mathbf{S}_{k+1} = [\mathbf{B}_k^T \mathbf{H}^T \mathbf{R}^{-1} (\mathbf{I} - \mathbf{H} \mathbf{K}_{k+1}) \mathbf{H} \mathbf{B}_k]^{-1} \quad (3.32)$$

dove \mathbf{I} è la matrice identica. Le Eq. (3.31) e (3.32) sono rappresentate schematicamente in Figura 3.8.

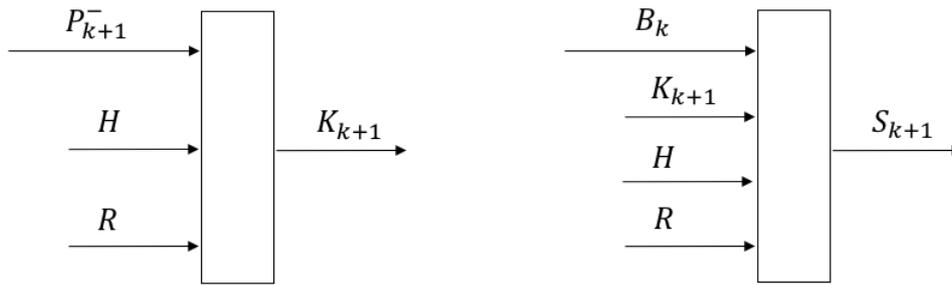


Figura 3.8 Schema a blocchi del calcolo delle matrici K e S in UIEKF

La fase di aggiornamento necessita del confronto dello stato predetto dal modello \mathbf{x}_{k+1}^- con la misura \mathbf{z}_{k+1} . Si è detto che le $\dot{\mathbf{q}}$ vengono stimate mediante un filtro di Kalman con un modello cinematico. In Figura 3.9 è rappresentato lo schema a blocchi che illustra questo processo.

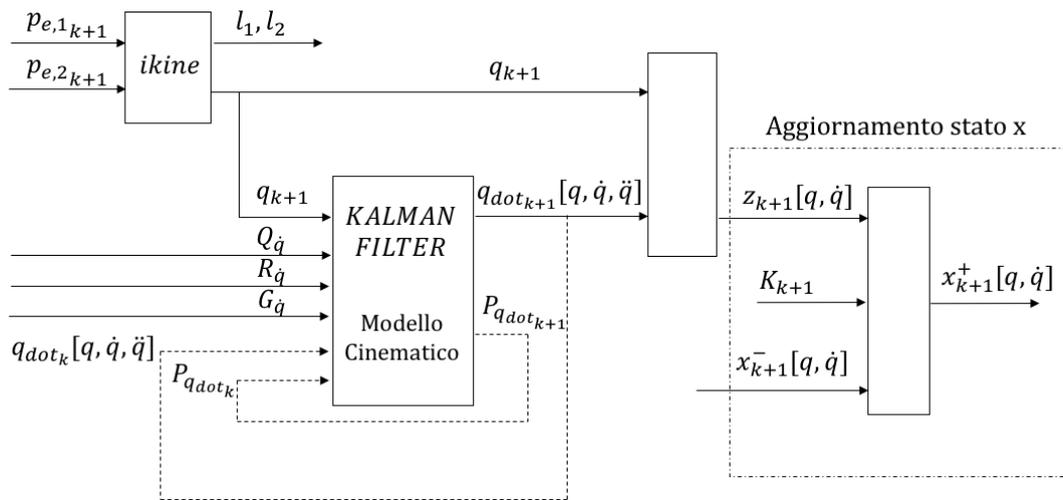


Figura 3.9 Schema a blocchi della fase di aggiornamento dello stato del sistema di UIEKF e stima delle $\dot{\mathbf{q}}$ misurate mediante un KF con modello cinematico a partire dalle posizioni misurate

Dalle posizioni di gomito $\mathbf{p}_{e,1,k+1}$ e polso $\mathbf{p}_{e,2,k+1}$ misurate nello spazio operativo, tramite cinematica inversa (*ikine*) si ottiene il vettore \mathbf{q}_{k+1} contenente le variabili di giunto all'istante $k + 1$. Tale misura viene utilizzata da un filtro di Kalman con modello cinematico per stimare lo stato $\mathbf{q}_{dot,k+1}$. Da questo stato stimato si prendono le $\dot{\mathbf{q}}$ stimate, andando a costituire il vettore \mathbf{z}_{k+1} che rappresenta per l'UIEKF la misura dello stato del sistema. Tale misura viene confrontata con lo stato stimato \mathbf{x}_{k+1}^- e, attraverso il guadagno di Kalman \mathbf{K}_{k+1} si

ottiene lo stato aggiornato \mathbf{x}_{k+1}^+ . Riprendendo l'Eq. (2.17), lo stato aggiornato è ottenuto come:

$$\mathbf{x}_{k+1}^+ = \mathbf{x}_{k+1}^- + \mathbf{K}_{k+1}(\mathbf{z}_{k+1} - \mathbf{H}\mathbf{x}_{k+1}^-) \quad (3.33)$$

La fase di aggiornamento viene completata con la stima della covarianza dello stato del sistema \mathbf{P}_{k+1}^+ e con la stima dell'input ignoto \mathbf{u}_{k+1} [20]:

$$\mathbf{P}_{k+1}^+ = (\mathbf{I} - \mathbf{K}_{k+1}\mathbf{H})[\mathbf{P}_{k+1}^- + \mathbf{B}_k\mathbf{S}_{k+1}\mathbf{B}_k^T(\mathbf{I} - \mathbf{K}_{k+1}\mathbf{H})^T] \quad (3.34)$$

$$\mathbf{u}_{k+1} = \mathbf{S}_{k+1}\mathbf{B}_k^T\mathbf{H}^T\mathbf{R}^{-1}(\mathbf{I} - \mathbf{H}\mathbf{K}_{k+1})[\mathbf{z}_{k+1} - \mathbf{H}\mathbf{x}_{k+1}^+ + \mathbf{H}\mathbf{B}_k\mathbf{u}_k] \quad (3.35)$$

In Figura 3.10 è presentato lo schema a blocchi relativo alle Eq. (3.34) e (3.35):

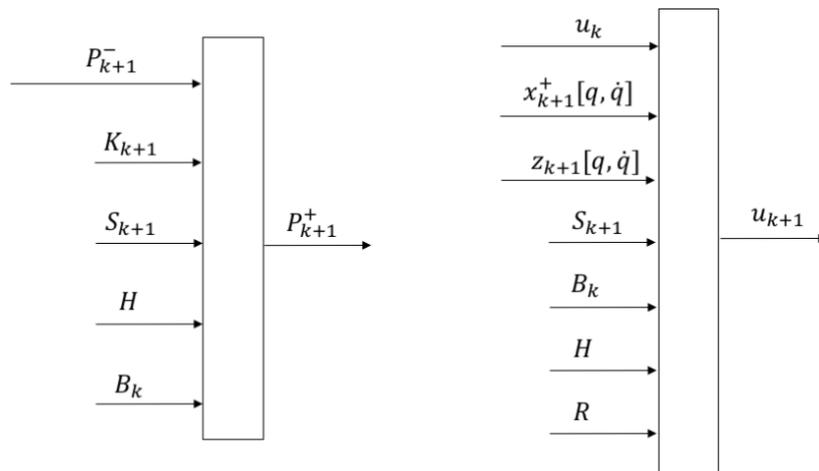


Figura 3.10 Schema a blocchi della covarianza dello stato \mathbf{P} aggiornata e dell'input \mathbf{u}

In Figura 3.11 si scompone il blocco del filtro di Kalman mostrato in Figura 3.9. Non si entra nel dettaglio ad analizzare tutti i passaggi in quanto il suo funzionamento è molto simile al modello che verrà descritto successivamente nel paragrafo 3.5.

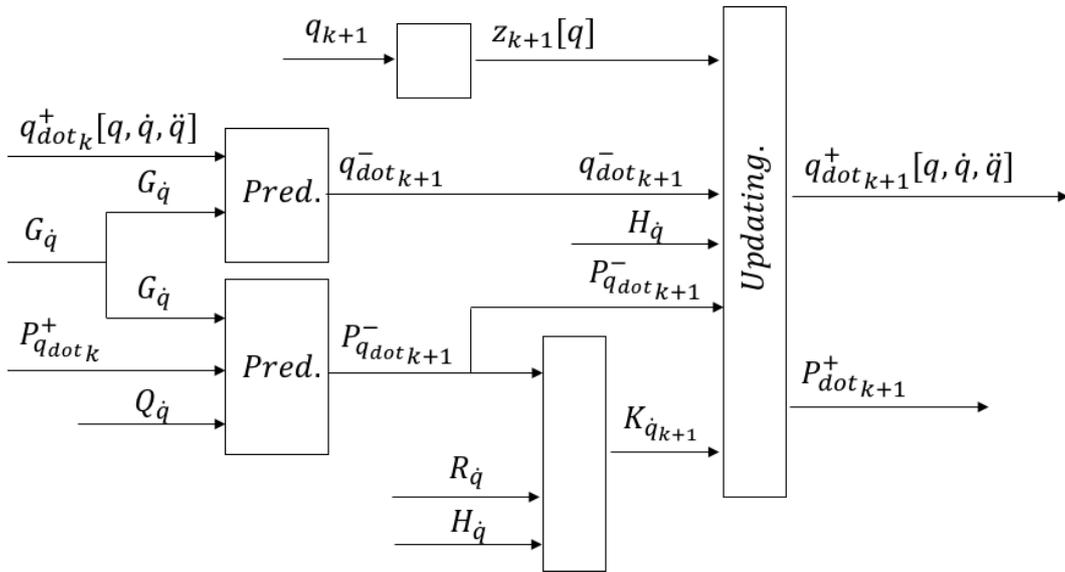


Figura 3.11 Schema a blocchi del filtro di Kalman utilizzato per la stima delle \dot{q} misurate

3.5 Modello cinematico del sistema

Il modello dinamico precedentemente descritto è un modello complesso. Esso ha lo scopo di descrivere al meglio il sistema, ma presenta alcune assunzioni come la stima degli input ignoti e la stima delle velocità misurate attraverso un ulteriore modello, questa volta cinematico. Si è ritenuto di dover confrontare i risultati del modello dinamico con un modello più semplice per valutarne costi e benefici.

Si decide di utilizzare direttamente un modello cinematico ad accelerazione costante per la descrizione del sistema. Tale modello è definito dalle equazioni:

$$\begin{cases} \ddot{\mathbf{q}}_{k+1} = \ddot{\mathbf{q}}_k \\ \dot{\mathbf{q}}_{k+1} = \dot{\mathbf{q}}_k + \ddot{\mathbf{q}}_k \Delta t \\ \mathbf{q}_{k+1} = \mathbf{q}_k + \dot{\mathbf{q}}_k \Delta t + \ddot{\mathbf{q}}_k \Delta t^2 / 2 \end{cases} \quad (3.36)$$

in cui Δt è l'intervallo temporale tra due istanti.

Poiché il modello è lineare, è possibile utilizzare un filtro di Kalman semplice. Esso utilizza come stato del sistema la variabile $\mathbf{x} \in \mathbb{R}^{3n}$ comprensiva delle variabili di giunto $\mathbf{q} \in \mathbb{R}^n$ e delle loro derivate $\dot{\mathbf{q}} \in \mathbb{R}^n$ e $\ddot{\mathbf{q}} \in \mathbb{R}^n$, ovvero:

$$\mathbf{x} = \begin{bmatrix} \mathbf{q} \\ \dot{\mathbf{q}} \\ \ddot{\mathbf{q}} \end{bmatrix} \quad (3.37)$$

La misura è rappresentata da $\mathbf{z} \in \mathbb{R}^n$, contenente le variabili di giunto \mathbf{q} misurate, che derivano dalla cinematica inversa. La misura ha quindi diversa dimensione rispetto allo stato.

Date le Eq.(3.36) è possibile ottenere la matrice \mathbf{G} che rappresenta il modello:

$$\mathbf{G} = \begin{bmatrix} \mathbf{I} & \mathbf{I} \cdot \Delta t & \mathbf{I} \cdot \Delta t^2 / 2 \\ \mathbf{0} & \mathbf{I} & \mathbf{I} \cdot \Delta t \\ \mathbf{0} & \mathbf{0} & \mathbf{I} \end{bmatrix} \quad (3.38)$$

con $\mathbf{I} \in \mathbb{R}^{n \times n}$ la matrice identica. La matrice $\mathbf{G} \in \mathbb{R}^{3n \times 3n}$ è costante se Δt è costante.

3.5.1 Algoritmo KF

In Figura 3.12 si mostra lo schema a blocchi dell'algoritmo implementato.

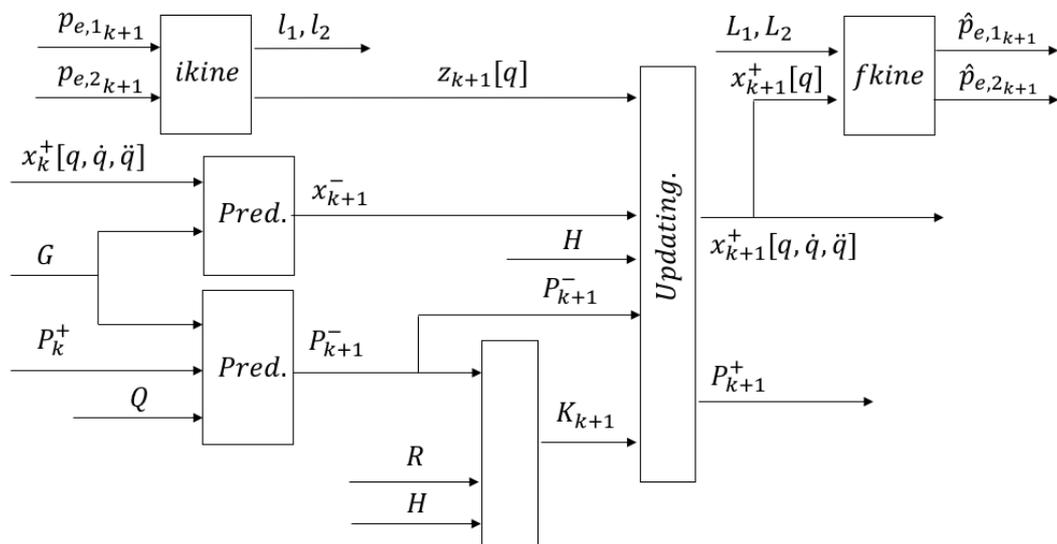


Figura 3.12 Schema a blocchi del filtro di Kalman con modello cinematico

La misura viene effettuata sulle posizioni dei giunti nello spazio operativo. Attraverso cinematica inversa si ottengono le variabili di giunto \mathbf{q} .

Mediante la matrice \mathbf{G} e la matrice di covarianza di processo \mathbf{Q} , è possibile la fase di *predizione* del filtro di Kalman. Ricordando le Eq. (2.20) e (2.21) si ha:

$$\mathbf{x}_{k+1}^- = \mathbf{G} \cdot \mathbf{x}_k^+ \quad (3.39)$$

$$\mathbf{P}_{k+1}^- = \mathbf{G}\mathbf{P}_k^+\mathbf{G}^T + \mathbf{Q} \quad (3.40)$$

Le matrici \mathbf{G} , \mathbf{P}_k^+ , \mathbf{P}_{k+1}^- e \mathbf{Q} hanno dimensione (12×12) , ovvero $(n_x \times n_x)$. La matrice di covarianza di processo $\mathbf{Q} \in \mathbb{R}^{3n \times 3n}$ non varia nel tempo e viene assunta nella seguente forma [21]:

$$\mathbf{Q} = \begin{bmatrix} \Delta t^5/20 & \Delta t^4/8 & \Delta t^3/6 \\ \Delta t^4/8 & \Delta t^3/3 & \Delta t^2/2 \\ \Delta t^3/6 & \Delta t^2/2 & \Delta t \end{bmatrix} \sigma_q^2 \quad (3.41)$$

dove σ_q^2 è definita varianza del processo, assunta arbitrariamente.

Per il calcolo del guadagno di Kalman \mathbf{K}_{k+1} sono necessarie le matrici \mathbf{H} e \mathbf{R} . La matrice di covarianza della misura, di dimensione (4×4) , ovvero $(n_z \times n_z)$, risulta:

$$\mathbf{R} = \sigma_r^2 \mathbf{I} \quad (3.42)$$

dove σ_r^2 è la varianza della misura. Si suppone che la varianza sulle quattro variabili di giunto sia dunque la stessa. Il valore di σ_r^2 non è però noto e viene assunto arbitrariamente. La matrice \mathbf{R} è assunta costante nel tempo.

La matrice \mathbf{H} , che lega lo stato alla misura, di dimensione (4×12) , ovvero $(n_z \times n_x)$, risulta:

$$\mathbf{H} = [\mathbf{I} \quad \mathbf{0} \quad \mathbf{0}] \quad (3.43)$$

dove \mathbf{I} è una matrice identità e $\mathbf{0}$ sono matrici nulle di dimensione (4×4) , ovvero $(n_z \times n_z)$. È presente una matrice identità in quanto le quattro componenti

della misura \mathbf{z} , che rappresentano le variabili di giunto \mathbf{q} , corrispondono esattamente alle prime quattro componenti dello stato \mathbf{x} .

È possibile calcolare guadagno di Kalman \mathbf{K}_{k+1} con riferimento all'Eq. (2.16):

$$\mathbf{K}_{k+1} = \mathbf{P}_{k+1}^- \mathbf{H}^T (\mathbf{H} \mathbf{P}_{k+1}^- \mathbf{H}^T + \mathbf{R})^{-1} \quad (3.44)$$

La matrice \mathbf{K}_{k+1} ha dimensione (12×4) , ovvero $(n_x \times n_z)$.

Dalle posizioni di gomito $\mathbf{p}_{e,1k+1}$ e polso $\mathbf{p}_{e,2k+1}$ misurate nello spazio operativo, tramite cinematica inversa, si ottiene il vettore \mathbf{z}_{k+1} contenete le variabili di giunto \mathbf{q}_{k+1} misurate. La misura viene confrontata con lo stato predetto attraverso la fase di *aggiornamento*. Riprendendo le Eq. (2.17) e (2.18) si ha:

$$\mathbf{x}_{k+1}^+ = \mathbf{x}_{k+1}^- + \mathbf{K}_{k+1} (\mathbf{z}_{k+1} - \mathbf{H} \mathbf{x}_{k+1}^-) \quad (3.45)$$

$$\mathbf{P}_{k+1}^+ = \mathbf{P}_{k+1}^- + \mathbf{K}_{k+1} \mathbf{H} \mathbf{P}_{k+1}^- \quad (3.46)$$

Si ottiene così la stima dello stato \mathbf{x}_{k+1}^+ . Si estrapolano dunque dal vettore \mathbf{x}_{k+1}^+ la stima delle variabili di giunto $\hat{\mathbf{q}}$. Attraverso cinematica diretta vengono dunque ottenute le posizioni stimate di gomito $\hat{\mathbf{p}}_{e,1k+1}$ e polso $\hat{\mathbf{p}}_{e,2k+1}$ nello spazio operativo. Durante la cinematica inversa si registrano i valori delle lunghezze misurate braccio l_1 e avambraccio l_2 istante per istante. Per la cinematica diretta vengono utilizzati come valori di lunghezza del braccio e avambraccio L_1 e L_2 , in generale diversi. Come si vedrà, tale procedura costituirà un'ulteriore azione filtrante.

Capitolo 4

Sperimentazione e sviluppo degli algoritmi

Attraverso il modello multi-body realizzato in Simscape è possibile effettuare delle simulazioni che permettano di valutare l'efficacia degli algoritmi elaborati. Tali simulazioni ricevono come input dei valori delle variabili di giunto \mathbf{q} nel tempo che seguono le leggi sinusoidali e lineari presentate dalle Eq. (3.16). Inoltre, utilizzano come valori misurati i dati che fuoriescono da un modello del sensore. In tale modello il rumore del sensore Kinect è simulato dall'introduzione di rumore sulle posizioni reali dei giunti. Nella realtà i valori delle variabili di giunto potranno essere ben più variabili e difficilmente descrivibili mediante un modello così semplice, in aggiunta il rumore di Kinect è più nervoso e imprevedibile rispetto al rumore con distribuzione gaussiana utilizzato nel modello del sensore.

Si utilizza dunque il modello multi-body per un'analisi preliminare di confronto fra il modello cinematico e dinamico e per indagare la risposta dinamica del sistema. Per valutare l'effettiva efficacia degli algoritmi sviluppati sarà necessario svolgere delle prove sperimentali e utilizzare i dati relativi alle posizioni dei giunti effettivamente forniti dal sensore Kinect a seguito di particolari movimenti eseguiti.

Nel seguente capitolo si illustra la procedura di esecuzione delle prove, le considerazioni che derivano dalla loro analisi e gli sviluppi e le scelte perpetrate sulla base dei risultati ottenuti.

4.1 Set-up sperimentale

In generale è possibile acquisire i dati mediante due approcci: utilizzando una sola Kinect e utilizzando due Kinect in contemporanea. Si predispone un set-up sperimentale che sia in grado di lavorare con entrambi gli approcci. Si collocano due Kinect v2 su cavalletti e tre marker realizzati mediante produzione additiva. I marker sono utilizzati per l'individuazione del sistema di un sistema di riferimento fisso World (W). L'asse z è diretto verso l'alto. In Figura 4.1 viene mostrato il set-up sperimentale. Viene rappresentato anche il sistema di riferimento K di ogni Kinect. (chiamate rispettivamente KA e KB). Il sistema di riferimento K è quello che Kinect usa per rappresentare la posizione dei giunti dello scheletro nello spazio.

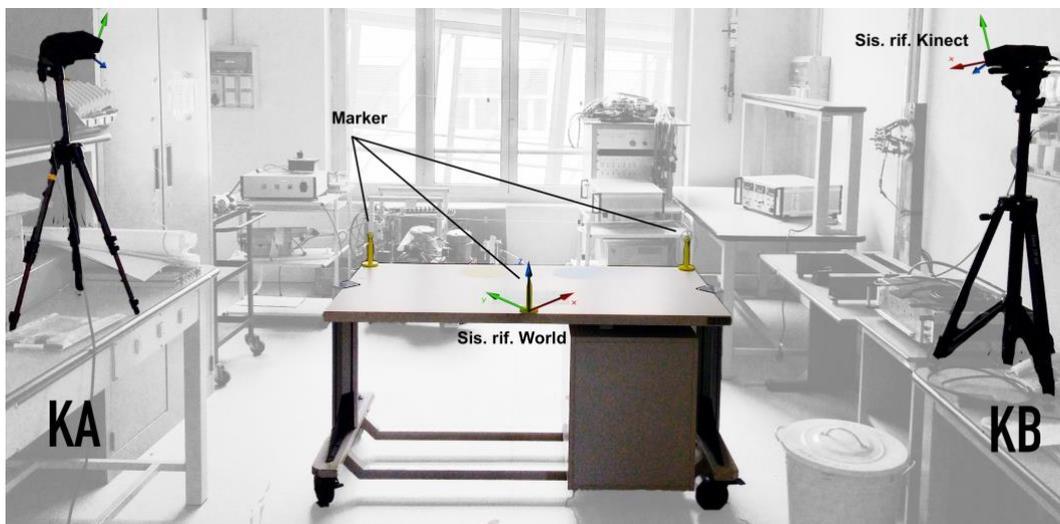


Figura 4.1 Set-up sperimentale

Il filtro di Kalman, come detto, agisce a livello delle variabili di giunto \mathbf{q} . Prima di effettuare l'operazione di cinematica inversa si devono riportare le coordinate delle posizioni dei giunti gomito $\mathbf{p}_{e,1}$ e polso $\mathbf{p}_{e,2}$ nel sistema di riferimento spalla O . In Figura 4.2 si mostra il sistema di riferimento W , il sistema di riferimento O ,

relativo alla spalla sinistra, avente centro nel punto di centro spalla e lo scheletro superiore rilevato da Kinect.

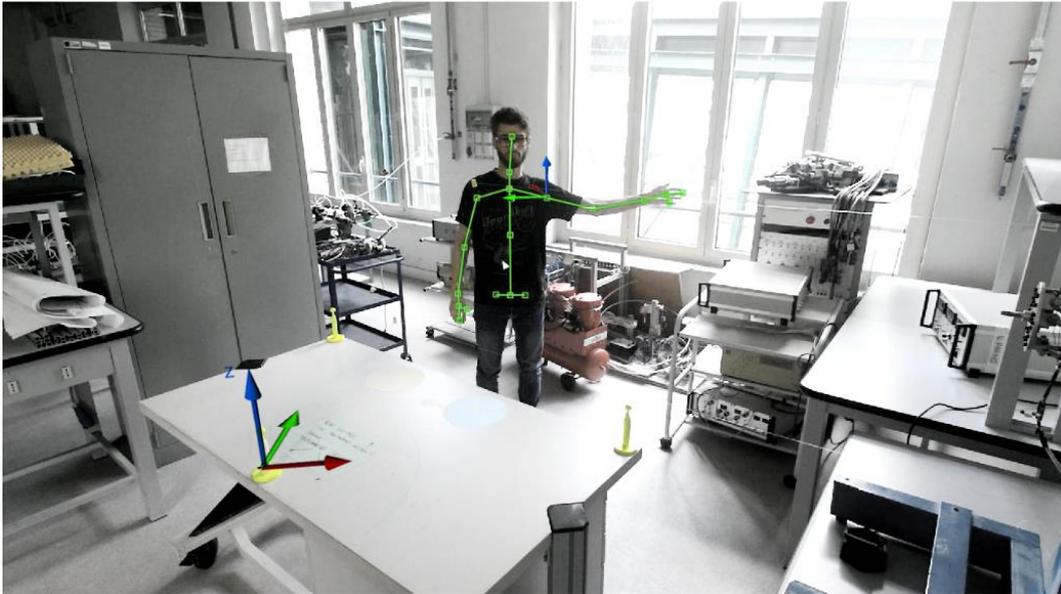


Figura 4.2 Set-up sperimentale, sistema di riferimento World e spalla, scheletro

4.1.1 Determinazione sistema di riferimento World

Per determinare la terna levogira del sistema di riferimento W , si prendono a riferimento le posizioni di 3 marker posizionati su una scrivania come mostrato in Figura 4.3. I marker vengono posizionati in modo tale che uno identifichi il centro del sistema di riferimento (marker A), mentre gli altri individuino la direzione dell'asse x e dell'asse y (rispettivamente marker B e C). I marker hanno forma di birilli con una testa sferica in modo da poter facilmente identificare, mediante la nuvola di punti, le coordinate del punto superiore di ogni sfera.

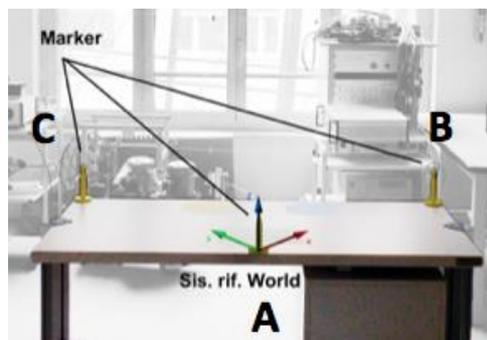


Figura 4.3 Marker per calcolo della terna levogira del sistema di riferimento World

La Kinect utilizza un ulteriore sistema di riferimento “Point Cloud” (Kpc) nel quale esprime le posizioni di tutti i punti della nuvola di punti. Tale sistema di riferimento è ruotato rispetto il sistema di riferimento K dello scheletro della matrice:

$$\mathbf{R}_{Kpc}^K = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (4.1)$$

Dalla nuvola di punti si individuano dunque le coordinate del punto superiore della testa dei marker $\mathbf{p}_A^{Kpc}, \mathbf{p}_B^{Kpc}, \mathbf{p}_C^{Kpc}$. Si riportano tali punti nel sistema di riferimento K ottenendo $\mathbf{p}_A^K, \mathbf{p}_B^K, \mathbf{p}_C^K$ mediante la matrice \mathbf{R}_{Kpc}^K :

$$\mathbf{p}^K = \mathbf{R}_{Kpc}^K \mathbf{p}^{Kpc} \quad (4.2)$$

A partire dai punti $\mathbf{p}_A^K, \mathbf{p}_B^K, \mathbf{p}_C^K$ si vuole costruire una terna levogira. I punti individuati non possono essere direttamente utilizzati per individuare una terna levogira in quanto la retta individuata dai punti A-B non è ortogonale rispetto quella individuata dai punti A-C. Si utilizza una procedura in grado di ottenere una terna levogira ortonormale utilizzando informazioni non esatte. Per prima cosa si individuano dei vettori ausiliari:

$$\begin{cases} \mathbf{v}_A = \mathbf{p}_B^K - \mathbf{p}_A^K \\ \mathbf{v}_B = \mathbf{p}_C^K - \mathbf{p}_A^K \\ \mathbf{v}_C = \mathbf{v}_A \times \mathbf{v}_B \end{cases} \quad (4.3)$$

da cui è possibile calcolare i versori della terna levogira del sistema di riferimento W nel sistema di riferimento K , mediante un ulteriore vettore ausiliario:

$$\hat{\mathbf{z}} = \frac{\mathbf{v}_C}{\|\mathbf{v}_C\|} \quad (4.4)$$

$$\hat{\mathbf{x}} = \frac{\mathbf{v}_A}{\|\mathbf{v}_A\|} \quad (4.5)$$

$$\mathbf{v}_D = \hat{\mathbf{z}} \times \hat{\mathbf{x}} \quad (4.6)$$

$$\hat{\mathbf{y}} = \frac{\mathbf{v}_D}{\|\mathbf{v}_D\|} \quad (4.7)$$

È possibile infine ottenere la matrice omogenea di rototraslazione \mathbf{A}_W^K del sistema di riferimento World rispetto a quello Kinect:

$$\mathbf{A}_W^K = \begin{bmatrix} \hat{\mathbf{x}} & \hat{\mathbf{y}} & \hat{\mathbf{z}} & \mathbf{p}_A^K \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.8)$$

4.1.2 Elaborazione dei dati Kinect

La determinazione della matrice di rototraslazione \mathbf{A}_W^K fissa, permette di calcolare le coordinate dei punti rappresentanti i giunti dello scheletro, nel sistema di riferimento W . Esso è un sistema di riferimento fisso. Si ha, per ogni istante k :

$$\tilde{\mathbf{p}}_{e,i_k}^W = \mathbf{A}_K^W \tilde{\mathbf{p}}_{e,i_k}^K \quad (4.9)$$

dove $\mathbf{A}_K^W = (\mathbf{A}_W^K)^{-1}$, con $i = 0, 1, 2$.

Per poter effettuare l'inversione cinematica per il calcolo delle variabili di giunto \mathbf{q} , è necessario ottenere per ogni istante k le posizioni di giunto gomito $\mathbf{p}_{e,1}^O$ e polso $\mathbf{p}_{e,2}^O$ nel sistema di riferimento O . Instante per istante si deve predisporre una traslazione che permetta al sistema di riferimento O di avere come origine la posizione del giunto spalla, ovvero $\mathbf{p}_{e,0}^O = \mathbf{0}$. Inoltre, si deve disporre una rotazione del sistema di riferimento in modo tale da ottenere l'orientazione della terna del sistema di riferimento O voluta specificata nel paragrafo 3.1.

In prima analisi risulta efficace fissare la rotazione del sistema di riferimento O . Durante l'esecuzione dei movimenti nelle prove è quindi necessario che l'operatore non cambi orientamento del busto per preservare l'orientamento del sistema di riferimento. Successivamente si predisporrà una rototraslazione real-time, facendo ricorso a informazioni di altri punti dello scheletro (paragrafo 4.8).

È possibile dunque ricavare la seguente matrice di rototraslazione, diversa per ogni istante k

$$\mathbf{A}_{O_k}^W = \begin{bmatrix} \mathbf{R}_O^W & \mathbf{p}_{e,0k}^W \\ 0 & 1 \end{bmatrix} \quad (4.10)$$

dove \mathbf{R}_O^W è la matrice di rotazione, presa inizialmente fissa. Per esempio, in Figura 4.2 il soggetto è diritto di fronte alla Kinect KB. Il sistema di riferimento O è ruotato di $+90^\circ$ intorno all'asse z rispetto il sistema di riferimento W e risulterà:

$$\mathbf{R}_O^W = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (4.11)$$

Per ogni istante k si ottiene quindi:

$$\tilde{\mathbf{p}}_{e,i_k}^O = \mathbf{A}_{W_k}^O \tilde{\mathbf{p}}_{e,i_k}^W \quad (4.12)$$

dove $\mathbf{A}_{W_k}^O = (\mathbf{A}_{O_k}^W)^{-1}$, con $i = 1, 2$.

Risulta possibile effettuare l'inversione cinematica, istante per istante, per il calcolo del vettore delle variabili di giunto \mathbf{q}_k che costituiscono la misura dello stato \mathbf{z}_k . A seguito del filtro di Kalman si ottengono la miglior stima $\hat{\mathbf{q}}_k$ delle variabili di giunto che, attraverso cinematica diretta, permettono di ottenere la stima delle posizioni di gomito $\hat{\mathbf{p}}_{e,1k}^O$ e polso $\hat{\mathbf{p}}_{e,2k}^O$ nel sistema di riferimento O . Successivamente, attraverso la matrice $\mathbf{A}_{O_k}^W$ si ottengono le posizioni di gomito $\hat{\mathbf{p}}_{e,1k}^W$ e polso $\hat{\mathbf{p}}_{e,2k}^W$ nel sistema di riferimento W :

$$\tilde{\mathbf{p}}_{e,i_k}^W = \mathbf{A}_{O_k}^W \tilde{\mathbf{p}}_{e,i_k}^O \quad (4.13)$$

In Figura 4.4 uno schema a blocchi riassuntivo dei vari passaggi eseguiti nell'elaborazione dei dati forniti dal sensore Kinect.

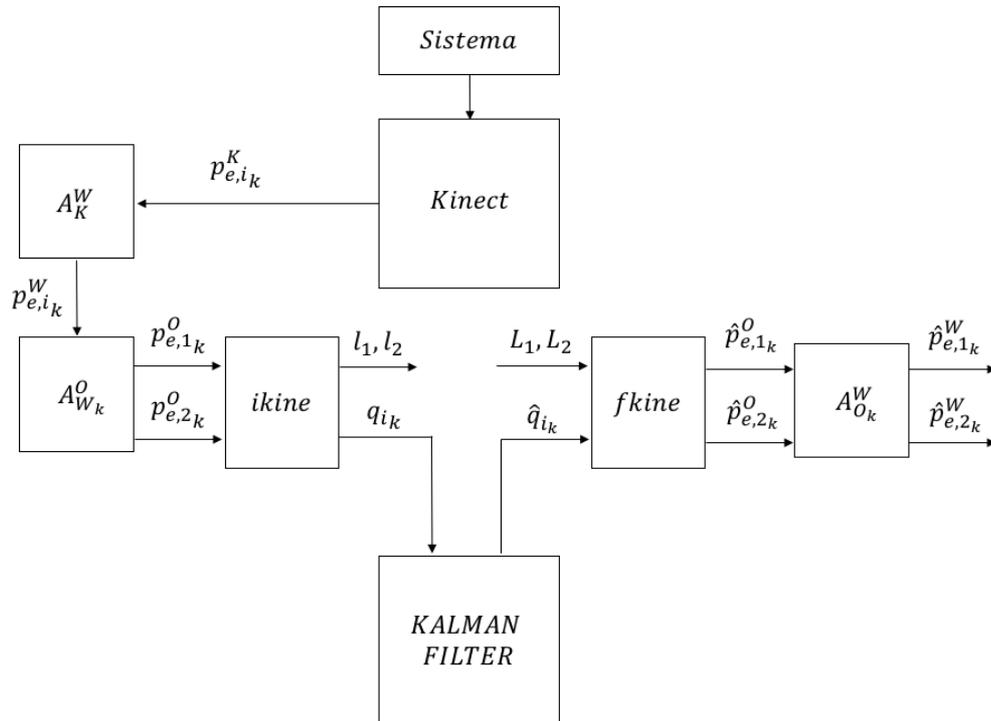


Figura 4.4 Schema a blocchi, elaborazione dati Kinect

4.1.3 Utilizzo di Kinect singola o doppia

Si è detto come si possano utilizzare due Kinect in contemporanea. In questo caso le Kinect vengono collegate a due pc differenti che acquisiscono i dati separatamente. Le acquisizioni vengono allineate mediante una procedura di sincronizzazione. Risulta dunque possibile confrontare i due scheletri e, mediante un algoritmo di ottimizzazione, estrapolare uno scheletro ottimo. Le prove eseguite nel seguente lavoro sono realizzate mediante l'utilizzo di una singola Kinect per permettere la migliore valutazione delle performance dell'algoritmo sviluppato per la miglior stima delle posizioni dei giunti dello scheletro.

4.1.4 Prove eseguite e acquisizione dati Kinect

Tra i vari output che fornisce Kinect, si è interessati all'acquisizione della struttura *depthMetaData*. Tale struttura contiene diversi campi tra cui *AbsTime* contenente l'istante temporale in cui ogni frame è stato acquisito, *IsBodyTracked* che indica se e quale scheletro è rilevato e *JointPositions* nel quale sono presenti le coordinate di tutti i giunti rilevati nel sistema di riferimento K di Kinect. Ogni giunto è

caratterizzato da un indice che lo identifica. Con riferimento alla Figura 1.6, gli indici dei giunti dello scheletro di interesse sono presentati in Tabella 4.1. Tali indici non variano con l'utilizzo di Kinect v1 e Kinect v2, in quanto la notazione utilizzata da Kinect v2 conserva quella precedente: i punti aggiuntivi che Kinect v2 è in grado di rilevare, hanno indici che proseguono la numerazione della versione precedente.

Tabella 4.1 Indici dei giunti spalla, gomito e polso

Joint	Index
ShoulderLeft	5
ElbowLeft	6
WristLeft	7
ShoulderRight	9
ElbowRight	10
WristRight	11

In Tabella 4.2 si elencano le prove sperimentali utilizzate nel lavoro con le loro caratteristiche.

Tabella 4.2 Prove sperimentali eseguite con Kinect

	Denominazione	Durata	Descrizione
Prova 0.1	Mov_G1	30 s	Mov. lento verso alto e basso con braccio disteso
Prova 0.2	Mov_G1_2_2	50 s	Mov. veloci verso alto e basso ripetuto, postura 1
Prova 0.3	Mov_G1_3	50 s	Mov. veloci verso alto e basso ripetuto, postura 2
Prova 1	Test_05_KB	35 s	Movimenti casuali lenti
Prova 2	Test_06_KB	70 s	Movimenti casuali prima lenti poi veloci
Prova 3	Test_03_KB	20 s	Movimento prima casuale poi rettilineo

Le prove 0 sono state effettuate precedentemente con un sensore Kinect v1 con un set-up differente che presentava un solo sensore. Esse vengono utilizzate per un'analisi preliminare insieme al modello multi-body.

Tutte le altre sono eseguite con Kinect v2 (KB) secondo il set-up precedentemente descritto. Le prove 1 e 2 sono utilizzate per indagare il comportamento dinamico del modello (paragrafo 4.4). La prova 3 è eseguita non solo da Kinect, ma anche dal sensore Optitrack (paragrafo 4.9) per indagare la precisione ottenibile dall'utilizzo dell'algoritmo.

4.2 Problematiche delle prove sperimentali

4.2.1 Problematica dell'intervallo temporale Δt di Kinect

Nominalmente l'intervallo di tempo Δt , fra un frame e l'altro, è costante ed è pari all'inverso della frequenza di campionamento di Kinect $f = 30$ Hz, ovvero $\Delta t = 1/f \approx 33.3$ ms

Analizzando i dati sperimentali (MOV_G1_3) si è riscontrato che questa assunzione non è sempre vera e che i valori veri di Δt variano. Per i dati analizzati risulta $\Delta t = 32,6 \pm 2,4$ ms (68%).

L'intervallo Δt viene utilizzato nel filtro di Kalman in fase di predizione comparando nell'Eq. (3.28) nella definizione della matrice \mathbf{G} e nell'Eq. (3.41) nella definizione della matrice \mathbf{Q} . In fase di predizione si è assunto $\Delta t = 1/f = \text{cost.}$ in quanto, in applicazioni real-time, non è possibile conoscere la durata dell'intervallo Δt successivo: risulta pertanto lecito adottare il suo valore nominale.

4.2.2 Problematica delle lunghezze misurate

Già nel paragrafo 3.5.1, si è accennato alla possibilità di effettuare l'operazione di cinematica diretta, a valle del filtro di Kalman, mediante dei valori di lunghezza braccio L_1 e avambraccio L_2 diversi da quelli misurati istantaneamente l_1 e l_2 .

Con i dati simulati conosco i valori di lunghezza braccio L_1 e avambraccio L_2 con i quali il modello è stato costruito. I valori misurati l_1 e l_2 saranno diversi per ogni istante a causa del rumore introdotto.

Con i dati sperimentali non conosco i valori veri delle lunghezze e ottengo dei valori misurati l_1 e l_2 diversi per ogni istante e che risultano molto nervosi con scatti repentini. In Figura 4.6 si mostrano due frame consecutivi di un'acquisizione che evidenziano la variabilità delle lunghezze misurate l_1 e l_2 . Questo comportamento genera inesattezze sulla posizione dei giunti.

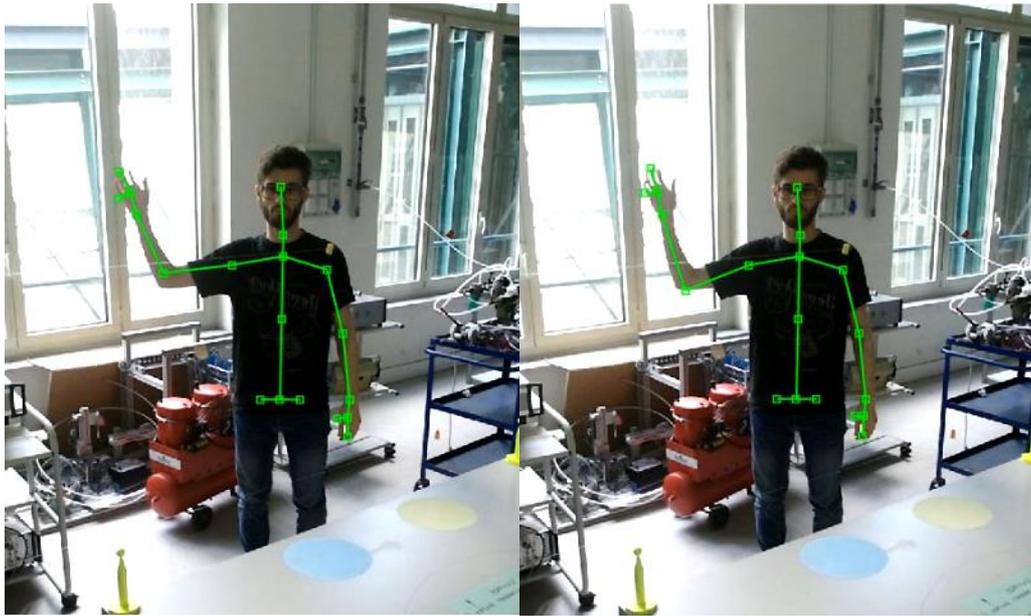


Figura 4.5 Frame consecutivi che evidenziano una variazione repentina delle lunghezze di braccio l_1 e avambraccio l_2

Emerge dunque il problema di dover effettuare una stima sulle lunghezze di braccio L_1 e avambraccio L_2 .

Si sviluppano le seguenti procedure al fine di valutarne i risultati:

- Utilizzo del valore istantaneo misurato, ovvero $L_{1,k} = l_{1,k}$ e $L_{2,k} = l_{2,k}$;
- Utilizzo di valori fissi di L_1 e L_2 , ad esempio quelli misurati alla prima iterazione, ovvero $L_{1,k} = l_{1,1}$ e $L_{2,k} = l_{2,1}$;
- Utilizzo di medie mobili, relative ai valori misurati precedentemente, ovvero $L_{1,k} = \bar{l}_{1,1:k}$ e $L_{2,k} = \bar{l}_{2,1:k}$;

Si indaga il problema utilizzando l'algoritmo che presenta il filtro di Kalman con modello cinematico, applicato alla prova 0.3.

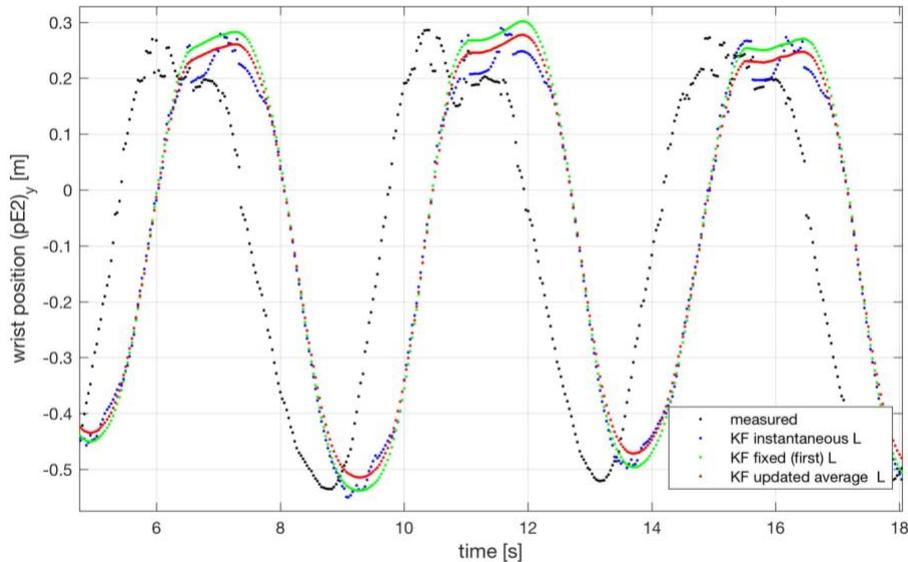


Figura 4.6 Posizioni del polso su asse y ($\mathbf{p}_{e,2}^0$)_y misurate e stimate con L istantanee, L fisse, L medie mobili

La Figura 4.6 mostra l'andamento nel tempo della coordinata y delle posizioni del polso $\mathbf{p}_{e,2}^0$. In generale si nota un ritardo prodotto dal filtro di Kalman, che sarà oggetto di studio nel paragrafo 4.4. È possibile constatare che l'utilizzo di misure istantanee di L_1 e L_2 produce una stima più nervosa, mantenendo la poca accuratezza dei dati grezzi del sensore. Si ricorda che l'azione del filtro di Kalman si concentra sulla stima delle variabili di giunto q_i che, attraverso la cinematica diretta, porta alle posizioni dei giunti nello spazio operativo. Nell'operazione di cinematica diretta il subentro di un termine affetto da importante rumore, compromette i tutti i risultati ottenuti.

Utilizzando un valore fisso si evita il pericolo che il rumore sulla misura di l_1 e l_2 si rifletta sull'output. Tuttavia assumere un valore a priori come ad esempio del primo istante, può risultare poco accurato. L'utilizzo di una media mobile produce risultati analoghi in termini di filtraggio e presenta un'accuratezza maggiore. L'accuratezza ottenuta dipende comunque dal tipo di movimenti prodotti e dalla vicinanza al sensore. È possibile prevedere di effettuare un'operazione di taratura di L_1 e L_2 a priori rispetto il funzionamento dell'algoritmo e di utilizzare quindi valori delle lunghezze fisse. In questo modo nell'operazione di taratura si effettuerebbero quei movimenti sui quali il sensore Kinect è più accurato: movimenti lenti svolti su un piano frontale al sensore e a breve distanza.

L'assunzione di valori fissi di L_1 e L_2 , basata su una stima preliminare, risulterebbe dunque l'opzione più consigliata, in quanto quella più accurata.

Si è mostrato come l'imposizione di lunghezze fisse o calcolate mediante media mobile di L_1 e L_2 , introduca un effetto filtrante sui risultati finali relativi alle posizioni nello spazio operativo. Il filtro di Kalman introduce, inoltre, un ritardo.

Si indaga dunque l'effetto che avrebbe la sola imposizione delle lunghezze fisse o mediate sulle posizioni stimate nello spazio operativo. Si costruisce un algoritmo privo di KF, in grado di effettuare solo cinematica inversa e tornare, con la cinematica diretta, usando i valori mediati delle lunghezze, alle posizioni nello spazio operativo.

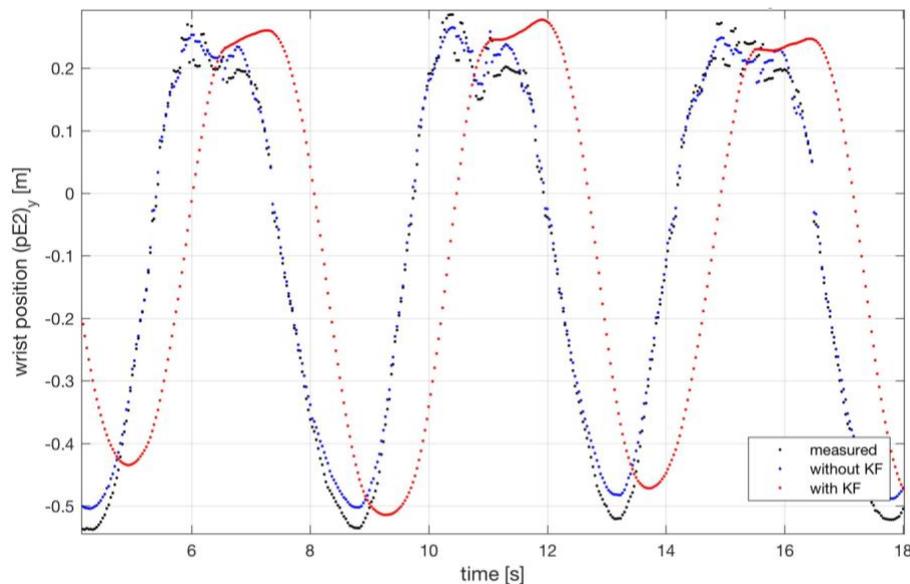


Figura 4.7 Posizioni del polso $(\mathbf{p}_{e,2}^0)_y$ su asse y misurate e stimate con e senza filtro di Kalman (KF)

Dalla Figura 4.7 emerge come il KF risulta essenziale per un filtraggio adeguato della misura. Per ulteriori approfondimenti si rimanda al paragrafo 4.6.

4.3 Confronto modello dinamico e cinematico

Si vogliono confrontare gli algoritmi con UIEKF basato su un modello dinamico e KF basato su un modello cinematico al fine di valutare quale risulti più performante. I due algoritmi vengono testati sulle stesse prove simulate e sperimentali.

4.3.1 Confronto basato sui dati simulati

Mediante il modello del braccio realizzato su Simscape è possibile simulare i dati misurati dal sensore (paragrafo 3.2.5). Si fornisce come input l'andamento delle variabili di giunto nel tempo, specificate nelle Eq. (3.16). Si adottano degli andamenti rettilinei per le variabili q_1 e q_3 e degli andamenti sinusoidali per q_2 e q_4 . Gli andamenti dati in input rappresentano i valori veri delle variabili di giunto. L'analisi con dati simulati ha proprio il vantaggio di poter confrontare la stima effettuata con il valore vero delle variabili che descrivono il sistema osservato. In Figura 4.8 sono mostrate le variabili di giunto q_i nel tempo misurate e stimate mediante il modello dinamico del UIEKF.

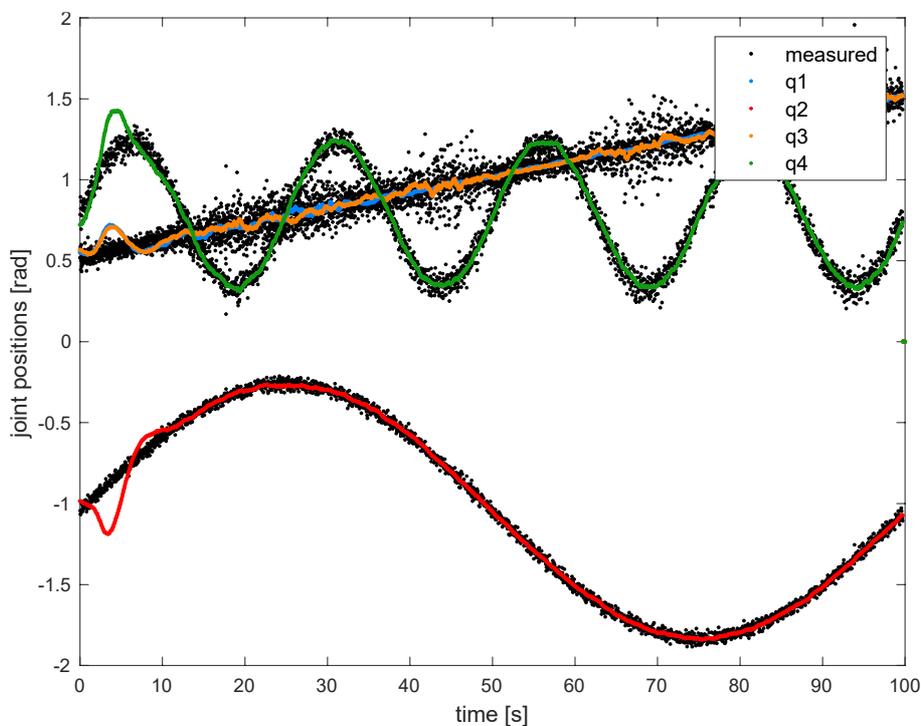


Figura 4.8 Variabili di giunto q misurate e stimate con UIEKF, dati simulati

Emerge che l'UIEKF impiega un certo lasso di tempo per andare a regime. Nel caso specifico si registra un transitorio di circa 10 s. Si evidenzia come la misura, nel caso di dati simulati, sia affetta da un rumore gaussiano che si distribuisce uniformemente intorno al valore vero.

Una volta sviluppato il KF con modello cinematico, lo si applica nella stessa situazione a partire dagli stessi dati misurati. In Figura 4.9 e Figura 4.10 si mostra un confronto fra UIEKF e KF, ovvero fra modello dinamico e cinematico, riferito a due specifiche variabili di giunto che evidenziano più chiaramente le differenze.

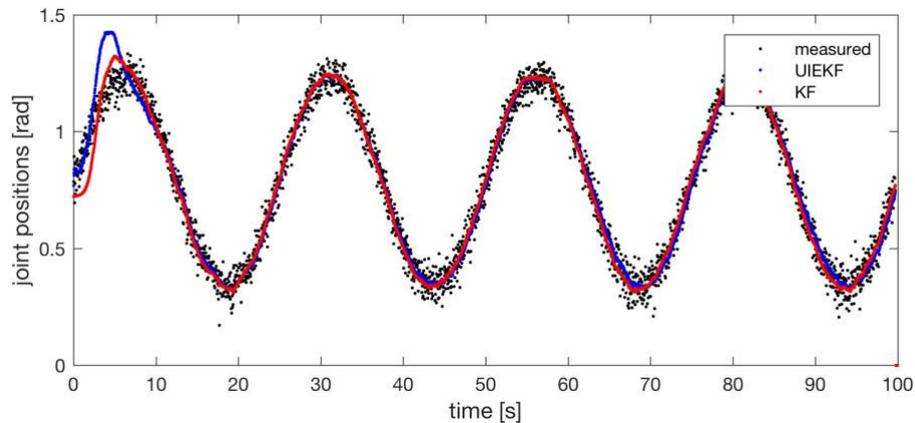


Figura 4.9 Variabile di giunto q_4 misurata e stimata con UIEKF e KF, dati simulati

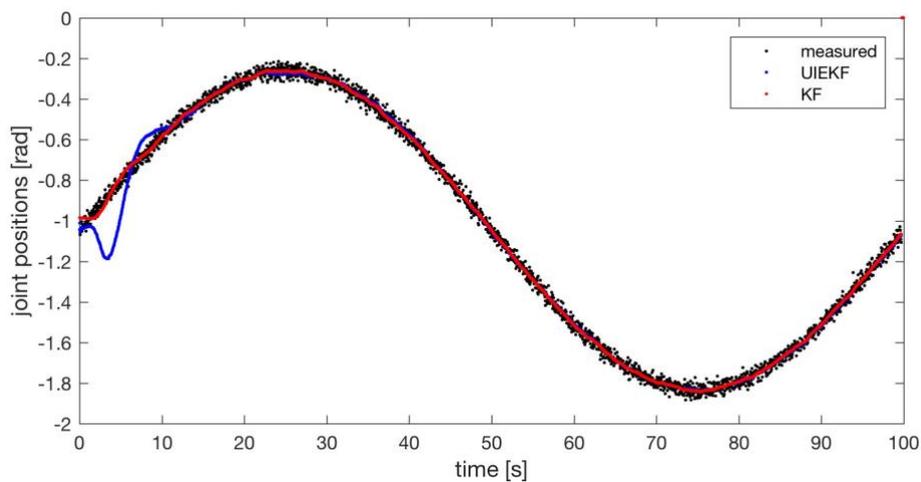


Figura 4.10 Variabile di giunto q_2 misurata e stimata con UIEKF e KF, dati simulati

I risultati a regime sono paragonabili, ma il transitorio del KF basato su modello cinematico è ridotto rispetto al UIEKF basato su modello dinamico. Si registra, inoltre, un'importante differenza in termini di tempi di calcolo che risultano per il

modello cinematico un ordine di grandezza più bassi: 2 ms contro 20 ms circa ad iterazione. La differenza in termini di complessità dei due algoritmi si riflette sui tempi di calcolo e transitorio, senza produrre effetti su una migliore accuratezza dovuta alla complessità del modello.

Note le variabili di giunto q_i nel tempo, è nota anche la loro velocità. Si esegue un simile confronto sulle velocità \dot{q}_i stimate dai due modelli.

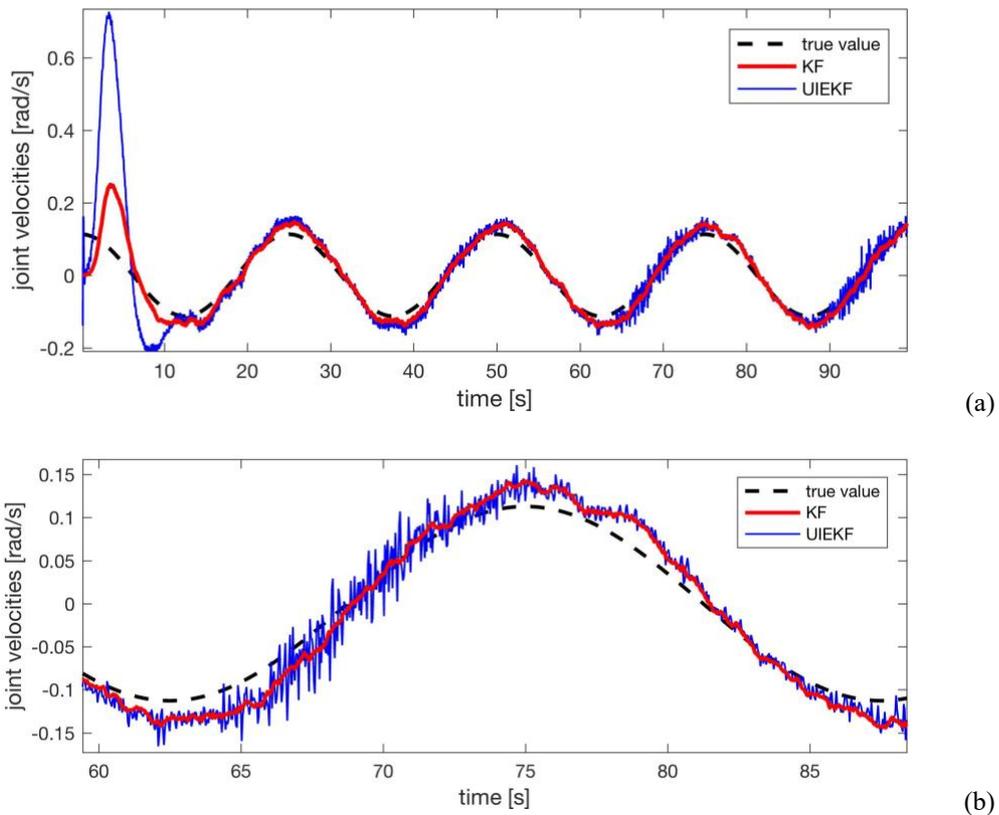


Figura 4.11 Velocità \dot{q}_4 misurata e stimata con UIEKF e KF (a) e ingrandimento (b), dati simulati

In Figura 4.11 viene mostrato un confronto fra le velocità \dot{q}_4 stimate da modello cinematico e dinamico con quelle vere. Si evidenzia ancora una volta un transitorio più marcato per il modello dinamico. Inoltre anche in fase di regime le velocità stimate attraverso il modello cinematico risultano affette da meno rumore, seppur ripercorrendo lo stesso andamento di quelle stimate con il modello dinamico. A fronte di questa osservazione a maggior ragione risulta più adeguato l'uso di un modello cinematico: più leggero e più efficace.

Ottenuta la stima delle q_i è possibile ricavare le posizioni del gomito $\mathbf{p}_{e,1}^0$ e polso $\mathbf{p}_{e,2}^0$ attraverso cinematica diretta utilizzando le lunghezze note L_1 e L_2 . Le osservazioni effettuate per le q_i stimate si riflettono nello spazio operativo. In Figura 4.12 si mostra una rappresentazione 3D della traiettoria seguita dal polso $\mathbf{p}_{e,2}^0$ nella prova simulata e in Figura 4.13 l'andamento nel tempo della sola coordinata z . In entrambe le rappresentazioni è evidente il transitorio del UIEKF con modello dinamico. Nello spazio operativo il transitorio del KF con modello cinematico non risulta neanche percepibile.

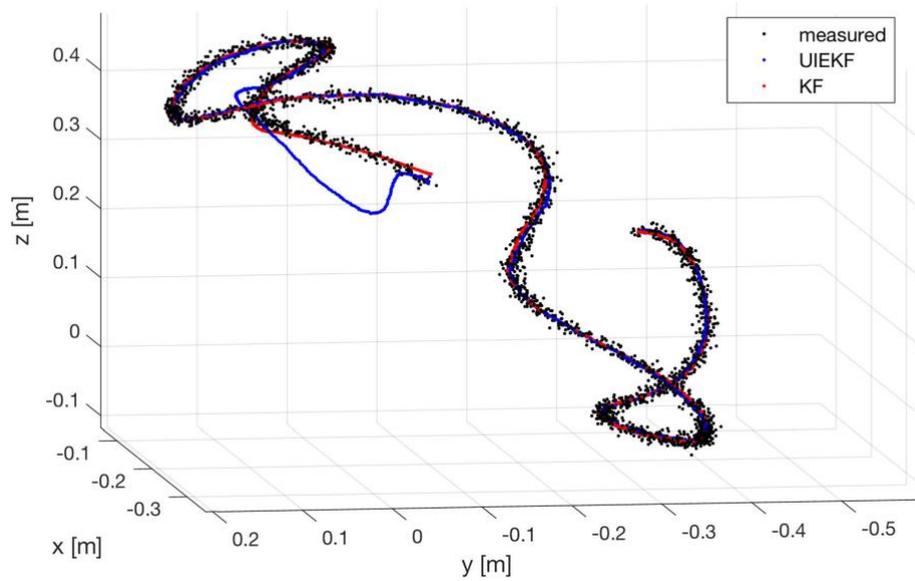


Figura 4.12 Visualizzazione 3D della traiettoria del polso $\mathbf{p}_{e,2}^0$ misurata e stimata con UIEKF e KF, dati simulati

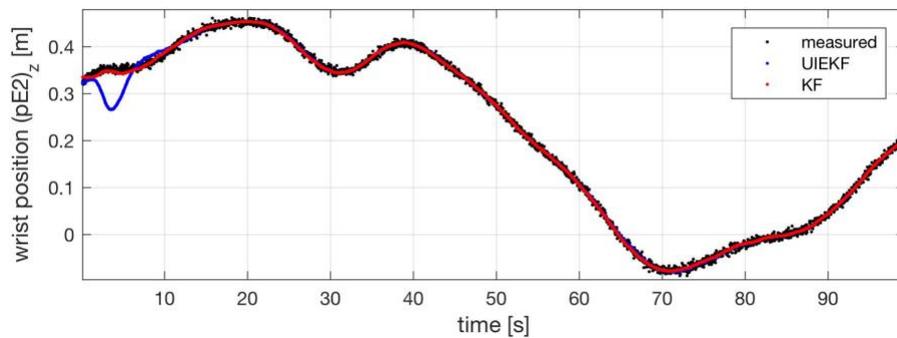


Figura 4.13 Coordinata z della posizione del polso $(\mathbf{p}_{e,2}^0)_z$ misurata e stimata con UIEKF e KF, dati simulati

Viene effettuata un'ulteriore analisi per comprendere l'accuratezza della stima e l'errore effettuato dalla fase di predizione fornita dai due modelli. È possibile confrontare ad ogni istante il valore dello stato predetto con quello vero, il quale è noto lavorando con dati simulati. È inoltre possibile lavorare solo in fase di predizione, effettuando l'operazione con diversi orizzonti predittivi.

Per prima cosa viene verificata l'accuratezza delle posizioni di giunto q_i dopo un certo lasso di tempo, esaurito il transitorio. Le analisi verranno effettuate confrontando le posizioni di giunto stimate con quelle vere. In Figura 4.14 si sovrappongono tutti gli errori relativi a tutte le variabili di giunto stimate con UIEKF e KF. Come sostenuto precedentemente, non si notano importanti differenze.

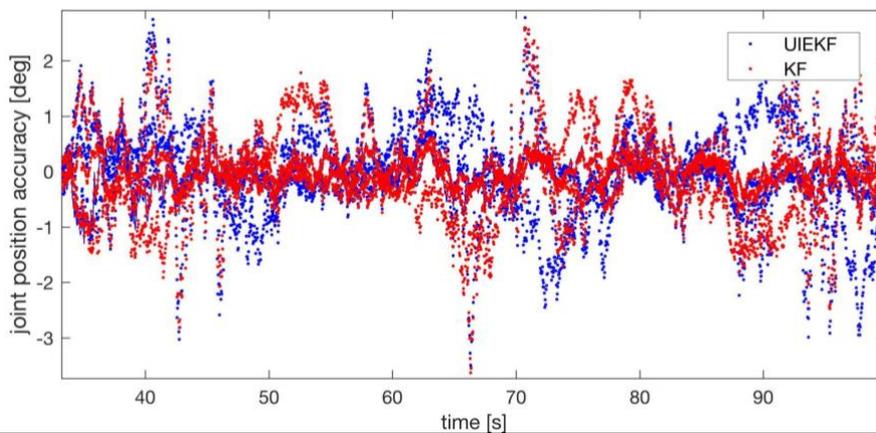


Figura 4.14 Accuratezza della stima delle q_i , dati simulati

Si confronta adesso la predizione a 3, 6 e 9 istanti successivi (ovvero rispettivamente a 1 s, 2 s, 3 s), fornita dai due modelli, con i valori delle q_i vere relative agli istanti di cui si è eseguita la predizione. In questo modo è possibile valutare quanto un modello riesca a prevedere con efficacia il valore dello stato del sistema.

In Figura 4.15 si mostrano i risultati ottenuti. Si nota come il modello cinematico riesca a prevedere in modo più accurato rispetto quello dinamico. La differenza aumenta con l'allungamento dell'orizzonte predittivo.

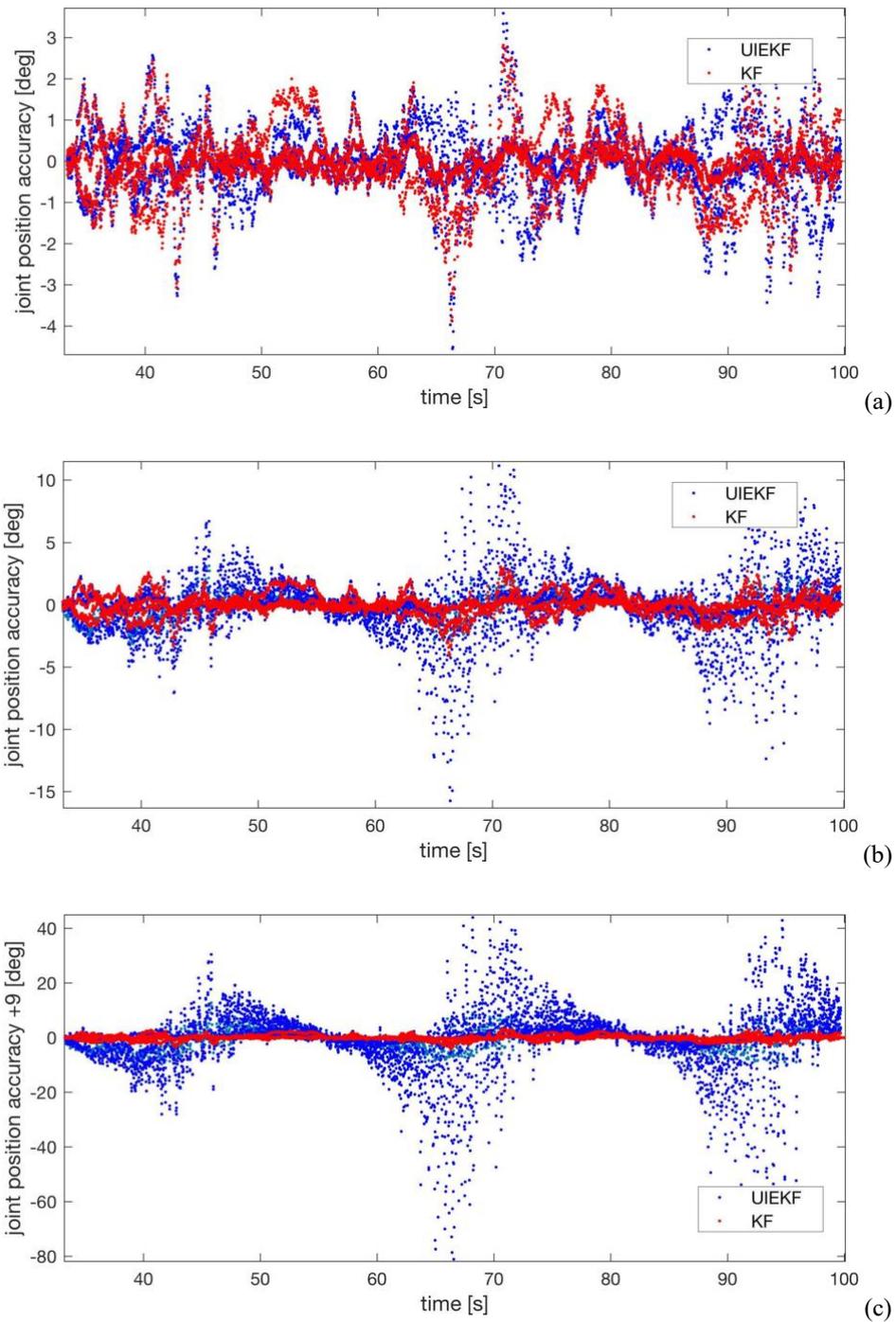


Figura 4.15 Errori delle predizioni con anticipo di 3 (a), 6 (b) e 9 (c) istanti temporali, prove simulate

A fronte delle considerazioni effettuate in merito al transitorio, ai tempi di calcolo, alla stima delle velocità e all'accuratezza della predizione, si ritiene che il modello cinematico sia più idoneo rispetto quello dinamico.

4.3.2 Confronto basato sui dati sperimentali

Le osservazioni effettuate attraverso l'analisi dei dati simulati devono essere verificate anche sperimentalmente a causa della più complessa natura del movimento naturale del braccio umano e dal maggiore nervosismo dei dati forniti dal sensore Kinect. Si eseguono entrambi gli algoritmi basati su modello dinamico e cinematico sia con una prova con movimenti veloci (prova 0.3), che una con movimenti lenti (prova 0.1).

Si analizza prima la prova più veloce. In Figura 4.16 vengono mostrate le stime sulle variabili di giunto, riferite alla prova 0.3, eseguite mediante il modello cinematico. Si noti come i dati misurati dal sensore Kinect sperimentalmente siano diversi rispetto a quelli simulati: presentano salti bruschi e andamento fortemente irregolare. Il filtro di Kalman, rende l'andamento delle q_i più continuo, ma introduce un ritardo.

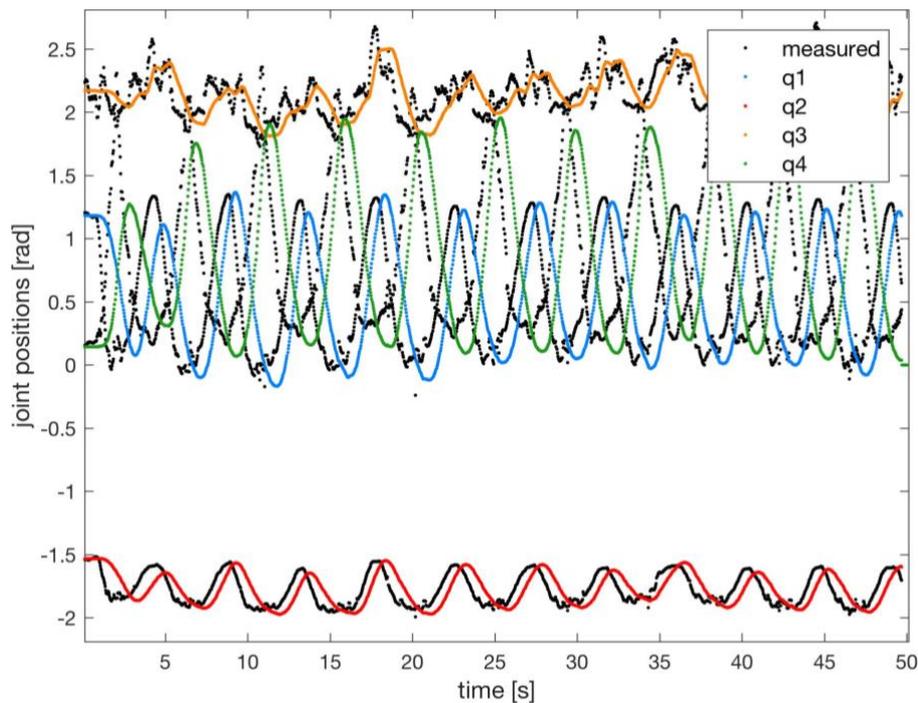


Figura 4.16 Variabili di giunto q misurate e stimate con KF con modello cinematico, prova 0.3

In Figura 4.17 e Figura 4.18 si confrontano i due modelli sulla stima delle variabili di giunto q_4 e q_2 . Ancora una volta si registra un transitorio maggiore per

il modello dinamico, ma a regime nulla si può dire su quale dei due esegua una migliore stima.

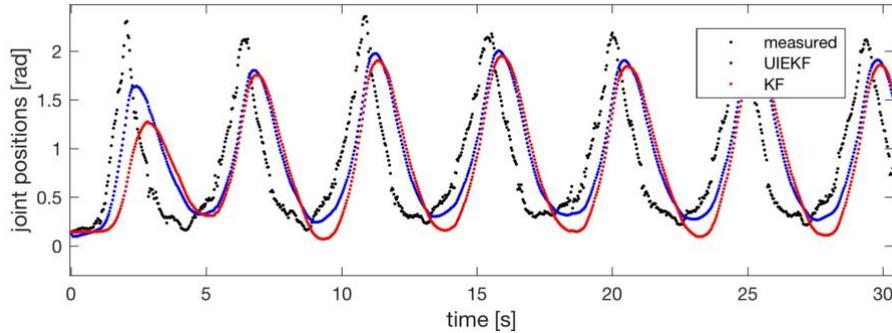


Figura 4.17 Variabile di giunto q_4 misurata e stimata con UIEKF e KF, prova 0.3

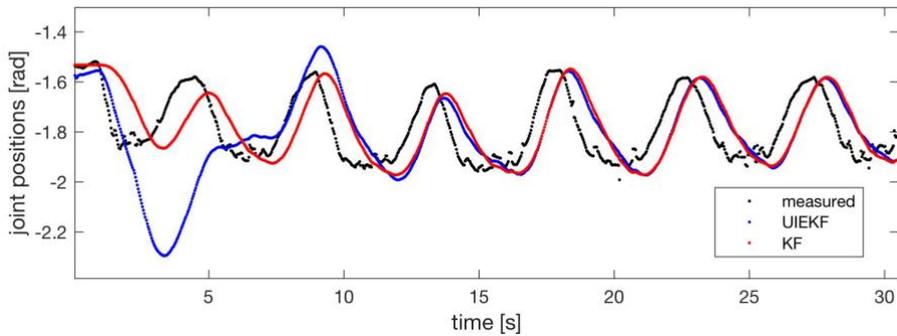


Figura 4.18 Variabile di giunto q_2 misurata e stimata con UIEKF e KF, prova 0.3

Si utilizza ora una prova con un movimento eseguito più lento (prova 0.1) e si esegue la medesima analisi.

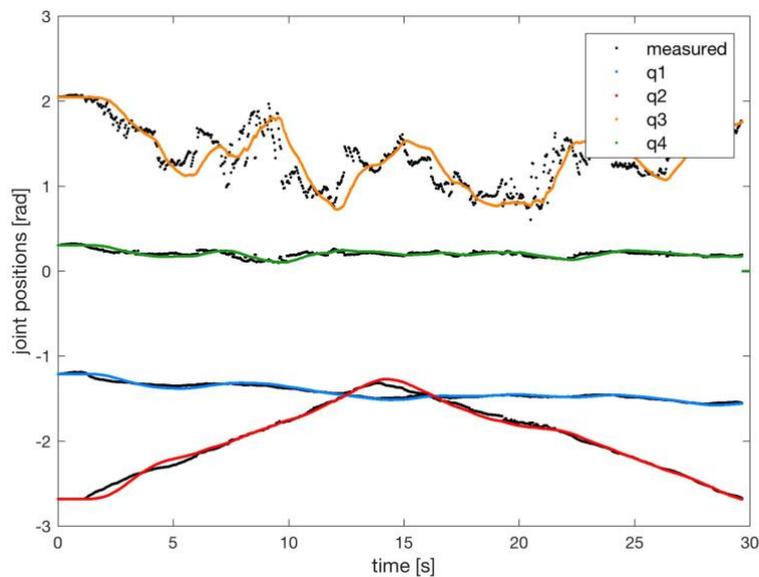


Figura 4.19 Variabili di giunto q misurate e stimate con KF con modello cinematico, prova 0.1

In Figura 4.19 vengono mostrate le stime sulle variabili di giunto, riferite alla prova 0.1, eseguite mediante il modello cinematico. Avendo delle dinamiche più basse, il filtro di Kalman non produce un ritardo apprezzabile su q_1 , q_2 e q_4 .

In Figura 4.20 si presenta il confronto fra i due modelli sulla stima di q_2 . Il modello dinamico presenta un transitorio di circa 10 s con amplificazione notevole. A regime i due segnali non presentano differenze apprezzabili.

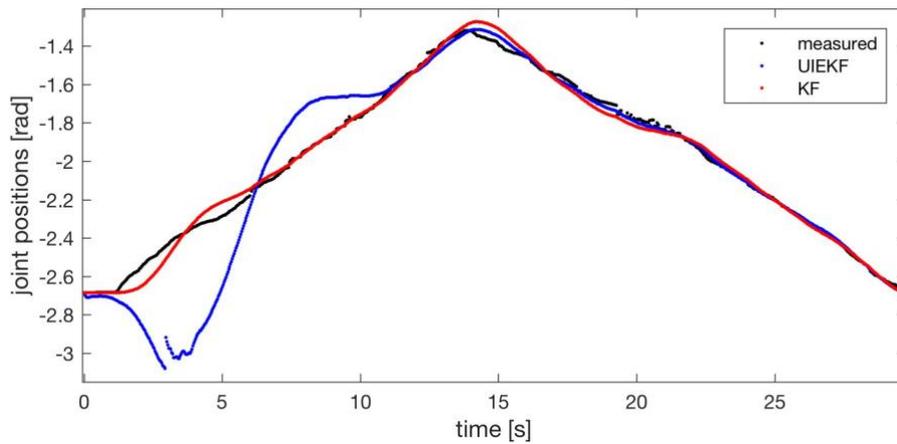


Figura 4.20 Variabile di giunto q_2 misurata e stimata con UIEKF e KF, prova 0.1

I tempi di calcolo per iterazione sono in media di 0,57 ms per il KF e di 18 ms per l'UIEKF.

È possibile infine eseguire la cinematica diretta. In questo caso si utilizza come stima per le lunghezze L_1 e L_2 la media mobile.

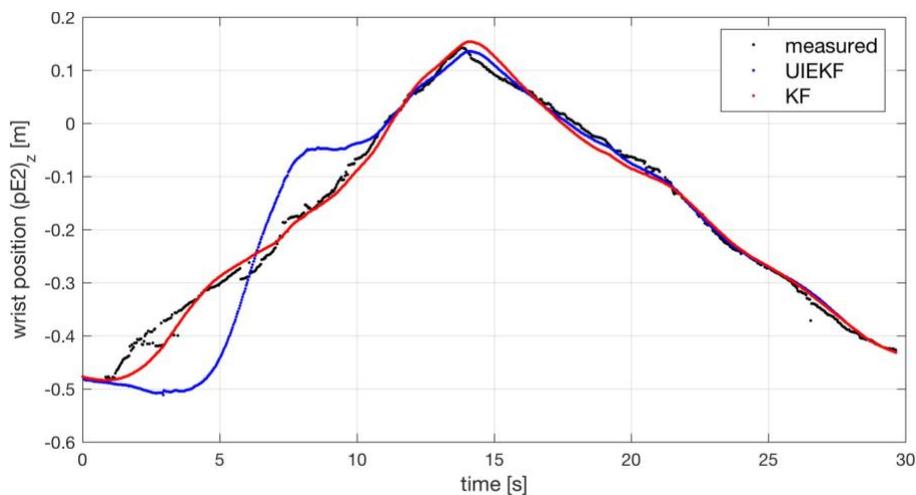


Figura 4.21 Coordinata z della posizione del polso $(p_{e,2}^0)_z$ misurata e stimata con UIEKF e KF, prova 0.1

In Figura 4.21 si mostra l'andamento nel tempo della coordinata z della posizione del polso $(\mathbf{p}_{e,2}^0)_z$ stimata dal modello cinematico e dinamico.

4.3.3 Conclusioni sul modello da adottare

Dall'analisi dei dati simulati e sperimentali è emerso come l'utilizzo di un UIEKF con modello dinamico, per la sua complessità, risulta molto più oneroso dal punto di vista dei tempi di calcolo per ogni iterazione. Esso, inoltre, presenta delle fasi di transitorio più lunghe rispetto quelle prodotte dal filtro di Kalman basato su modello cinematico. A regime la differenza fra le posizioni stimate dai due modelli non è apprezzabile, ma il modello cinematico dimostra di ottenere migliori stime sulla velocità dei giunti \dot{q}_i e migliore capacità predittiva.

Ci si sarebbe potuto aspettare che l'utilizzo di un modello dinamico potesse essere più accurato. Il modello però prevedeva diverse assunzioni, quale la stima degli input ignoti \mathbf{u} e la stima delle velocità $\dot{\mathbf{q}}$ effettuata a sua volta mediante un modello cinematico. Ulteriori approssimazioni erano indotte dalla fase di linearizzazione.

Nel seguito si procederà ad analizzare e sviluppare l'algoritmo con filtro di Kalman basato sul modello cinematico. Per la cinematica diretta verranno utilizzati, se non diversamente specificato, valori fissi di L_1 e L_2 , in quanto si presume derivanti da una procedura di taratura con migliore accuratezza rispetto la stima che può fornire una media mobile.

4.4 Risposta dinamica e matrici di covarianza

Sia nelle prove simulate che in quelle sperimentali è emersa un'azione ritardante del filtro di Kalman. Si vuole dunque analizzare il comportamento dinamico del Filtro di Kalman.

Allo scopo si utilizza il modello Simscape del braccio per produrre traiettorie controllate del braccio umano. Nello specifico si interviene modificando il moto di un solo giunto (q_4) e si analizzano le performance del filtro riferite ad esso.

Si impone una legge nel tempo sinusoidale e si analizzano gli effetti prodotti dalla variazione della sua frequenza f :

$$q_4 = \frac{\pi}{7} \sin(2\pi f t) + \frac{\pi}{4} \quad (4.14)$$

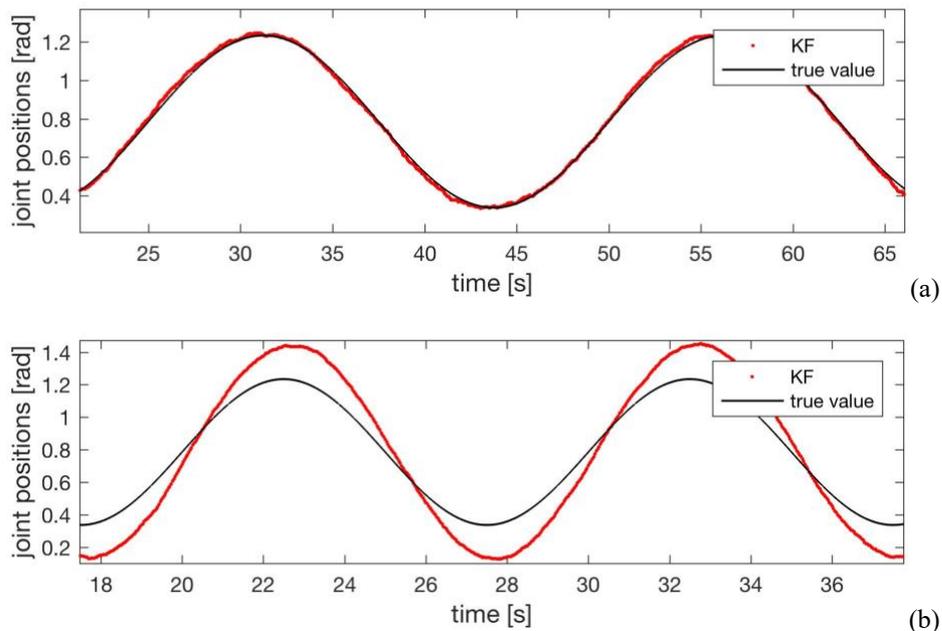
dove

$$\begin{cases} f_1 = 0,04 \text{ Hz} \\ f_2 = 0,10 \text{ Hz} \\ f_3 = 0,20 \text{ Hz} \\ f_4 = 0,40 \text{ Hz} \\ f_5 = 1,00 \text{ Hz} \end{cases} \quad (4.15)$$

Per ogni valore di frequenza f viene eseguita una simulazione su Simscape. La frequenza f_1 è quella adottata in precedenza. Viene dunque aumentato progressivamente il valore di f per valutare il comportamento della stima prodotta dal filtro di Kalman a seguito di un segnale misurato con dinamiche maggiori.

In

Figura 4.22 si mostrano i risultati ottenuti. Si presentano due creste successive relative alla sinusoida generata dall'andamento di q_4 nel tempo, in modo da evidenziare la posizione reciproca fra la stima prodotta e il valore vero.



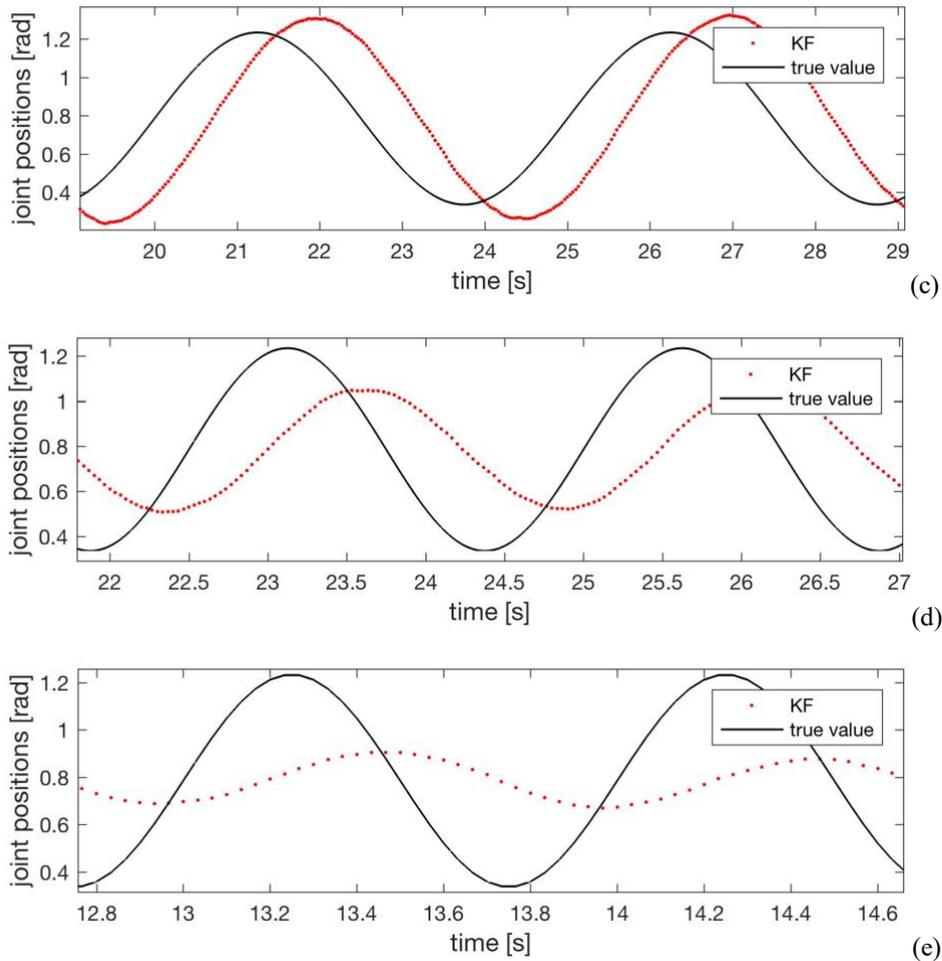


Figura 4.22 Variabile di giunto q_4 stimata con KF e simulata con frequenza f_1 (a), f_2 (b), f_3 (c), f_4 (d) e f_5 (e)

Risulta che per f_1 si ha un andamento della stima aderente a quello vero, senza amplificazioni o ritardi apprezzabili. Per f_2 si ha un'amplificazione del segnale con un minimo ritardo. Per f_3 l'amplificazione si riduce e aumenta il ritardo. Per f_3 e f_5 il segnale stimato attenua progressivamente, rimanendo sempre in ritardo.

Attraverso i risultati ottenuti viene prodotto un diagramma di Bode, rappresentato in Figura 4.23. Nelle prossimità del valore di f_2 si ha una risonanza. Lo sfasamento cresce al crescere della frequenza f . Per dinamiche troppo elevate il filtro di Kalman non riesce a seguire efficacemente il segnale, attenuandolo. In sostanza il KF si comporta come un sistema dinamico.

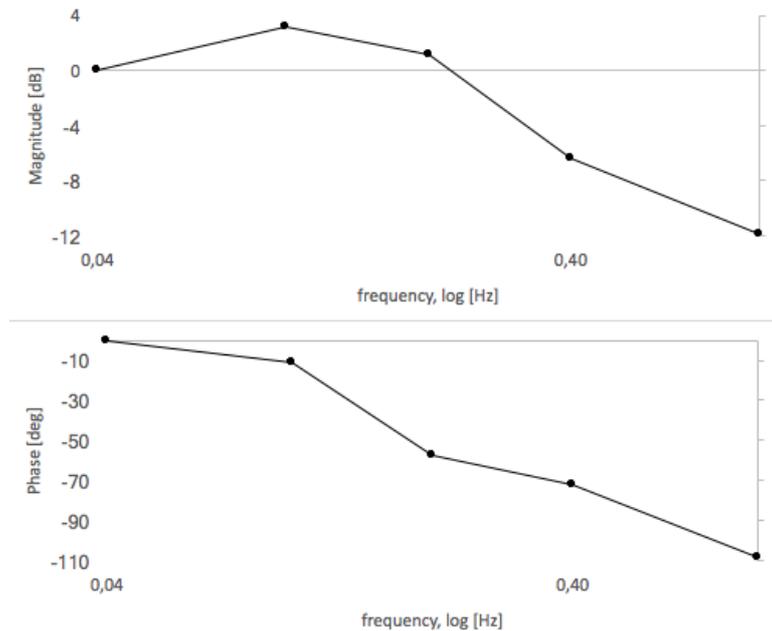


Figura 4.23 Diagramma di Bode di q_4 stimata con KF

Si rende necessario indagare quali parametri, che caratterizzano il filtro di Kalman, intervengano sulla sua prestazione dinamica. In particolare è di interesse allontanare la frequenza di risonanza del filtro di Kalman dalla frequenza propria dei movimenti eseguiti dal braccio umano.

Vengono individuati due parametri principali sui quali è possibile intervenire: la matrice di covarianza del processo \mathbf{Q} e la matrice di covarianza della misura \mathbf{R} .

4.4.1 Effetto matrice di covarianza di processo \mathbf{Q}

La matrice di covarianza \mathbf{Q} è espressa nella forma esplicitata dall'Eq. (3.41). In tale espressione è presente lo scalare σ_q^2 che moltiplica una data matrice. Il valore di σ_q^2 , detta per semplicità varianza di processo, non è noto a priori ed è stato preso arbitrariamente.

Si indagano gli effetti che la variazione di tale parametro ha sul comportamento dinamico del KF. Si effettuano simulazioni, come in precedenza, variando il valore di σ_q^2 , analizzando la risposta sulla stima di q_4 . Si prende come frequenza della sinusoide $f_3 = 0,20 \text{ Hz}$. Si prende il valore adottato precedentemente $\sigma_{q_3}^2$ e si testano altri valori di σ_q^2 a diversi ordini di grandezza:

$$\left\{ \begin{array}{l} \sigma_{q_1}^2 = 1,5 \cdot 10^{-7} \\ \sigma_{q_2}^2 = 1,5 \cdot 10^{-6} \\ \sigma_{q_3}^2 = 1,5 \cdot 10^{-5} \\ \sigma_{q_4}^2 = 1,5 \cdot 10^{-4} \\ \sigma_{q_5}^2 = 1,5 \cdot 10^{-3} \\ \sigma_{q_6}^2 = 1,5 \cdot 10^{-2} \\ \sigma_{q_7}^2 = 1,5 \cdot 10^{-1} \end{array} \right. \quad (4.16)$$

In Figura 4.24 si mostrano i risultati ottenuti. È evidente come l'utilizzo di varianze σ_q^2 più elevate riduce il ritardo del KF, ma allo stesso tempo ne riduce l'azione filtrante. Per $\sigma_{q_6}^2$ e $\sigma_{q_7}^2$ infatti il KF inizia a presentare dei disturbi evidenti. Per $\sigma_{q_5}^2$ il sistema è prossimo alla risonanza, evidenziando una notevole amplificazione del segnale.

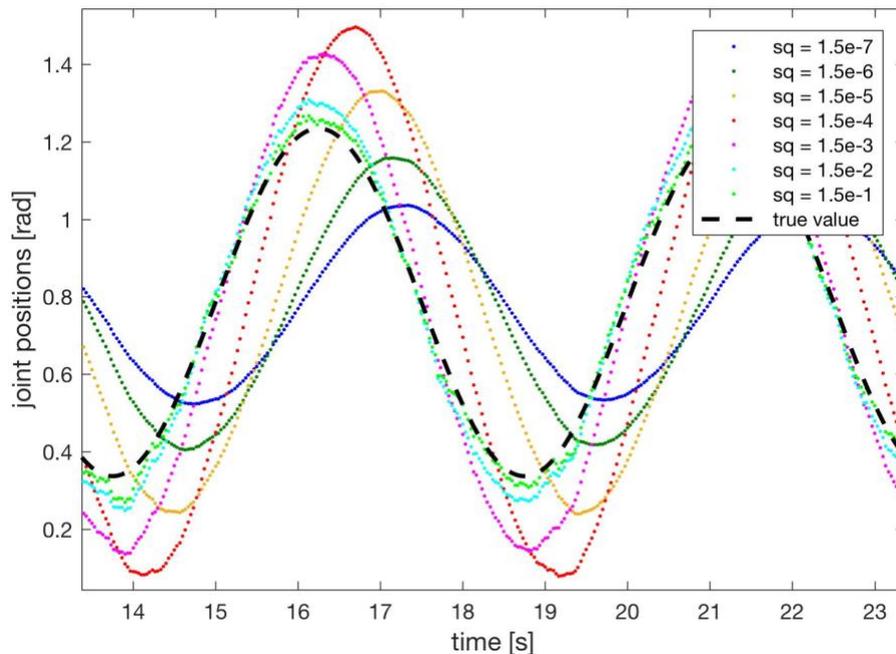


Figura 4.24 Confronto al variare di σ_q^2 delle stime con KF di q_4 , dati simulati con $f = f_3$

L'incremento di σ_q^2 sposta a frequenze più alta la risonanza del sistema, ma diminuisce il potere filtrante del KF. Questo effetto è atteso in quanto la varianza del processo σ_q^2 interviene sulla covarianza della predizione dello stato \mathbf{x} e quindi sul calcolo del guadagno di Kalman \mathbf{K} . Per valori σ_q^2 elevati, prevale quindi il valore

della misura su quello della predizione in fase di aggiornamento e si ha una ridotta azione filtrante.

4.4.2 Effetto matrice di covarianza di misura \mathbf{R}

La matrice di covarianza \mathbf{R} è espressa nella forma esplicitata dall'Eq. (3.42). In tale espressione è presente lo scalare σ_r^2 che moltiplica una matrice identica. Il valore σ_r^2 , non noto a priori ma assunto arbitrariamente, è dunque la varianza della misura.

Anche in questo caso si indagano gli effetti che la variazione di tale parametro ha sul comportamento dinamico del KF. Si effettuano simulazioni, come in precedenza, variando il valore di σ_r^2 , analizzando la risposta sulla stima di q_4 . Si prende come frequenza della sinusoide $f_3 = 0,20 \text{ Hz}$ e si mantiene $\sigma_q^2 = \sigma_{q_3}^2$. Il valore adottato precedentemente è $\sigma_{r_4}^2$. Si testano altri valori di σ_q^2 a diversi ordini di grandezza:

$$\begin{cases} \sigma_{r_1}^2 = 2,5 \cdot 10^{-6} \\ \sigma_{r_2}^2 = 2,5 \cdot 10^{-5} \\ \sigma_{r_3}^2 = 2,5 \cdot 10^{-4} \\ \sigma_{r_4}^2 = 2,5 \cdot 10^{-3} \\ \sigma_{r_5}^2 = 2,5 \cdot 10^{-2} \\ \sigma_{r_6}^2 = 2,5 \cdot 10^{-1} \end{cases} \quad (4.17)$$

In Figura 4.24 si mostrano i risultati ottenuti. È evidente come l'utilizzo di varianze σ_r^2 più basse riduce il ritardo del KF, ma ne riduce l'azione filtrante. Per $\sigma_{r_1}^2$ infatti il KF inizia a presentare dei disturbi evidenti. Per $\sigma_{r_2}^2$ ritroviamo qualche disturbo, un leggero ritardo e una notevole amplificazione. In questo caso il sistema è vicino alla risonanza.

La diminuzione di σ_r^2 sposta a frequenze più alta la risonanza del sistema, ma diminuisce il potere filtrante del KF. Questo effetto è atteso in quanto la varianza della misura σ_r^2 interviene nel calcolo del guadagno di Kalman \mathbf{K} . Per valori σ_r^2 bassi, prevale quindi il valore della misura su quello della predizione in fase di aggiornamento e si ha una ridotta azione filtrante.

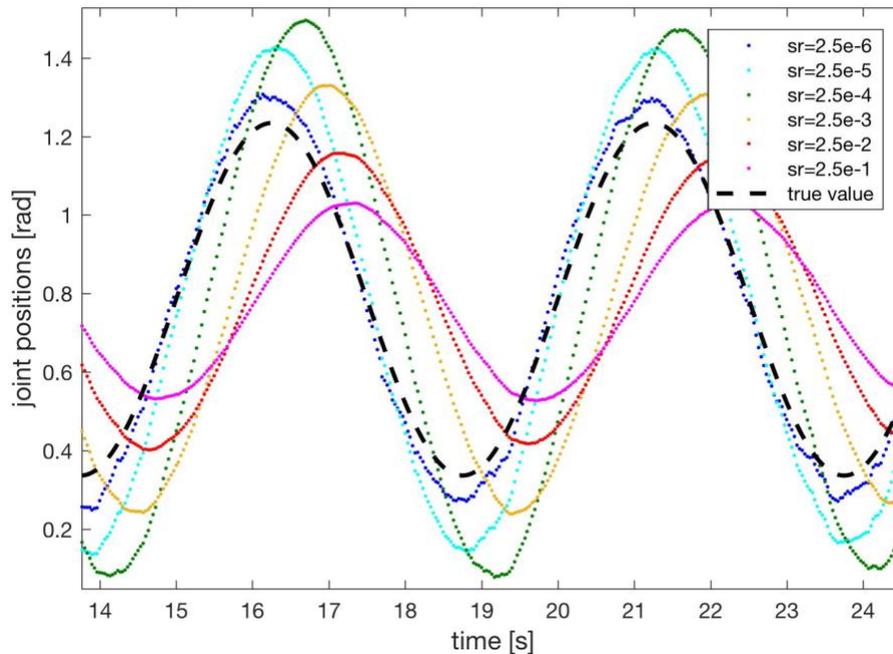


Figura 4.25 Confronto al variare di σ_r^2 delle stime con KF di q_4 , dati simulati con $f = f_3$

Si noti da un confronto come aumentare di un ordine di grandezza σ_r^2 o diminuire di un ordine di grandezza σ_q^2 produca effetti qualitativamente comparabili.

4.4.3 Risposta dinamica con dati sperimentali

Come è emerso dalle analisi effettuate, gli effetti delle varianze σ_q^2 e σ_r^2 sulla risposta dinamica del filtro di Kalman sono complementari. Più è alto il valore di σ_q^2 e più è basso il valore di σ_r^2 , più la frequenza di risonanza del filtro di Kalman si sposta su frequenze più elevate permettendo di ridurre i ritardi, ma diminuendo l'azione filtrante.

Diventa fondamentale effettuare un'azione di regolazione sui parametri σ_q^2 e σ_r^2 al fine di ottenere un'adeguata azione filtrante con il minimo ritardo.

Per poter valutare con efficacia la risposta del filtro di Kalman è necessario effettuare delle prove sperimentali in modo da testare il suo funzionamento con movimenti reali del braccio umano. Vengono eseguite due prove. Nella prova 1 vengono eseguiti movimenti continui, casuali e lenti. Nella prova 2 vengono eseguiti movimenti continui e casuali, prima lenti, poi veloci. In generale infatti il

movimento umano può avere dinamiche più o meno elevate ed è utile valutare i valori più efficaci di σ_q^2 e σ_r^2 adatti per buona parte dei movimenti possibili.

Si analizza la prova 1, presentante movimenti lenti. Si effettua un'analisi al variare di σ_q^2 , mantenendo $\sigma_r^2 = 2,5 \cdot 10^{-3}$. Si utilizzano i valori della prova simulata $\sigma_{q_{2+6}}^2$. In Figura 4.26 si mostrano i risultati relativi all'influenza di σ_q^2 sulla stima di q_4 . I due valori più bassi sembrano introdurre eccessivo ritardo, il più grande presenta eccessivo rumore. Per questo motivo i valori più ragionevoli per σ_q^2 in questa prova specifica sembrano essere nell'ordine di $\sigma_q^2 = 10^{-3} \div 10^{-2}$ nonostante qualche amplificazione del segnale.

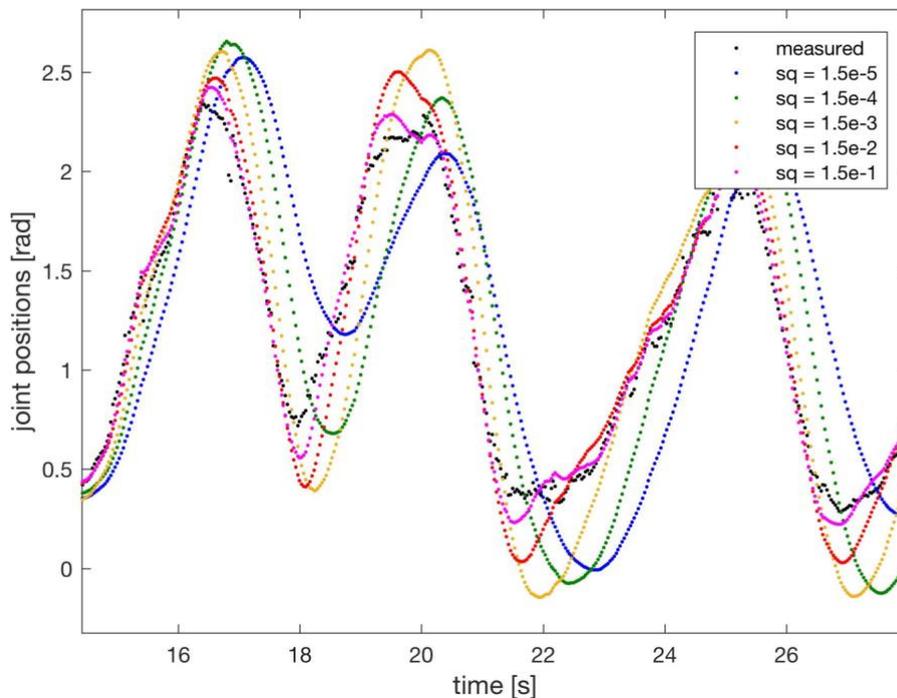


Figura 4.26 Confronto al variare di σ_q^2 delle stime con KF di q_4 , $\sigma_r^2 = 2,5 \cdot 10^{-3}$, prova 1

Sulla stessa prova si effettua un'analisi al variare di σ_r^2 , mantenendo $\sigma_q^2 = cost$. con osservazioni simili a quelle già effettuate.

Risulta più interessante analizzare il comportamento del filtro di Kalman con valori fissi di σ_q^2 e σ_r^2 , ma con un movimento misurato avente una dinamica variabile. Si analizza allo scopo la prova 2 presentante un movimento casuale prima lento e poi veloce. Vengono fatti variare le varianze σ_q^2 e σ_r^2 e osservata la risposta caso per caso su tutti quattro i giunti.

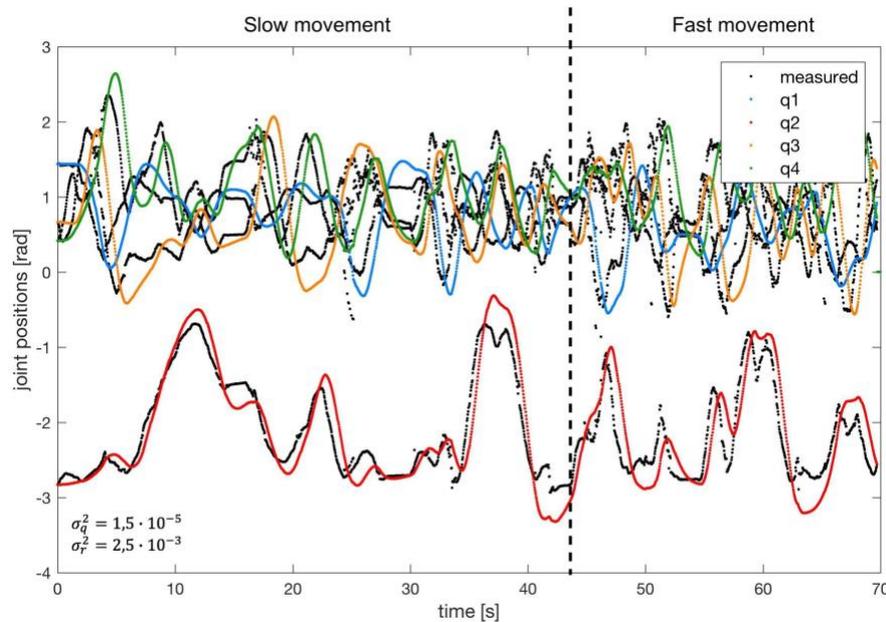


Figura 4.27 Stima delle q_i con KF, $\sigma_q^2 = 1,5 \cdot 10^{-5}$, $\sigma_r^2 = 2,5 \cdot 10^{-3}$, prova 2

In Figura 4.27 si mostrano le stime eseguite dal KF su tutti i giunti con $\sigma_q^2 = 1,5 \cdot 10^{-5}$ e $\sigma_r^2 = 2,5 \cdot 10^{-3}$. Viene evidenziata la parte in cui viene eseguito un movimento più lento e quella dove viene eseguito un movimento più rapido. Si mostra come anche quando viene eseguito un movimento con una stessa velocità percepita dall'operatore, non tutti i giunti presentano stesse dinamiche. Con i valori delle varianze impostate si registra un ritardo su tutti i giunti e in tutti i momenti. Durante il movimento lento il KF va in risonanza in diversi punti, considerazione dimostrata dalle amplificazioni registrate.

Per spostare la frequenza di risonanza del filtro di Kalman a frequenze più alte, come si è visto, una possibile soluzione è l'aumento del valore della covarianza σ_q^2 . Si esegue dunque l'algoritmo con $\sigma_q^2 = 1,5 \cdot 10^{-2}$ i cui risultati sono mostrati in Figura 4.28. Si evidenzia che le escursioni che si registravano precedentemente sui movimenti lenti si sono attenuate. Inoltre non sono presenti importanti ritardi. Si nota però come durante i movimenti veloci ci siano delle amplificazioni. Questo perché è stata aumentata la frequenza di risonanza del filtro di Kalman: così facendo il fenomeno di risonanza si è verificato per movimenti più veloci, ovvero a frequenza più elevate.

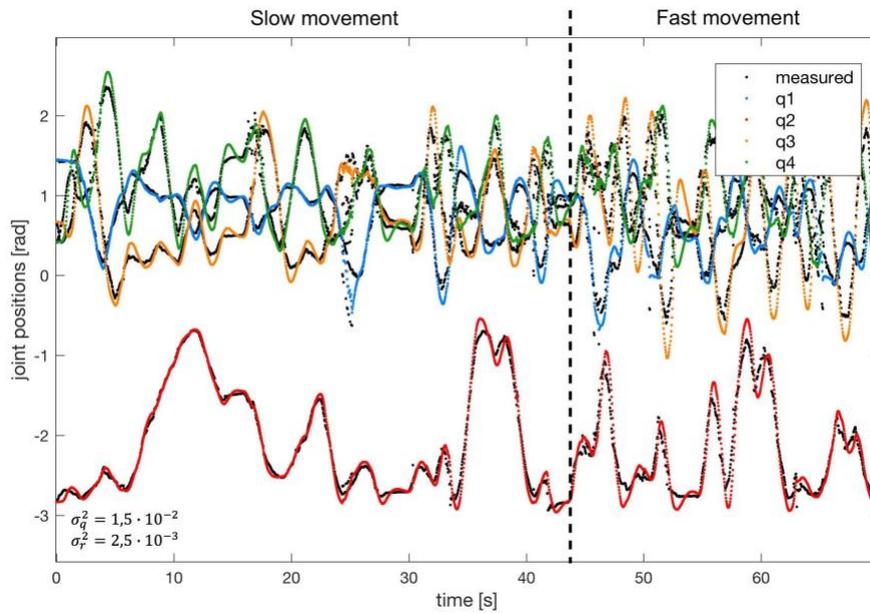


Figura 4.28 Stima delle q_i con KF, $\sigma_q^2 = 1,5 \cdot 10^{-2}$, $\sigma_r^2 = 2,5 \cdot 10^{-3}$, prova 2

Si vuole accrescere ulteriormente la frequenza di risonanza del filtro di Kalman. Si può aumentare ancora σ_q^2 oppure diminuire σ_r^2 . Si propongono entrambe le soluzioni. In Figura 4.29 viene incrementato σ_q^2 e impostato $\sigma_q^2 = 1,5 \cdot 10^0$ con $\sigma_r^2 = 2,5 \cdot 10^{-3}$. In Figura 4.30 viene lasciato $\sigma_q^2 = 1,5 \cdot 10^{-2}$ e diminuito $\sigma_r^2 = 2,5 \cdot 10^{-4}$.

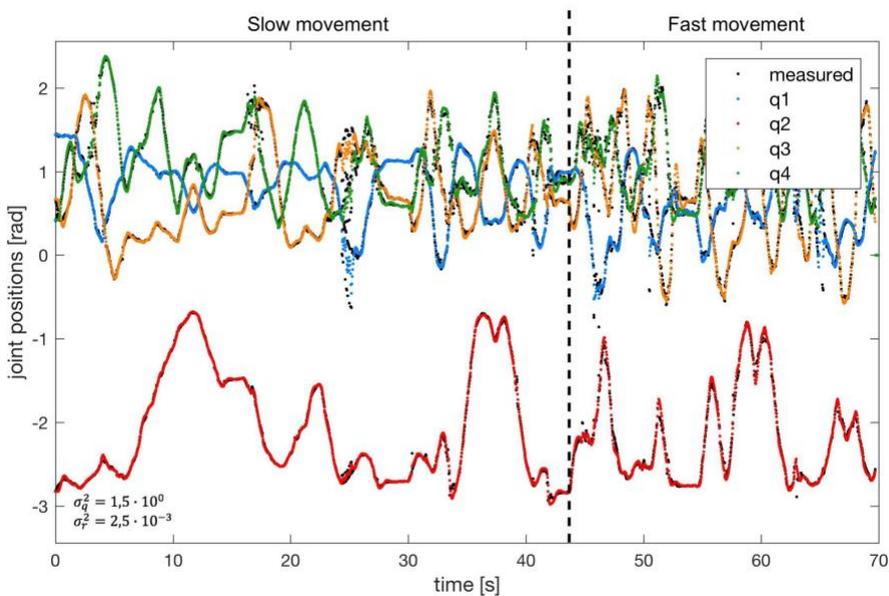


Figura 4.29 Stima delle q_i con KF, $\sigma_q^2 = 1,5 \cdot 10^0$, $\sigma_r^2 = 2,5 \cdot 10^{-3}$, prova 2

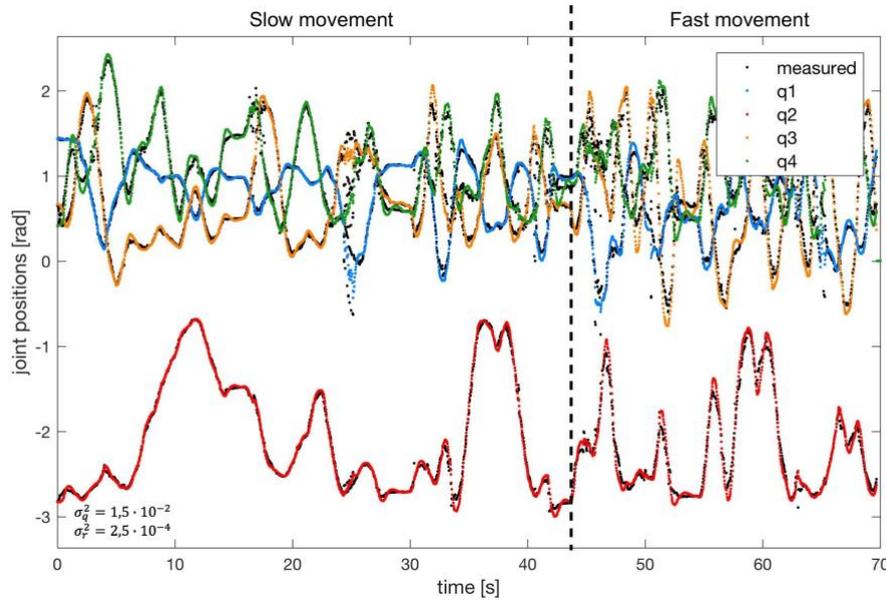


Figura 4.30 Stima delle q_i con KF, $\sigma_q^2 = 1,5 \cdot 10^{-2}$, $\sigma_r^2 = 2,5 \cdot 10^{-4}$, prova 2

Si è così eliminata ogni risonanza e ritardo in entrambi i casi. Lo spostamento della frequenza di risonanza a valori troppo alti non risulta però particolarmente vantaggioso. L'effetto filtrante in questi casi è minimo e rende vano l'utilizzo del KF. È più ragionevole accontentarsi di piccoli ritardi e leggere amplificazioni, ma ottenere dati più puliti da rumore. Nei casi visti il filtro di Kalman più efficace risulta quello con $\sigma_q^2 = 1,5 \cdot 10^{-2}$, $\sigma_r^2 = 2,5 \cdot 10^{-3}$. Il filtraggio sui movimenti lenti è adeguato e ci si può accontentare di amplificazioni su movimenti più veloci. In un'ottica di collaborazione uomo-robot infatti, è plausibile pensare che l'uomo abbia movimenti non troppo rapidi. I parametri σ_q^2 e σ_r^2 devono essere dunque valutati a seconda dell'applicazione. Se l'obiettivo è una *collision avoidance* la risposta deve essere quanto più rapida, se è un *hand following* è bene preferire una maggiore azione filtrante per individuare con maggiore precisione e minor rumore la posizione della mano. Come possibile sviluppo è possibile pensare di poter adottare dei valori σ_q^2 e σ_r^2 adattativi a seconda della dinamica del sistema osservato.

Nelle analisi seguenti si adotteranno i valori $\sigma_q^2 = 1,5 \cdot 10^{-3}$ e $\sigma_r^2 = 2,5 \cdot 10^{-5}$, ritenuti un buon compromesso tra basso ritardo e azione filtrante, per applicazioni di *hand following*.

4.5 Effetto inserimento filtro passa-basso

Si vuole inserire un filtro passa-basso da applicare alle posizioni di gomito $p_{e,1}^0$ e polso $p_{e,2}^0$ forniti dal sensore Kinect, a monte del filtro di Kalman. Si effettua una prima analisi utilizzando un filtro non real-time, a fase nulla. Questo permette di valutare il possibile effetto migliorativo dell'introduzione di un filtro real-time a valle dell'acquisizione dei dati di Kinect. Si utilizza dunque, un filtro ideale, privo di ritardi, analizzando così il caso più conservativo. La funzione utilizzata per la creazione del filtro a fase nulla è la funzione “*filtfilt*” di Matlab. Per poter utilizzare la funzione è necessario prima definire i parametri caratteristici del filtro IIR di partenza, specificati in Tabella 4.3.

Tabella 4.3 Definizione dei parametri del filtro passa basso a fase nulla

PassbandFrequency	3 Hz
StopbandFrequency	6 Hz
StopbandAttenuation	10 dB
DesignMethod	equiripple

Si analizzano gli effetti dell'introduzione del filtro passa basso nello spazio giunti, a seguito di cinematica inversa. Si confrontano i valori delle variabili di giunto relativi ai dati grezzi misurati da Kinect non filtrati, alla stima prodotta da KF a partire dai dati grezzi non filtrati, ai dati grezzi di Kinect filtrati e alla stima prodotta da KF a partire dai dati grezzi filtrati. In Figura 4.31 si mostra l'analisi relativa a una ridotta finestra temporale della variabile di giunto q_4 relativa alla prova 2. La finestra temporale mostrata è scelta per la particolare instabilità riscontrata nei dati grezzi prodotti da Kinect, la quale permette di valutare con maggiore efficacia gli effetti prodotti dal filtro passa basso su un segnale più rumoroso. Si nota che applicare un filtro passa basso a monte del filtro di Kalman rende il segnale di q_4 più stabile. Tuttavia è doveroso precisare che in altre finestre temporali la differenza risulta non essere particolarmente rilevante.

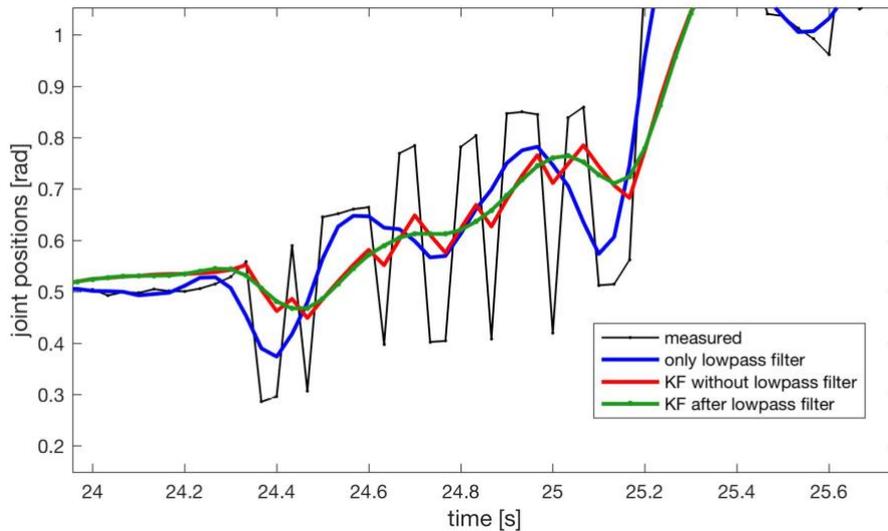


Figura 4.31 Effetto del filtro passa basso applicato a gomito e polso su q_4 nello spazio giunti, con e senza KF, prova 2

Poiché il segnale di interesse è quello nello spazio operativo della posizione del gomito $\hat{\mathbf{p}}_{e,1}^0$ e del polso $\hat{\mathbf{p}}_{e,2}^0$, si analizzano tali segnali nella stessa finestra temporale che aveva mostrato essere particolarmente affetta da rumore. Si mostra l'analisi effettuata sul polso $\hat{\mathbf{p}}_{e,2}^0$ in quanto è il segnale soggetto a maggiore instabilità. Dai dati precedentemente elaborati, viene effettuata un'operazione di cinematica diretta con imposizione di lunghezze di braccio L_1 e avambraccio L_2 fisse.

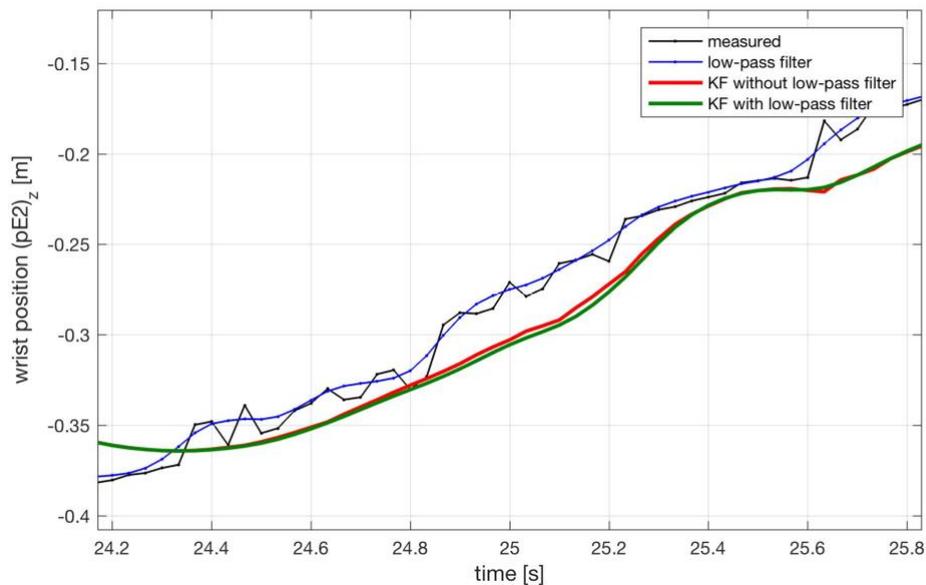


Figura 4.32 Effetto del filtro passa basso applicato a gomito e polso sulla stima della posizione polso $\hat{\mathbf{p}}_{e,2}^0$ nello spazio operativo con imposizione delle lunghezze fisse L , con e senza KF, prova 2

In Figura 4.32 si mostra l'andamento nel tempo della coordinata z della posizione del polso $\hat{\mathbf{p}}_{e,2}^O$. Ci si concentra sul confronto fra le stime effettuate dal KF utilizzando o meno i dati precedentemente filtrati dal filtro passa basso. Si nota come la differenza che si mostrava nello spazio giunti, non risulta rilevante nello spazio operativo.

L'imposizione delle lunghezze L fisse nell'operazione di cinematica diretta evidentemente svolge un'azione filtrante (paragrafo 4.6). Tale azione filtrante rende vana l'introduzione di un filtro passa basso sulle posizioni di gomito $\mathbf{p}_{e,1}^O$ e polso $\mathbf{p}_{e,2}^O$ forniti dal sensore Kinect, a monte del filtro di Kalman.

Poiché l'analisi effettuata viene prodotta mediante ricorso a un filtro passa basso a fase nulla, l'introduzione di un filtro passa basso real-time risulterebbe dannosa, in quanto non produrrebbe una migliore azione filtrante, ma introdurrebbe del ritardo nel segnale.

Si sconsiglia l'utilizzo di un filtro passa basso a monte su gomito $\mathbf{p}_{e,1}^O$ e polso $\mathbf{p}_{e,2}^O$, ma è da considerare l'introduzione di un filtro passa basso unicamente sul giunto spalla $\mathbf{p}_{e,0}^W$ in modo da poter ottenere, nel sistema di riferimento World W , dei segnali $\mathbf{p}_{e,1}^W$ e $\mathbf{p}_{e,2}^W$ più stabili. La questione viene approfondita nel paragrafo 4.7.

Ci si potrebbe chiedere se non sia il caso di rinunciare al filtro di Kalman e utilizzare unicamente un filtro passa basso e l'imposizione delle lunghezze L fisse. Come già discusso nel paragrafo 1.1.1, la Kinect presenta delle inaccurately, dipendenti per esempio dalla distanza del soggetto dal sensore, che possono originare anche a fenomeni di deriva della posizione rilevata, i quali verranno evidenziati nel paragrafo 4.9.2. Anche lo stesso SKD, nella fase di estrapolazione dello scheletro, introduce una serie di stime non esatte. Il filtro di Kalman produce una stima, basata su un modello, che migliora notevolmente l'accuratezza della posizione finale.

4.6 Analisi sull'imposizione delle lunghezze fisse

Le prove eseguite nel paragrafo 4.5 hanno fornito importanti spunti per una più attenta analisi degli effetti prodotti dall'imposizione delle lunghezze L fisse in fase di cinematica diretta e dagli effetti del filtro di Kalman. In Figura 4.32 si era ottenuta una stima attraverso filtro passa basso a fase nulla e imposizione di lunghezze fisse priva di particolare rumore. In assenza di altre indicazioni, non si può dire nulla su quale delle stime prodotte sia la migliore. Di seguito si propone però un'analisi che vuole mettere chiarezza sugli effetti prodotti dai vari elementi inseriti nell'algoritmo. Si vogliono confrontare due situazioni esplicitate in Figura 4.33, in cui in una è presente l'imposizione di lunghezze L fisse e nell'altra vi è anche il filtro di Kalman. In entrambe è stato applicato un filtro passa basso a fase nulla a monte, con il solo scopo di ripulire maggiormente i dati, permettendo un confronto più diretto fra i due algoritmi per gli aspetti che si vogliono analizzare. Si era visto come la presenza di un filtro a monte non influisse significativamente sulla stima prodotta dall'algoritmo con il filtro di Kalman, mentre può aiutare quella prodotta dall'algoritmo con solo l'imposizione delle lunghezze fisse alla luce anche di analisi effettuate in precedenza in Figura 4.7.



Figura 4.33 Schema sul confronto eseguito fra effetto della sola imposizione delle lunghezze fisse e dell'effetto del filtro di Kalman con imposizione delle lunghezze fisse

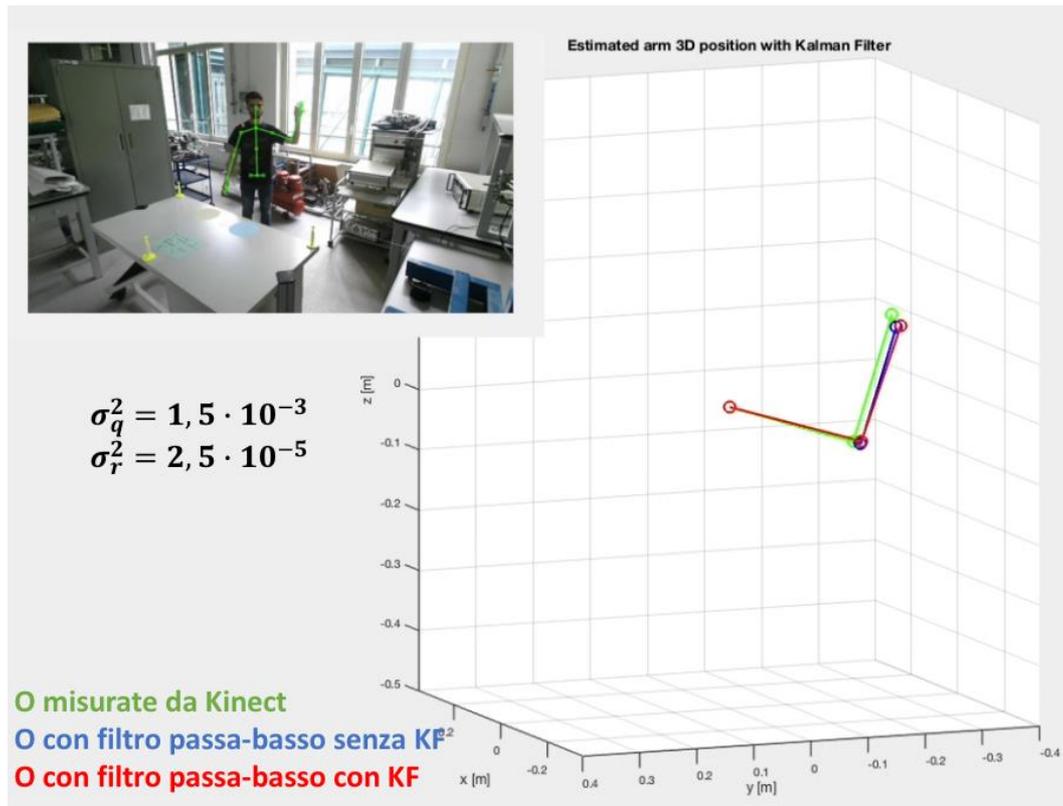


Figura 4.34 Confronto qualitativo eseguito mediante riproduzione filmato fra misura grezza di Kinect, stima con sola imposizione delle lunghezze fisse e stima con filtro di Kalman con imposizione delle lunghezze fisse

È eseguita una riproduzione video dei giunti rappresentanti i giunti spalla $\mathbf{p}_{e,0}^O$, gomito $\mathbf{p}_{e,1}^O$ e polso $\mathbf{p}_{e,2}^O$. In parallelo si riproduce il video RGB prodotto da Kinect. In Figura 4.34 ne viene mostrato un frame. La riproduzione video contiene sia le informazioni sulle posizioni grezze ottenute da Kinect che la stima prodotta dai due algoritmi alternativi, entrambi con imposizione delle lunghezze L fisse, con e senza KF. Si sceglie questo tipo di approccio in quanto, seppure qualitativo, permette di ottenere delle indicazioni su quanto la stima prodotta migliori il segnale di partenza. Riproducendo il video RGB in parallelo, è possibile valutare i momenti in cui sono presenti evidenti incongruenze fra il dato Kinect e il movimento reale, allo stesso tempo è possibile valutare se la stima prodotta migliori o meno il risultato finale.

Si ricordi che l'azione filtrante del KF è regolabile mediante i parametri di covarianza che permettono di ottenere movimenti più o meno fluidi con più o meno ritardo. Si eseguono più prove con differenti valori di covarianza per apprezzare maggiormente questo aspetto con video 3D.

Dall'analisi delle diverse prove emerge l'effetto sostanziale dell'imposizione delle lunghezze braccio e avambraccio: il filtraggio ottenuto da esse opera lungo l'asse dei collegamenti (*link*), mantenendo del rumore nelle altre direzioni.

Si mostra a titolo qualitativo e di esempio la Figura 4.35 [10] che rappresenta quello che potremmo definire "ellissoide di covarianza". Analogamente all'ellissoide di manipolabilità, rappresenterebbe la varianza della stima della posizione del polso a seguito dell'imposizione delle sole lunghezze L fisse, in diverse configurazioni del braccio.

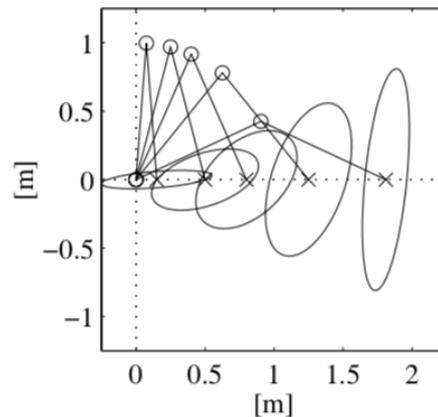


Figura 4.35 Ellissoide di covarianza

La Figura 4.36 chiarisce questo concetto. Si evidenzia come la stima prodotta abbia maggiore accuratezza lungo l'asse del link per merito dell'imposizione delle lunghezze L fisse.

Come è noto, il filtro di Kalman implementato lavora sulle variabili di giunto q_i . Come è possibile apprezzare dai video prodotti, esso produce un'azione filtrante proprio su tali variabili. In Figura 4.37 si mostrano gli effetti prodotti dall'algoritmo completo. Il filtro di Kalman ottiene una miglior stima per quanto riguarda gli angoli fra i link, in direzione quindi perpendicolare rispetto l'asse dei link. Successivamente l'imposizione delle lunghezze fisse migliora l'accuratezza della stima anche lungo la direzione dei link.

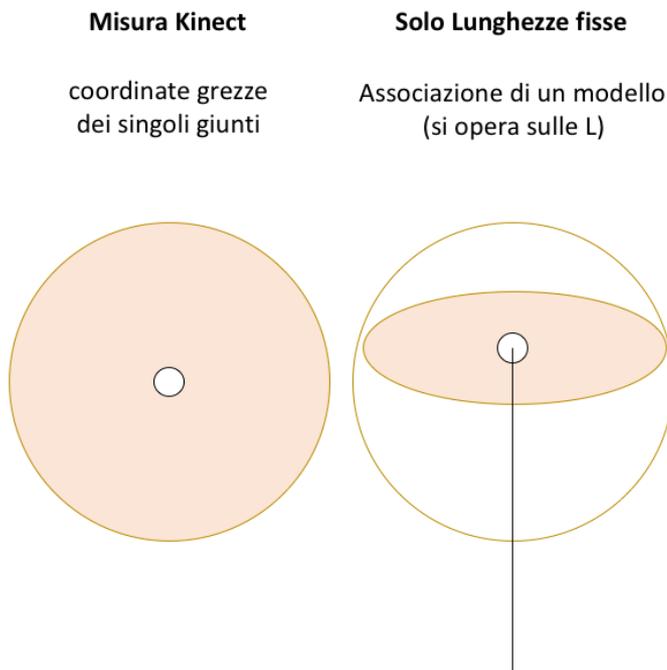


Figura 4.36 Ellissoide di covarianza del polso con la sola imposizione delle L fisse

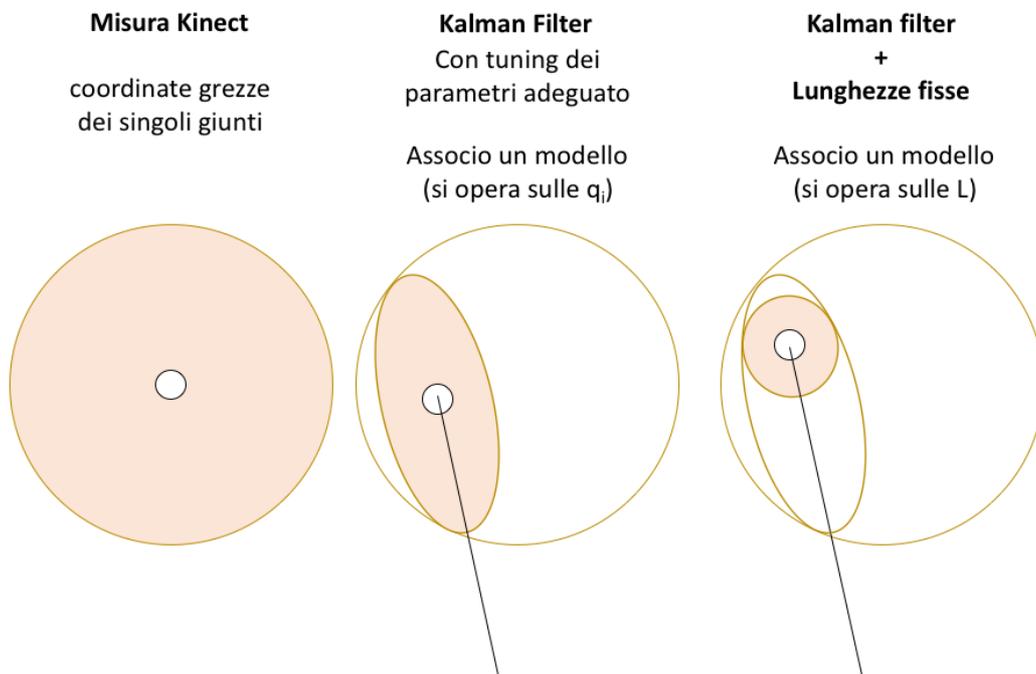


Figura 4.37 Ellissoide di covarianza del polso a seguito di KF e imposizione delle L fisse

4.6.1 Problematica della taratura delle lunghezze fisse

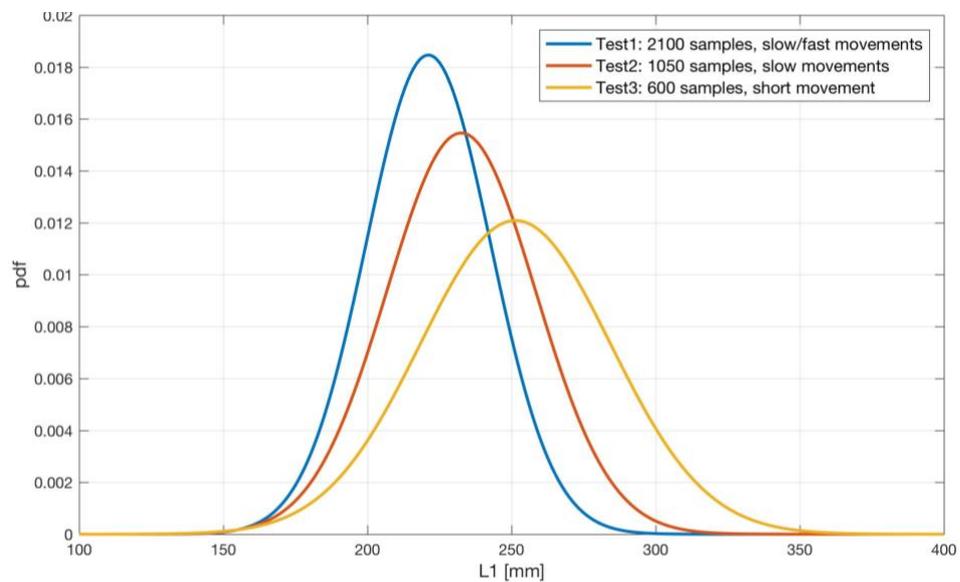
Per semplicità nelle prove eseguite si sono adottati i valori medi di L_1 e L_2 relativi alle prove specifiche. È chiaro che, visto che l'operatore non è cambiato da una prova all'altra, il valore da adottare per L_1 e L_2 deve essere teoricamente lo stesso. Si è rilevato invece che la media delle lunghezze nelle diverse prove cambia, producendo discrepanze di qualche decina di millimetri. Si effettua un'analisi più approfondita su media e deviazione standard sulle prove 1, 2 e 3. I risultati sono riassunti in Tabella 4.4 e riprodotti graficamente in Figura 4.38.

Tabella 4.4 Analisi lunghezze medie nelle diverse prove

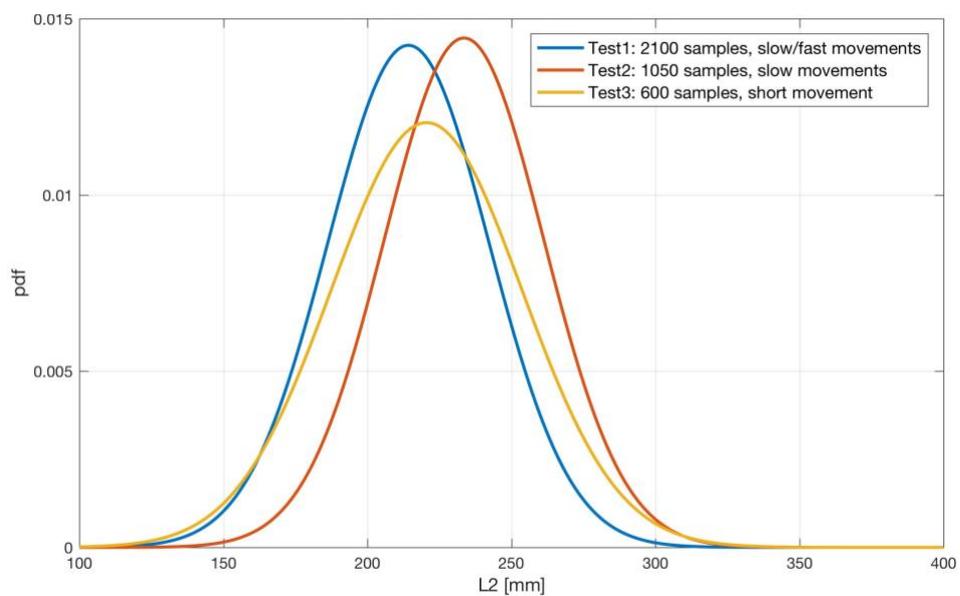
	samples	L_1		L_2	
		Media [m]	Dev.St. [m]	Media [m]	Dev.St. [m]
Prova 1	2100	0,2211	0,0216	0,2142	0,0280
Prova 2	1050	0,2325	0,0258	0,2335	0,0276
Prova 3	600	0,2513	0,0330	0,2205	0,0331

In un'applicazione real-time, inoltre, questo tipo di approccio non è utilizzabile. Sarebbe possibile solo una stima mediante media mobile che non garantisce particolare accuratezza per via della natura del sensore.

È consigliabile effettuare un'operazione di taratura specifica per ogni operatore che andrà ad utilizzare il sistema di rilievo del movimento. L'operazione di taratura deve consistere in una prova nella quale vengono eseguiti movimenti ripetuti e lenti a favore del sensore Kinect. Si devono eseguire, quindi, movimenti che siano frontali al sensore, mantenendo una distanza costante dello stesso. Si deve inoltre mantenere il soggetto staccato dallo sfondo. Questa procedura permette di ottenere dei dati caratterizzati da maggiore accuratezza. Dalla prova o dalle prove di taratura effettuate vengono poi prelevati i valori di L_1 e L_2 medi, considerati come la miglior stima possibile. Tali valori verranno utilizzate in tutte le applicazioni successive dall'operatore.



(a)



(b)

Figura 4.38 Rappresentazione secondo distribuzione gaussiana, delle lunghezze di braccio L_1 (a) e avambraccio L_2 (b) nelle prove eseguite

4.7 Filtraggio della posizione del giunto spalla

Fino a questo momento le analisi sono state effettuate sulle posizioni di gomito $\mathbf{p}_{e,1}^O$ e polso $\mathbf{p}_{e,2}^O$ espresse nel sistema di riferimento spalla O . Si vuole adesso esprimere tutte le posizioni nello spazio operativo nel sistema di riferimento world W . In un primo momento si sommano vettorialmente le posizioni di gomito $\hat{\mathbf{p}}_{e,1}^O$ e polso $\hat{\mathbf{p}}_{e,2}^O$ con quelle della spalla $\mathbf{p}_{e,0}^W$ considerando una rotazione rigida fra i due sistemi di riferimento, secondo l'Eq. (4.10). Le prove sono state eseguite in modo che l'orientazione del busto dell'operatore non cambi significativamente, legittimando questa scelta. In questa fase ci si pone dunque l'obiettivo di sommare alle stime prodotte un vettore $\mathbf{p}_{e,0}^W$ che non sia affetto da rumore come quello registrato direttamente da Kinect. L'algoritmo elaborato permette una miglior stima delle posizioni di gomito $\hat{\mathbf{p}}_{e,1}^O$ e polso $\hat{\mathbf{p}}_{e,2}^O$, ma se queste vengono poi sommate con un segnale rumoroso e con bruschi salti, si vanifica tutta la procedura di stima. Si otterrebbero posizioni di gomito $\hat{\mathbf{p}}_{e,1}^W$ e polso $\hat{\mathbf{p}}_{e,2}^W$ non molto migliori di quelle misurate da Kinect.

Si rende necessario l'utilizzo di un filtro passa basso in grado di funzionare in real-time, da applicare alla posizione del giunto spalla $\mathbf{p}_{e,0}^W$. Si decide di inserire un filtro IIR passa basso con un polo del tipo:

$$y_k = bx_k + bx_{k-1} + ay_{k-1} \quad (4.18)$$

dove

$$\begin{cases} a = (1 - \alpha)/(1 + \alpha) \\ b = (1 - a)/2 \\ \alpha = \tan(\pi \cdot f_{cut}/f_s) \end{cases} \quad (4.19)$$

con x_k il segnale non filtrato e y_k il segnale filtrato all'istante k , f_{cut} la frequenza di taglio e $f_s = 30$ Hz la frequenza di campionamento del sensore Kinect.

Un filtro di questo tipo è applicabile anche alle posizioni di gomito $\mathbf{p}_{e,1}^W$ e polso $\mathbf{p}_{e,2}^W$ a monte, ma, come descritto precedentemente, tale filtraggio non produrrebbe risultati migliorativi. In Tabella 4.5 si consigliano dei valori che, a seguito di

sperimentazione, sono stati valutati ottimali per le diverse tipologie di giunto. I giunti gomito e polso sono caratterizzati da dinamiche più elevate rispetto alla spalla, per questo risulterebbe opportuno tagliare a frequenze più elevate.

Tabella 4.5 frequenze di cut-off consigliate per filtraggio dei giunti

Giunto	f_{cut} [Hz]
Spalla	0,1 ÷ 1
Gomito e polso	5 ÷ 6

Viene filtrata dunque la posizione del giunto spalla $\mathbf{p}_{e,0}^W$ istante per istante, con una $f_{cut} = 0,2$ Hz. In Figura 4.39 si riportano i risultati ottenuti, relativi alla prova 3. Si riporta l'andamento nel tempo delle tre coordinate di $\mathbf{p}_{e,0}^W$.

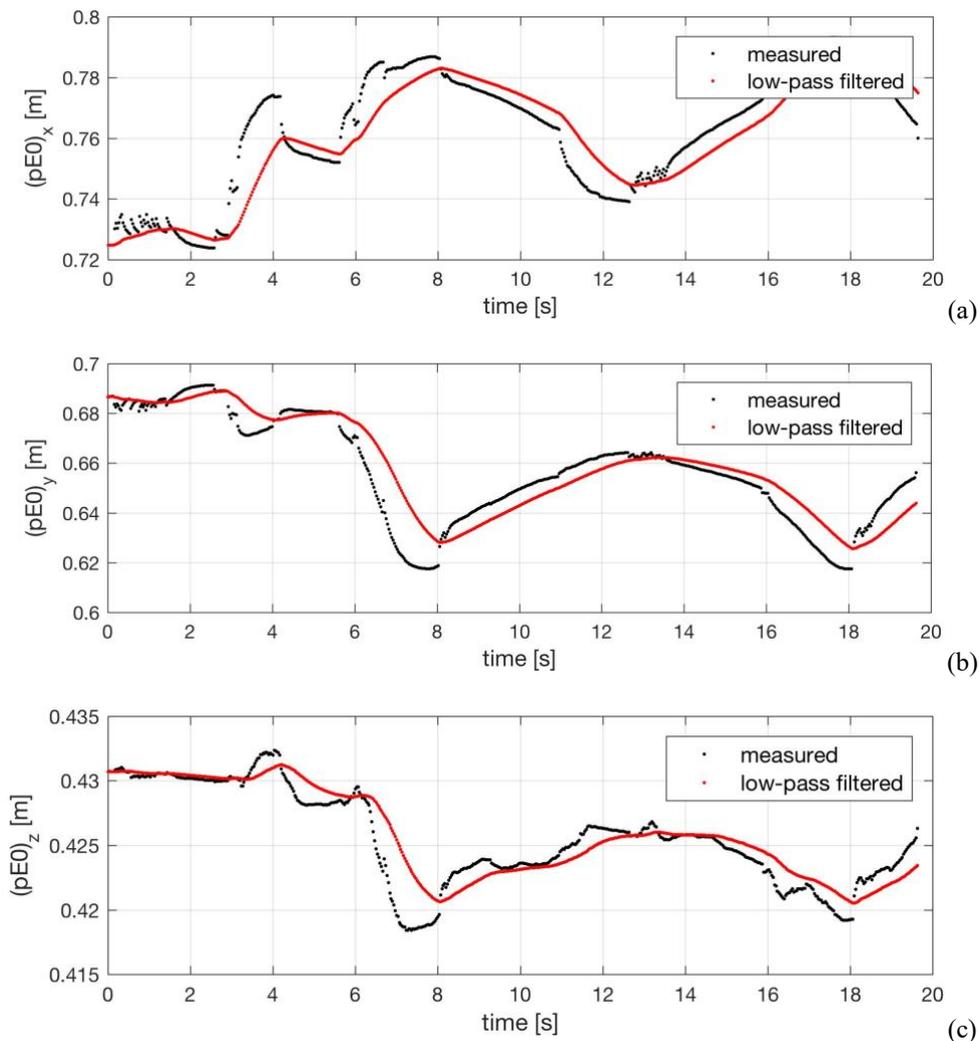


Figura 4.39 Coordinate x (a), y (b) e z (c) della posizione spalla $\mathbf{p}_{e,0}^W$ misurata e filtrata, prova 3

Il filtraggio così ottenuto risulta adeguato per la nostra applicazione. L'utilizzo di una frequenza di taglio f_{cut} così bassa permette di ottenere un segnale abbastanza pulito e privo dei bruschi salti che caratterizzano il segnale grezzo di Kinect. D'altra parte un filtraggio così spinto produce ritardi anche di 1 s. Le escursioni del giunto spalla risultano tuttavia contenute, intorno ai 10 mm. In applicazioni di collaborazione uomo-robot non sono previste brusche variazioni della posizione della spalla, che tende a rimanere stabile o spostarsi di poco. Inoltre la variazione registrata risulta trascurabile rispetto possibili errori derivanti nella stima della posizione di gomito e polso. Per queste ragioni il ritardo ottenuto risulta accettabile. La priorità, nel caso del giunto spalla, deve risiedere nella stabilità e fluidità del segnale, il quale non deve presentare salti.

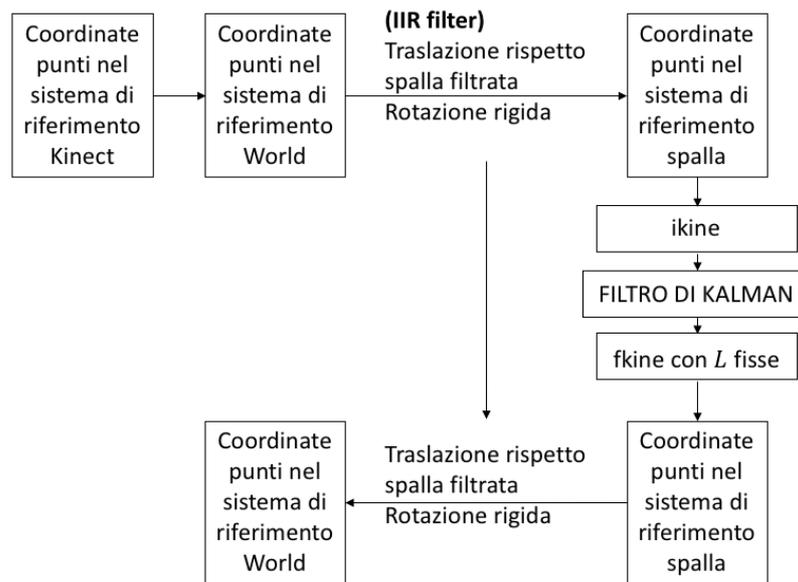


Figura 4.40 Schema dell'algoritmo con IIR filter su spalla e rotazione rigida $O - W$

In Figura 4.40 si riassumono le fasi principali di cui si compone l'algoritmo descritto nel presente paragrafo, con una stima delle posizioni dei giunti nel sistema di riferimento W .

4.8 Sistema di riferimento spalla rotante real-time

Si vuole completare il modello implementando la rotazione del sistema di riferimento spalla O istante per istante. In pratica si vuole permettere la rotazione del busto dell'operatore liberamente in ogni direzione.

Per farlo è necessario definire una procedura che permetta di ottenere una terna levogira mobile, ovvero capace di eseguire rototraslazioni in real-time. Nel paragrafo 4.1.1 era stato presentato un metodo capace di ottenere una terna levogira a partire da tre punti sperimentali, non perfettamente ortogonali fra loro. Si vuole utilizzare un metodo simile adattandolo nel caso di interesse.

Si rende necessaria l'individuazione di tre punti sullo scheletro che possano essere presi come riferimento per la definizione della terna levogira. La scelta ricade sul giunto spalla, centro del sistema di riferimento spalla, sul giunto "Spine Shoulder" $\mathbf{p}_{e,ss}$ e del giunto "Spine Mid" $\mathbf{p}_{e,sm}$. In Tabella 4.6 sono esplicitati gli indici relativi ai giunti scelti.

Tabella 4.6 Indici dei giunti necessari per individuazione terna mobile del sistema di rif. O

Joint	Index
ShoulderLeft	5
ShoulderRight	9
SpineShoulder	21
SpineMid	2

I due giunti aggiuntivi presi, sono disposti centralmente allo scheletro e consentono di essere facilmente utilizzabili sia nel caso di tracciamento del braccio destro che sinistro. Viene scelto il punto 21 in quanto più rigido e indipendente rispetto il punto della spalla opposta a quella analizzata. Il punto 2 è scelto in quanto interessato agli spostamenti della parte superiore del busto, escludendo il collo, il quale potrebbe avere spostamenti indipendenti. I tre punti scelti individuano dunque un piano che ben rappresenta le rotazioni che può avere il busto umano.

Dall'Eq. (4.9), si ottengono le coordinate di tali punti nel sistema di riferimento W . Si definiscono i punti $\mathbf{p}_{A_k}^W$, $\mathbf{p}_{B_k}^W$ e $\mathbf{p}_{C_k}^W$ per ogni istante k , con $\mathbf{p}_{A_k}^W = \mathbf{p}_{e,0_k}^W$, $\mathbf{p}_{B_k}^W = \mathbf{p}_{e,ss_k}^W$ e $\mathbf{p}_{C_k}^W = \mathbf{p}_{e,sm_k}^W$. La Figura 4.41 mostra la posizione dei punti e la terna levogira che si ha intenzione di ottenere.

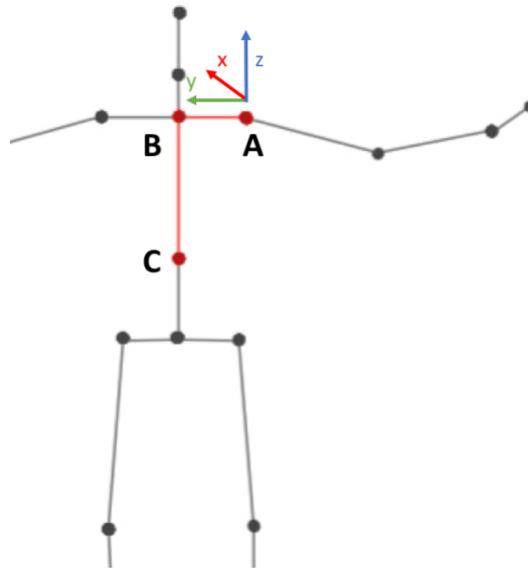


Figura 4.41 Punti necessari per individuazione terna mobile del sistema di rif. O

Precedentemente si era adottato un filtro IIR real-time su $\mathbf{p}_{e,0}^W$. Si applica la stessa tipologia di filtro con lo stesso valore di frequenza di taglio $f_{cut} = 0,2$ Hz anche sui punti $\mathbf{p}_{e,ss}^W$ e $\mathbf{p}_{e,sm}^W$. Tali punti presentano infatti dinamiche simili a quelle della spalla: risulta prioritaria un'azione filtrante che consenta di evitare eccessivo rumore e salti improvvisi che si riflettano poi sulla rotazione del sistema di riferimento O .

La procedura utilizzata viene reiterata ad ogni istante k . Per il tracciamento dei movimenti della spalla sinistra vengono introdotti i seguenti vettori ausiliari:

$$\begin{cases} \mathbf{v}_{A_k} = \mathbf{p}_{B_k}^W - \mathbf{p}_{A_k}^W \\ \mathbf{v}_{B_k} = \mathbf{p}_{C_k}^W - \mathbf{p}_{B_k}^W \\ \mathbf{v}_{C_k} = \mathbf{v}_{A_k} \times \mathbf{v}_{B_k} \end{cases} \quad (4.20)$$

Se ad essere tracciata è la spalla destra risulterà \mathbf{v}_{A_k} con segno inverso. Tale soluzione viene già implementata nell'algoritmo.

È possibile quindi calcolare i versori della terna levogira del sistema di riferimento O nel sistema di riferimento W per ogni istante k , mediante un ulteriore vettore ausiliario:

$$\hat{\mathbf{x}}_k = \frac{\mathbf{v}_{C_k}}{\|\mathbf{v}_{C_k}\|} \quad (4.21)$$

$$\hat{\mathbf{y}}_k = \frac{\mathbf{v}_{A_k}}{\|\mathbf{v}_{A_k}\|} \quad (4.22)$$

$$\mathbf{v}_{D_k} = \hat{\mathbf{x}}_k \times \hat{\mathbf{y}}_k \quad (4.23)$$

$$\hat{\mathbf{z}}_k = \frac{\mathbf{v}_{D_k}}{\|\mathbf{v}_{D_k}\|} \quad (4.24)$$

Si ottiene infine la matrice omogenea di rototraslazione $\mathbf{A}_{O_k}^W$ del sistema di riferimento spalla O rispetto a quello World W per ogni istante temporale k :

$$\mathbf{A}_{O_k}^W = \begin{bmatrix} \hat{\mathbf{x}}_k & \hat{\mathbf{y}}_k & \hat{\mathbf{z}}_k & \mathbf{p}_{e,0k}^W \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.25)$$

L'Eq. (4.25) differisce dall'Eq. (4.10) dal termine di rotazione, questa volta variabile istante per istante. Attraverso l'Eq. (4.12) è possibile quindi ottenere le posizioni dei giunti spalla $\mathbf{p}_{e,1}^O$ e polso $\mathbf{p}_{e,2}^O$ nel sistema di riferimento O . Viceversa, attraverso l'Eq. (4.13) è possibile tornare al sistema di riferimento W a valle della stima eseguita con filtro di Kalman e imposizione delle lunghezze fisse, ottenendo le posizioni $\hat{\mathbf{p}}_{e,1}^W$ e $\hat{\mathbf{p}}_{e,2}^W$. In Figura 4.42 si presenta la procedura seguita dall'algoritmo.

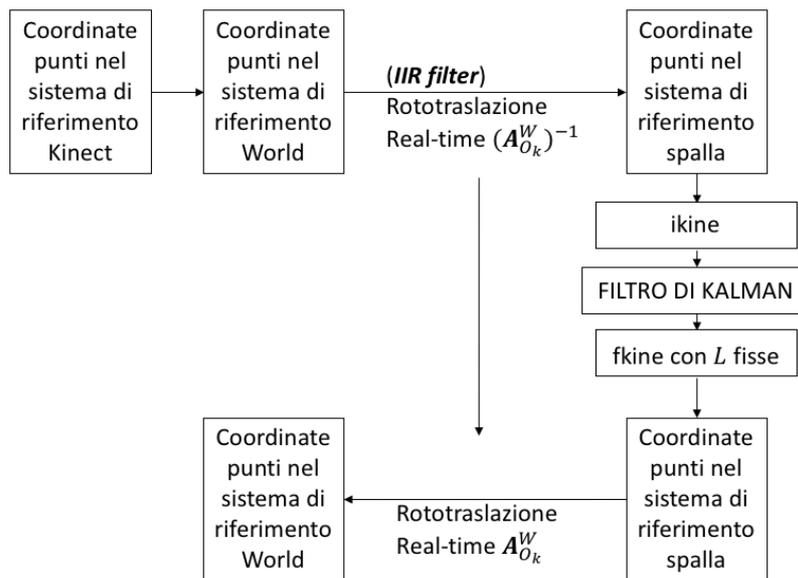


Figura 4.42 Schema dell'algoritmo con rototraslazione real-time del sistema di riferimento O

4.9 Confronto con dati Optitrack

Inizialmente, l'utilizzo di dati simulati permetteva di ottenere un confronto fra la stima prodotta dal modello con i dati veri del sistema osservato. Successivamente la bontà del modello è stata giudicata attraverso analisi qualitative, per esempio mediante confronto fra il segnale grezzo di Kinect e quello stimato. Questo perché gli unici dati a disposizione erano quelli del sensore Kinect e non vi era modo di confrontarli con dati più attendibili. Si rende necessario dunque ottenere dati più accurati di quelli forniti da Kinect per permettere un confronto tra questi e quelli stimati. È possibile perseguire l'obiettivo utilizzando un sensore di profondità con performance migliori, come Optitrack Trio. Nel paragrafo 1.1 si era accennato come l'accuratezza di tale sensore fosse notevolmente superiore a quella di Kinect.

Il sensore Optitrack funziona mediante applicazione di particolari marker sul corpo dell'operatore. In output fornisce la posizione di tali marker nel proprio sistema di riferimento, che verrà denominato “*Opt*”.

Viene eseguita la prova 3. Tale prova consiste nella riproduzione prima di un movimento casuale, poi di un movimento rettilineo. Del movimento rettilineo si effettuano due corse a velocità moderata e una mezza corsa a velocità più elevata. L'operatore effettua il movimento rettilineo con l'ausilio di un filo da seguire.

La prova viene acquisita sia da Kinect che da Optitrack. Per poter sincronizzare in post-produzione le due acquisizioni, viene utilizzato, come elemento di trigger, un flash.

Vengono applicati due marker in corrispondenza di polso e spalla per poterne registrare le posizioni. Lo strumento fornisce dunque la posizione dei marker “*M*” corrispondenti ai giunti spalla $\mathbf{p}_{e,0M}^{Opt}$ e polso $\mathbf{p}_{e,2M}^{Opt}$ nel sistema di riferimento Optitrack.

In Figura 4.43 viene mostrato un frame, acquisito dalla fotocamera RGB di Kinect, relativo alla prova 3. Si evidenzia la collocazione dei marker.

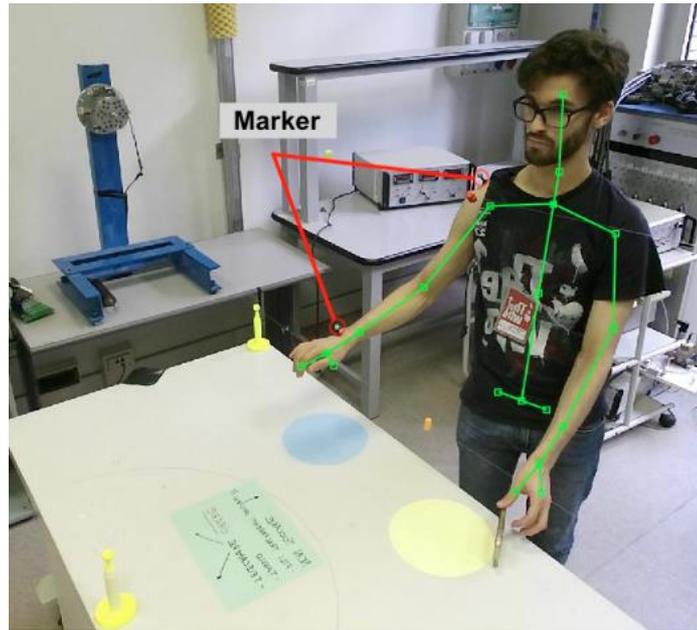


Figura 4.43 Frame relativo alla prova 3 attraverso fotocamera RGB di Kinect

4.9.1 Elaborazione dati Optitrack

Per poter ottenere le posizioni nel sistema di riferimento World, è necessario identificarlo mediante la stessa procedura utilizzata per Kinect (paragrafo 4.1.1). Si collocano dei marker di Optitrack immediatamente sopra i marker gialli utilizzati per identificare il sistema di riferimento World con i dati Kinect. Si registrano la posizione di tali marker, denominati \mathbf{p}_A^{opt} , \mathbf{p}_B^{opt} , \mathbf{p}_C^{opt} . Si introducono dei vettori ausiliari:

$$\begin{cases} \mathbf{v}_A = \mathbf{p}_B^{opt} - \mathbf{p}_A^{opt} \\ \mathbf{v}_B = \mathbf{p}_C^{opt} - \mathbf{p}_A^{opt} \\ \mathbf{v}_C = \mathbf{v}_A \times \mathbf{v}_B \end{cases} \quad (4.26)$$

Attraverso le Eq. da (4.4) a (4.7), è possibile calcolare i versori della terna levogira del sistema di riferimento W nel sistema di riferimento Opt . Si ricava quindi la matrice omogenea di rototraslazione \mathbf{A}_W^{opt} del sistema di riferimento World rispetto a quello Optitrack:

$$\mathbf{A}_W^{Opt} = \begin{bmatrix} \hat{x} & \hat{y} & \hat{z} & \mathbf{p}_A^{Opt} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.27)$$

La determinazione della matrice di rototraslazione \mathbf{A}_W^{Opt} fissa, permette di calcolare le coordinate dei marker in corrispondenza dei giunti, nel sistema di riferimento W . Si ha, per ogni istante k :

$$\tilde{\mathbf{p}}_{e,iM_k}^W = \mathbf{A}_K^{Opt} \tilde{\mathbf{p}}_{e,iM_k}^{Opt} \quad (4.28)$$

dove $\mathbf{A}_{Opt}^W = (\mathbf{A}_W^{Opt})^{-1}$.

Si ottengono quindi, per la prova 3, le posizioni di spalla $\mathbf{p}_{e,0M}^W$ e polso $\mathbf{p}_{e,2M}^W$ nel sistema di riferimento World. È possibile quindi un confronto fra i dati registrati mediante il sensore Optitrack e quelli stimati dall'algoritmo.

4.9.2 Analisi della traiettoria del giunto polso

A causa della precaria collocazione del marker sulla spalla e dalla posizione diversa rispetto a quella prodotta dall'SDK di Kinect, i dati forniti da Optitrack sulla spalla non sono molto significativi. Ciò non permette un confronto consistente con il filtraggio ottenuto sulla spalla.

Ci si concentra sull'analisi del solo giunto polso. In Figura 4.44 viene mostrata la traiettoria rettilinea eseguita, proiettata sul piano x - y .

Si evince come esiste un off-set tra i dati forniti da Kinect e quelli di Optitrack a causa dell'ingombro del marker. Tuttavia è possibile effettuare un'analisi qualitativa sulla stima ottenuta.

La traiettoria misurata da Kinect presenta una deriva della direzione, evidente se confrontata con quella di Optitrack. Una possibile spiegazione è che l'accuratezza nella misura di Kinect varia a seconda della posizione del polso, poiché il movimento rettilineo non è eseguito frontalmente al sensore.

Il modello permette di ottenere una traiettoria stimata più attendibile rispetto quelle stimate da Kinect. L'andamento della traiettoria, infatti, assomiglia maggiormente a quello registrato dal sensore Optitrack, risultandone quasi parallelo. Il modello, attraverso il filtro di Kalman e imposizione delle lunghezze

fisse, consente di ottenere una migliore stima, soprattutto in casi critici come questo, perfino in presenza di derive.

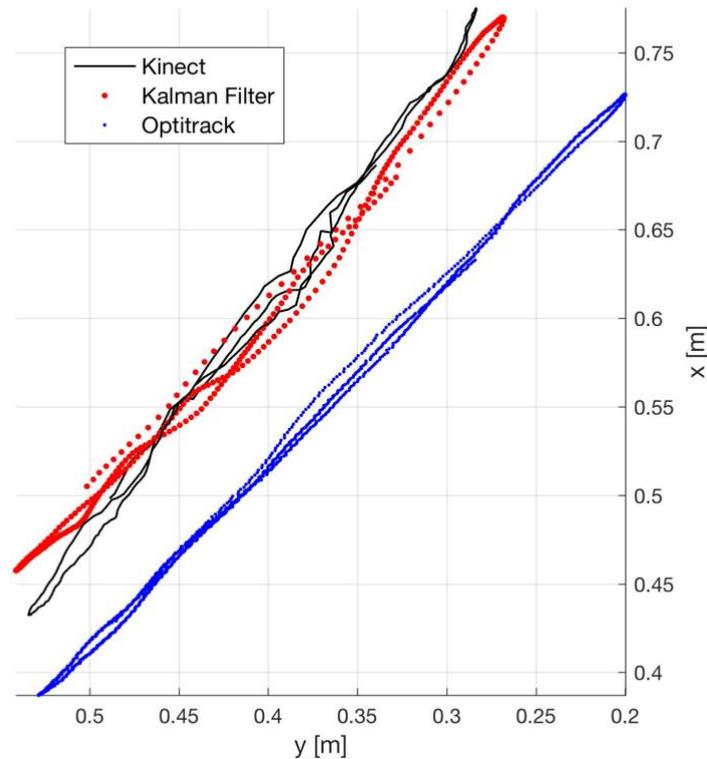


Figura 4.44 Traiettoria sul piano x-y del giunto polso, dato grezzo Kinect $\mathbf{p}_{e,2}^W$, stimato con filtro di Kalman $\hat{\mathbf{p}}_{e,2}^W$ e dato Optitrack $\mathbf{p}_{e,2M}^W$, prova 3

4.9.3 Analisi modulo delle velocità

La presenza dell'off-set evidenziato in precedenza, non compromette l'analisi sul modulo delle velocità. Si confrontano i moduli delle velocità, ricavate mediante derivazione numerica, per Optitrack e Kinect. Si concentra l'analisi sulle due corse idealmente rettilinee eseguite completamente.

Il sensore Optitrack ha frequenza di campionamento pari a 120 Hz, mentre Kinect ne ha una di 30 Hz. Per poter confrontare efficacemente i due segnali, si effettua una preliminare azione di down-sampling del segnale di Optitrack in modo da portarlo a 30 Hz.

Non si filtra la velocità con filtro passa basso a valle in quanto introdurrebbe ritardi slegati dal filtro di Kalman. In Figura 4.45 si mostra l'andamento nel tempo del modulo della velocità del polso, calcolata a partire dai dati Optitrack e dalla

stima prodotta dal modello. Si mostrano i risultati prima e dopo il down-sampling dei dati sulla posizione del marker registrata da Optitrack.

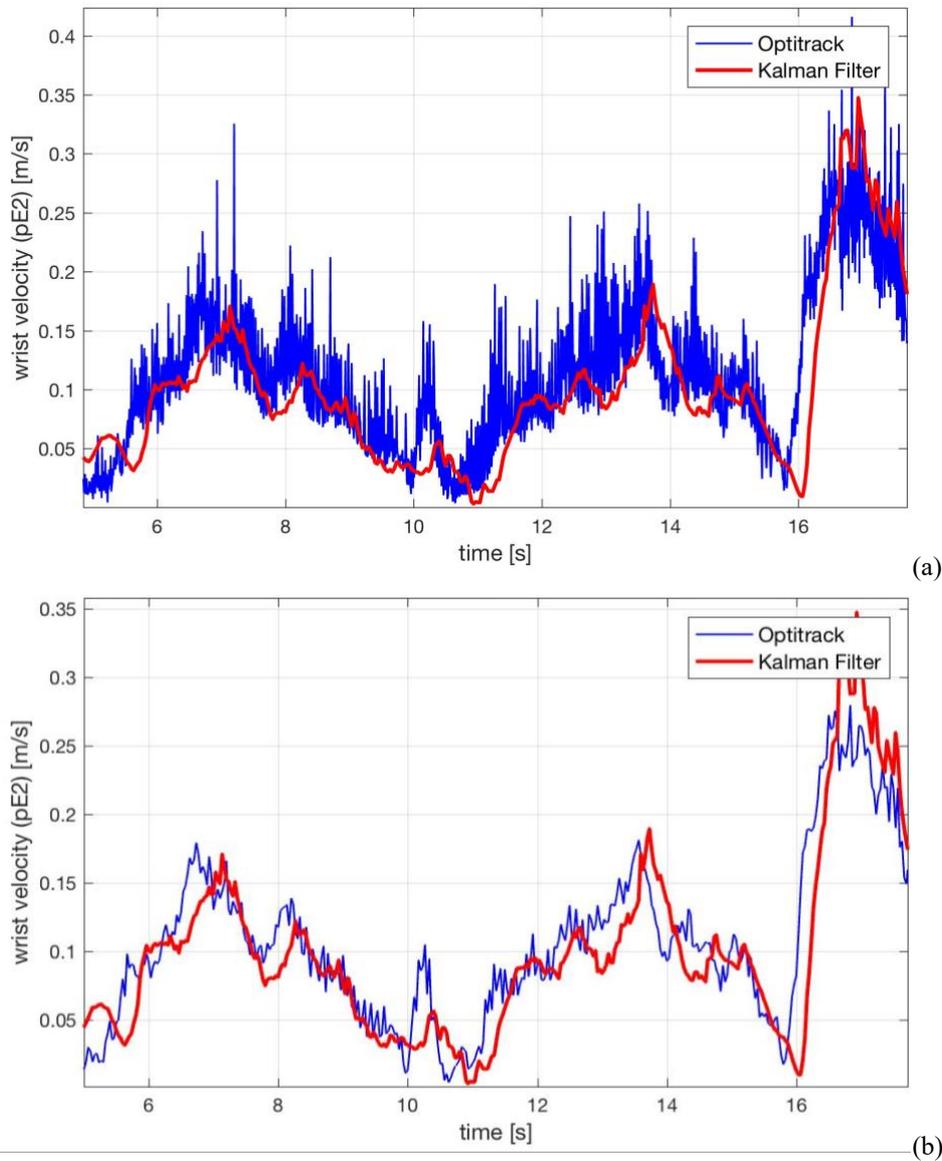


Figura 4.45 Analisi sul modulo delle velocità del polso, limitatamente alle traiettorie rettilinee, confronto fra Optitrack e modello, prima (a) e dopo (b) down-sampling di Optitrack, prova 3

La velocità calcolata a partire dalla stima prodotta dal modello segue abbastanza bene quella calcolata a partire dai dati Optitrack. È presente un ritardo di 0,2 s circa. È lecito pensare che tale ritardo sia proprio del funzionamento del filtro di Kalman.

Si confrontano, inoltre, le velocità calcolate a seguito del filtro di Kalman con quelle calcolate dai soli dati grezzi di Kinect. I risultati sono mostrati in Figura 4.46.

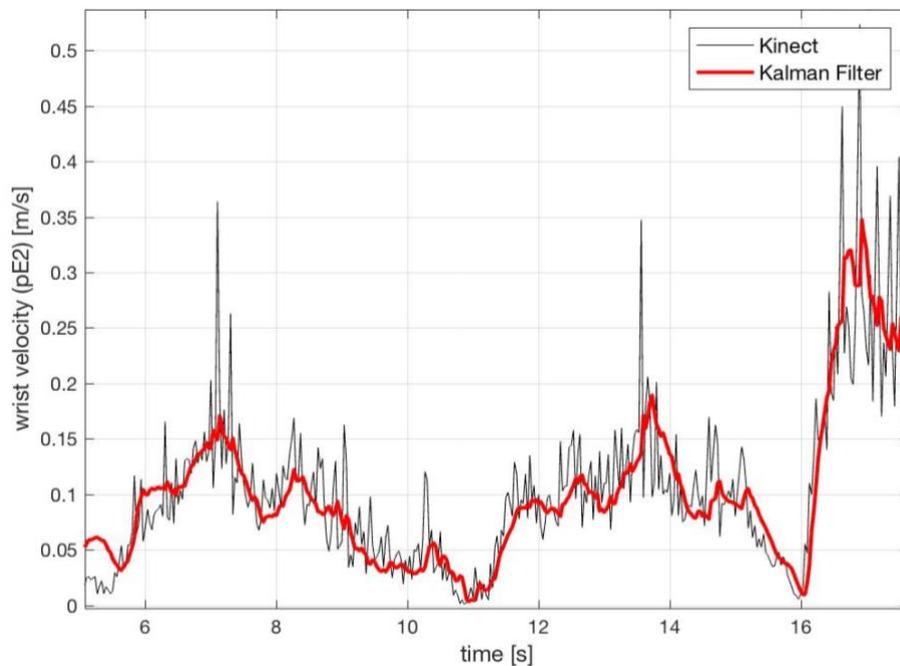


Figura 4.46 Analisi sul modulo delle velocità del polso, velocità ottenute dalle stime del modello e dai dati grezzi Kinect, prova 3

È evidente come l'algoritmo elaborato permetta di ottenere un effetto filtrante sulle velocità. Questo indica come la stima sulle posizioni sia più dolce e meno soggetta a bruschi salti, presenti nel segnale grezzo di Kinect, come testimoniano i picchi registrati sull'andamento del modulo della velocità.

In Figura 4.46 il ritardo delle velocità a partire dai dati stimati sembra visivamente inferiore a quello che si mostrava in Figura 4.45. Si produce, in Figura 4.47, un confronto diretto fra l'andamento del modulo della velocità a partire dai dati grezzi di Kinect e quelli forniti da Optitrack. In alcuni tratti, caratterizzati da una particolare accelerazione, la velocità prodotta dai dati Kinect sembra andare in ritardo. Dunque il ritardo complessivo presente in Figura 4.45 sarebbe il risultato sia del ritardo proprio del filtro di Kalman, sia di un eventuale ritardo del sensore Kinect. Questo aspetto è inatteso e tutto da indagare. Tuttavia si potrebbe pensare che tale ritardo sia prodotto non tanto dal sensore Kinect, quanto dall'SDK incaricato dell'estrapolazione delle informazioni di posizione dei giunti.

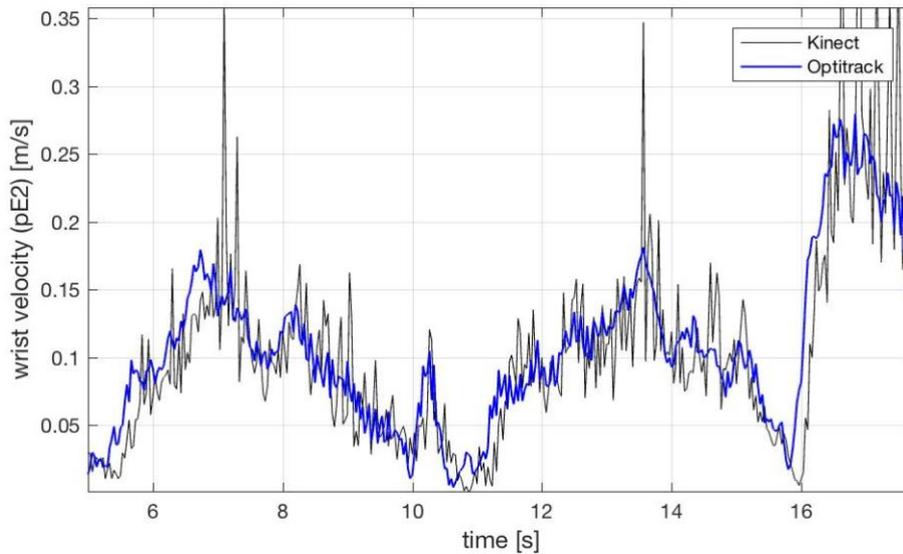


Figura 4.47 Analisi sul modulo delle velocità del polso, velocità ottenute dai dati grezzi Kinect e Optitrack a 30 Hz, prova 3

4.9.4 Analisi direzione e verso delle velocità

Si vuole eseguire un'analisi anche su direzione e verso della velocità. Si calcola il versore di velocità istantaneo $\hat{\mathbf{u}}_k$, definito come $\hat{\mathbf{u}}_k = \frac{\mathbf{u}_k}{\|\mathbf{u}_k\|}$ con $\mathbf{u}_k = \dot{\mathbf{p}}_{e,2k}^W$.

Per la prova 3, per una prima analisi, si introduce un versore ideale di riferimento $\hat{\mathbf{v}}$, indicante direzione e verso ideali della velocità coerentemente al movimento eseguito. Tale direzione corrisponde alla direzione del filo utilizzato come guida per l'esecuzione del movimento. Idealmente, la direzione della velocità deve corrispondere con la direzione del filo. In pratica ciò non è vero a causa di un movimento non perfettamente rettilineo eseguito. Si presenta dunque solo una prima analisi, da approfondire con prove specifiche che abbiano una direzione del movimento controllata. In Tabella 4.7 si mostrano i versori di riferimento utilizzati corrispondenti alle due corse rettilinee eseguite.

Tabella 4.7 Versore ideale di riferimento per le due corse effettuate nella prova 3

Movimento	I	I
Versore di riferimento	$\hat{\mathbf{v}}_I = \left(-\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}, 0\right)$	$\hat{\mathbf{v}}_{II} = \left(\frac{\sqrt{2}}{2}, -\frac{\sqrt{2}}{2}, 0\right)$

Il confronto viene ottenuto con l'introduzione dell'angolo istantaneo ϑ_k che si crea fra la velocità istantanea sperimentale $\hat{\mathbf{u}}_k$ e la direzione di riferimento $\hat{\mathbf{v}}$. I valori di ϑ_k costituiscono il vettore $\boldsymbol{\vartheta}$. Si analizza dunque il valore medio ϑ_m e la varianza di tutti i ϑ_k per ciascun movimento e per le diverse tipologie di dati a disposizione. La Tabella 4.8 mostra i risultati ottenuti.

Tabella 4.8 Analisi sull'angolo ϑ medio tra verso della velocità istantanea e ideale, confronto fra dati Kinect, stimati dal modello e Optitrack a 30 Hz, prova 3

	Dati grezzi Kinect	Stima con modello	Optitrack
Movimento I	$16,1^\circ \pm 12,4^\circ$	$10,5^\circ \pm 5,0^\circ$	$6,7^\circ \pm 5,0^\circ$
Movimento II	$18,0^\circ \pm 12,4^\circ$	$11,1^\circ \pm 5,7^\circ$	$7,7^\circ \pm 5,4^\circ$

Come si è detto, la significatività dei dati è da valutare, in quanto il reale movimento compiuto non è secondo il verso di $\hat{\mathbf{v}}$ e tantomeno non ha direzione costante. Tuttavia i dati ottenuti relativi alla covarianza, danno conto di come le direzioni delle velocità stimate con il modello siano più stabili e meno soggette a fluttuazioni rispetto a quelle calcolate dai dati Kinect.

4.10 Confronto con nuvola di punti Kinect

La valutazione della bontà della stima ottenuta mediante il modello sviluppato è ottenibile mediante un confronto con il sistema oggetto di osservazione. Si è cercato di ottenere una prima valutazione confrontando la stima ottenuta dal modello con i valori misurati con un sensore più accurato di Kinect. In questo modo è possibile confrontare un dato rappresentativo dello stato reale del sistema con il dato stimato. La valutazione sulla posizione dei giunti ottenuta mediante Optitrack non è risultata particolarmente esaustiva a causa di off-set dovuti alla presenza dei marker applicati al corpo. Si decide così di collocare le posizioni stimate direttamente sulla nuvola di punti fornita dal sensore di profondità Kinect. In questo modo è possibile verificare visivamente se la stima ottenuta è rappresentativa del reale movimento compiuto. Si vuole quindi ottenere un video della prova con l'acquisizione della nuvola di punti, in cui venga rappresentato anche lo scheletro relativo al braccio umano dei dati grezzi Kinect e dei dati stimati dal modello. In Figura 4.34 si era mostrata una rappresentazione di questo tipo, ma con lo scheletro stimato separato dall'immagine RGB. Almeno in questa fase, si preferisce rappresentare la stima ottenuta direttamente sulla nuvola di punti e non sull'immagine RGB di Kinect.

Per poterne ottenere la visualizzazione della nuvola di punti, è necessario utilizzare l'output "*depthMap*" fornito da Kinect, mai utilizzato fino a questo istante. Il formato "*depthMap*" deve essere convertito nel formato "*ptCloud*" costituito da una matrice ($424 \times 512 \times 3$), in cui le prime due dimensioni rappresentano i pixel della Kinect v2 e la terza le coordinate di ciascun pixel. Tale operazione è eseguita mediante la funzione "*depthToCloud*" fornita da [22]. I valori di parametri intrinseci di Kinect, come la lunghezza focale della lente, sono stati modificati sulla base dello script "*calibrationDemo*" presente in [23].

Tabella 4.9 Valori adottati dei parametri intrinseci di Kinect

Focal Length X	367,4633
Focal Length Y	367,4633
Principal Point X	256,2120
Principal Point Y	208,8752

Si ottengono così le coordinate di tutti i punti della nuvola nel sistema di riferimento Kpc . Attraverso le Eq. (4.2) e (4.9) è possibile ricavare le coordinate di tutti i punti della nuvola nel sistema di riferimento W . A questo punto si rappresentano i punti misurati $p_{e,i}^W$ e stimati $\hat{p}_{e,i}^W$ relativi ai giunti nella nuvola di punti nel sistema di riferimento W . Viene dunque prodotto un video in grado di visualizzare l'evoluzione nel tempo delle posizioni dei punti. In Figura 4.48 si mostra un frame ottenuto.

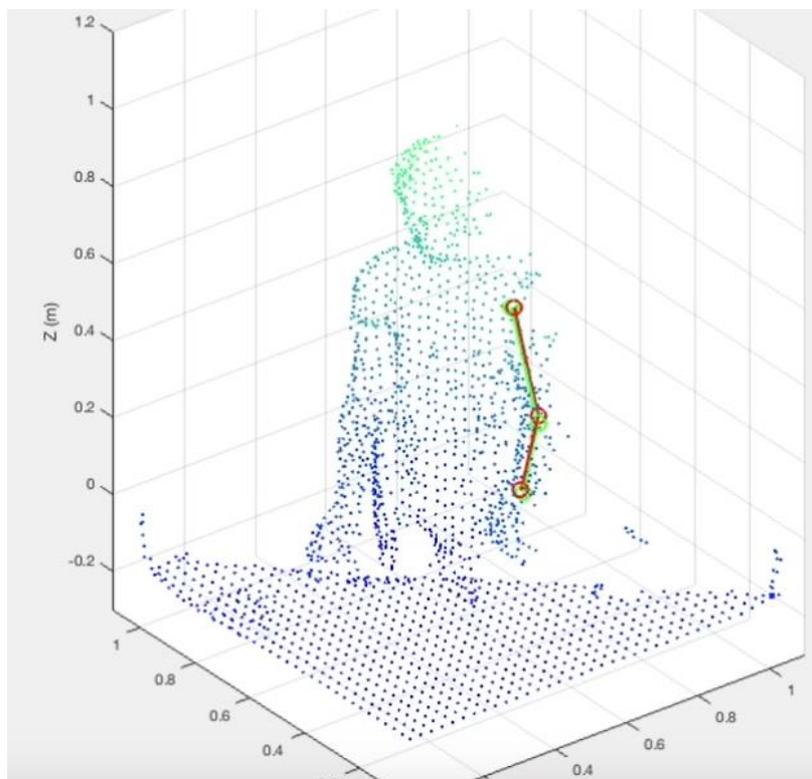


Figura 4.48 Frame con dati sulla posizione dei giunti nella nuvola di punti, in verde la misura di Kinect, in rosso la stima del modello, prova 3

Il video mostra come la stima ottenuta, relativa alla prova 3, sia ben rappresentativa della reale posizione del braccio umano. Inoltre si evidenzia come essa permetta di ottenere un movimento più fluido del braccio rispetto ai valori grezzi di Kinect. Il movimento del braccio risulta adeguato anche in casi critici in cui si ha una misura non accurata, con punti della nuvola che si disperdono particolarmente. Durante la prima fase della prova 3, l'esecuzione di movimenti particolarmente veloci e bruschi, evidenziano un ritardo nella stima prodotta dal filtro di Kalman, tuttavia il segnale ottenuto risulta privo di particolare rumore. Si

può pensare di poter modificare i valori di covarianza σ_q^2 e σ_r^2 per regolare questo aspetto. Come si è detto, per applicazioni di *hand following* ci si aspetta che l'operatore abbia un comportamento collaborativo prediligendo dinamiche del braccio più basse per poter essere seguito efficacemente dal robot.

Capitolo 5

Modello completo

Nel corso del capitolo precedente, si sono mostrati gli sviluppi dell'algoritmo che porta alla miglior stima del movimento del braccio umano a partire dai dati forniti dal sensore Kinect. Si offre ora una visione di insieme, mostrando le varie parti di cui è costituito il modello e dei parametri da fornire.

In Figura 5.1 viene riassunto il modello mediante una rappresentazione a blocchi. Il sensore di profondità Kinect osserva il sistema e, mediante il suo SDK, fornisce le informazioni di posizione sui giunti dello scheletro \mathbf{p}_{e,i_k}^K nel proprio sistema di riferimento K , in ogni istante k . Le coordinate delle posizioni dei giunti vengono poi riportate nel sistema di riferimento World W , ottenendo \mathbf{p}_{e,i_k}^W .

Viene dunque applicato un filtro passa basso in real-time sulle posizioni della spalla $\mathbf{p}_{e,0_k}^K$ e sulle posizioni di due giunti \mathbf{p}_{e,ss_k}^K e \mathbf{p}_{e,sm_k}^K .

Si ottengono dei valori filtrati che vengono utilizzati, istante per istante, per la definizione della matrice di rototraslazione $\mathbf{A}_{O_k}^W$ che lega il sistema di riferimento World con quello spalla. Il sistema di riferimento spalla O è un sistema roto-traslante: ciò permette di ottenere una terna di riferimento coerente rispetto al modello del braccio sviluppato, quale che sia l'orientamento e lo spostamento del busto durante il tracciamento. Si esegue dunque un cambiamento del sistema di riferimento, da W a O , limitatamente alle posizioni di gomito $\mathbf{p}_{e,1_k}^O$ e polso $\mathbf{p}_{e,2_k}^O$.

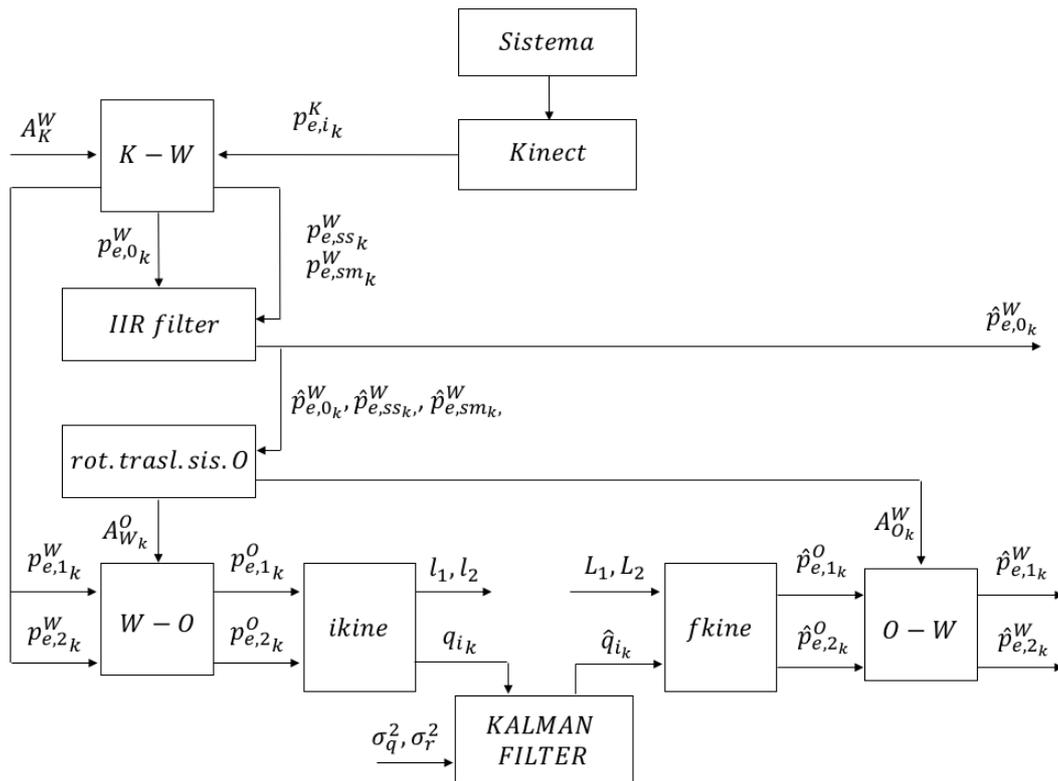


Figura 5.1 Schema del modello completo

Attraverso cinematica inversa (*ikine*) si ottengono i valori delle variabili di giunto q_{i_k} . Si ottengono, inoltre, i valori istantanei della misura della lunghezza braccio l_1 e avambraccio l_2 . Si ricorda che Il braccio umano è stato modellizzato come un robot antropomorfo a 4 gradi di libertà: 3 al giunto spalla e 1 sul giunto gomito.

Le variabili di giunto q_{i_k} costituiscono la misura \mathbf{z}_k per il filtro di Kalman. Si rimanda alla Figura 3.12 per la schematizzazione del suo funzionamento. Il filtro di Kalman è basato su un modello di predizione di tipo cinematico ad accelerazione costante, esplicitato dalla matrice \mathbf{G} . Il KF esegue un'azione filtrante che dipende dai valori di covarianza di processo σ_q^2 e della misura σ_r^2 impostati. Si ottiene dunque lo stato aggiornato nel quale è presente la stima delle variabili di giunto \hat{q}_{i_k} .

Viene eseguita la cinematica diretta (*fkine*) per passare da spazio giunti a spazio operativo, ottenendo la posizione stimata del di gomito $\hat{\mathbf{p}}_{e,1_k}^O$ e polso $\hat{\mathbf{p}}_{e,2_k}^O$. Questa operazione è effettuata mediante l'imposizione di lunghezze fisse per braccio L_1 e

avambraccio L_2 . Ciò permette di ottenere un'ulteriore azione filtrante, come descritto nel paragrafo 4.6.

A questo punto, mediante la matrice $A_{O_k}^W$ precedentemente calcolata, si è in grado di ottenere le posizioni di gomito $\hat{\mathbf{p}}_{e,1_k}^W$ e polso $\hat{\mathbf{p}}_{e,2_k}^W$ nel sistema di riferimento World. Con esse è anche disponibile la posizione della spalla $\hat{\mathbf{p}}_{e,0_k}^W$ precedentemente filtrata mediante filtro passa-basso.

5.1 Input da fornire al modello

Prima di eseguire l'algoritmo, bisogna definire una serie di parametri necessari al suo corretto funzionamento. Tali parametri sono:

- lunghezze fisse di braccio L_1 e avambraccio L_2 ;
- varianze di processo σ_q^2 e della misura σ_r^2 ;
- frequenza di taglio f_{cut} del filtro passa-basso;
- matrice di rototraslazione A_K^W fra il sistema di riferimento Kinect e World.

Inoltre bisogna specificare inizialmente su quale braccio, destro o sinistro, eseguirlo.

Si è già detto come le lunghezze fisse di braccio L_1 e avambraccio L_2 possano essere ottenute mediante un'operazione di taratura, attraverso l'esecuzione di movimenti che notoriamente sono favorevoli a una più corretta misura da parte di Kinect. Qualora non fosse disponibile questa possibilità, si consiglia di utilizzare una media mobile utilizzando i valori misurati di braccio l_1 e avambraccio l_2 fino a quel punto rilevati.

La scelta delle varianze di processo σ_q^2 e della misura σ_r^2 incide sulla prestazione del filtro di Kalman. Tali parametri consentono la regolazione della sua azione filtrante e, di conseguenza, dell'eventuale ritardo ottenuto. Essi devono essere regolati a fronte della dinamica del sistema osservato.

La frequenza di taglio f_{cut} del filtro passa-basso permette la regolazione dell'azione filtrante sulla spalla e sul movimento del sistema riferimento spalla. Si accettano maggiori ritardi, a fronte di una maggiore fluidità del movimento ottenuta.

La matrice di rototraslazione A_K^W fra il sistema di riferimento Kinect e World è ottenuta mediante un'operazione di calibratura. Essa viene determinata, una volta per tutte, a seguito della definizione del set-up sperimentale e delle posizioni relative fra il sensore Kinect e il sistema di riferimento World adottato.

5.2 Tempi di calcolo

Si analizzano i tempi di calcolo relativi all'algoritmo elaborato e alle varie fasi che lo caratterizzano. L'algoritmo viene eseguito in Matlab da una macchina con sistema operativo macOS, processore dual-core 2,3 GHz Intel Core i5, memoria RAM 16 GB 1333 MHz. Tale macchina è stata utilizzata anche nelle analisi precedenti.

Vengono individuate cinque fasi principali che caratterizzano il modello. La prima è costituita dall'estrapolazione dati dal sensore Kinect, fino ad ottenere le posizioni dei giunti nel sistema di riferimento World. La seconda è costituita dal filtro passa-basso. La terza contiene l'elaborazione della terna levogira del sistema di riferimento spalla e il passaggio alle coordinate del sistema di riferimento spalla. La quarta, la più complessa, è costituita dalla cinematica inversa, applicazione del filtro di Kalman e cinematica diretta. La quinta, infine, rappresenta il passaggio da coordinate nel sistema di riferimento spalla a World.

In Tabella 5.1 è mostrato il tempo di calcolo medio di ciascuna fase, considerando tutte le iterazioni eseguite per l'elaborazione della prova 3. Si mostra infine il tempo di calcolo medio relativo all'esecuzione di una iterazione del modello completo.

Tabella 5.1 Tempi di calcolo medio per iterazione nelle diverse fasi dell'algoritmo, prova 3

#	Fasi dell'algoritmo	Tempo di calcolo medio [ms]
1	Kinect2World	0,0860
2	Low-pass filter	0,0291
3	World2shoulder	0,0860
4	ikine + Kalman filter + fkine	0,2188
5	Shoulder2World	0,0329
	Intero algoritmo	0,4473

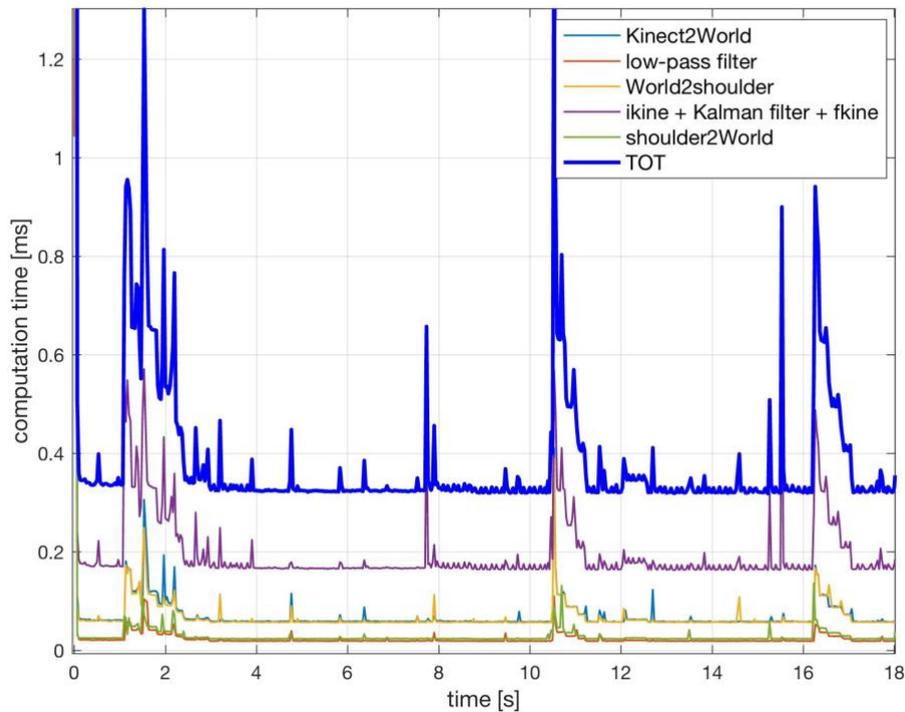


Figura 5.2 Tempi di calcolo nel tempo per iterazione nelle diverse fasi dell'algoritmo, prova 3

In Figura 5.2 si offre la rappresentazione nel tempo, del tempo di calcolo di ciascuna fase per ogni iterazione.

Come prevedibile la fase 4, contenente il filtro di Kalman, risulta essere la più dispendiosa. Subito dopo vi sono le fasi 1 e 2 con l'estrapolazione dati da Kinect e il calcolo della matrice istantanea di rototraslazione tra sistema di riferimento World e spalla. Infine, richiedono meno costo computazionale, il filtro passa basso e il semplice passaggio finale dalle coordinate del sistema riferimento spalla a World. I risultati in termini di tempi di calcolo complessivi risultano essere abbastanza buoni, risultando mediamente nell'ordine di mezzo millisecondo per iterazione.

Conclusioni

L'obiettivo principale del lavoro è stato lo sviluppo di un modello che permetta di ottenere un movimento del braccio umano più accurato e fluido, a partire dai dati grezzi forniti dal sensore Kinect. In pratica si è cercato di ottenere una migliore stima delle posizioni del giunto gomito e polso. Tali giunti sono quelli che hanno le dinamiche più alte, inoltre sono quelli più interessanti in applicazioni di robotica collaborativa.

Si è individuato nel *filtro di Kalman* lo strumento ideale da applicare. Esso deve basarsi su un modello in grado di descrivere il sistema oggetto di studio. La scelta è ricaduta su un modello cinematico ad accelerazione costante. Tale modello è lineare, facile da implementare e rapido in fase di esecuzione. Il sistema oggetto di studio è il braccio umano, modellizzato con quattro gradi di libertà: tre per il giunto spalla e uno per il gomito.

Si utilizzano le *matrici di covarianza del processo e della misura* del filtro di Kalman, come parametri di regolazione della sua azione filtrante e del ritardo generato nel segnale.

L'azione filtrante del filtro di Kalman agisce sulle variabili di giunto del modello del braccio. Ciò non è sufficiente per ottenere nello spazio operativo, risultati apprezzabili. Si è resa necessaria l'imposizione di *lunghezze fisse* per braccio e avambraccio, da applicare in fase di cinematica diretta. Tali lunghezze sono ottenibili attraverso una taratura, da eseguire precedentemente. In alternativa è possibile utilizzare una stima basata su una media mobile dei dati registrati.

Il filtro di Kalman e l'imposizione delle lunghezze fisse di braccio e avambraccio sono i due principali elementi filtranti utilizzati per la stima delle posizioni dei giunti gomito e polso.

Per poter ottenere le variabili di giunto da consegnare al filtro di Kalman, è necessaria un'operazione di cinematica inversa a partire dalle posizioni di gomito e polso in un sistema di riferimento avente come origine il giunto spalla e solidale con il busto dell'uomo. Tale sistema di riferimento è dunque in movimento. Per ottenere dei movimenti fluidi si è implementato un *filtro passa-basso* sulla posizione del giunto spalla e su altri due giunti dello scheletro utilizzati per la costruzione del sistema di riferimento mobile. A seguito di filtro di Kalman e cinematica diretta, è poi semplice tornare ad esprimere le posizioni stimate nel sistema di riferimento fisso di lavoro.

La valutazione dell'affidabilità della stima delle posizioni di spalla, gomito e polso, è ottenuta mediante la rappresentazione delle stesse sulla *nuvola di punti* fornita dal sensore Kinect. Si sono elaborati video che mostrano l'azione filtrante e le corrette stime ottenute.

La valutazione dell'affidabilità delle velocità ottenute a seguito dei dati stimati, è effettuata ricorrendo a un sensore con migliore accuratezza, *Optitrack*. Si è effettuata una valutazione sul modulo delle velocità, che ha mostrato ottimi risultati, ottenendo valori che descrivono efficacemente il sistema osservato. Si rende necessaria comunque una migliore valutazione sulla direzione e verso delle velocità, con l'esecuzione di prove controllate.

È possibile avere informazioni sul movimento con certi istanti di anticipo ricorrendo alla stessa fase di *predizione* del filtro di Kalman, oppure utilizzando le informazioni sulle sole velocità calcolate.

Come sviluppi futuri si consiglia una migliore regolazione delle matrici di covarianza di processo e di misura del filtro di Kalman, considerando la possibilità di valori adattativi; la definizione di una procedura di taratura per le lunghezze fisse di braccio e avambraccio che ne massimizzi l'accuratezza; l'indagine di ulteriori filtri passa-basso con migliori prestazioni.

Bibliografia

- [1] H. Liu e L. Wang, «Human motion prediction for human-robot collaboration» *Journal of Manufacturing Systems*, vol. 44, p. 287–294, 2017.
- [2] J. Han, L. Shao, D. Xu e J. Shotton, «Enhanced Computer Vision with Microsoft Kinect Sensor: A Review» *IEEE Transactions on Cybernetics*, vol. 43, n. 5, 2013.
- [3] J. Nagi, F. Ducatelle, G. A. D. Caro, D. Ciresan, U. Meier, A. Giusti, F. Nagi, J. Schmidhuber e L. M. Gambardella, «Max-Pooling Convolutional Neural Networks for Vision-based Hand Gesture Recognition» *IEEE International Conference on Signal and Image Processing Applications*, 2011.
- [4] G. Cicirelli, C. Attolico, C. Guaragnella e T. D'Orazio, «A Kinect-based Gesture Recognition Approach for a Natural Human Robot Interface» *International Journal of Advanced Robotic Systems*, 2014.
- [5] A. Kumar, A. Kumar, S. K. Singh e R. Kala, «Human Activity Recognition in Real-Time Environments using Skeleton Joints» *International Journal of Interactive Multimedia and Artificial Intelligence*, vol. 3, n. 7, 2016.
- [6] E. Coupeté, «Gesture recognition using a depth camera for human robot collaboration on assembly line» *Procedia Manufacturing*, n. 3, pp. 518-525, 2015.
- [7] J. Mainprice e D. Berenson, «Human-Robot Collaborative Manipulation Planning Using Early Prediction of Human Motion» *International Conference on Intelligent Robots and Systems (IROS)*, 2013.
- [8] S. Y. Chen, «Kalman Filter for Robot Vision: A Survey» *Transaction on Industrial Electronics*, vol. 59, n. 11, pp. 4409-4420, 2012.
- [9] S. Moon, Y. Park, D. W. Ko e a. I. H. Suh, «Multiple Kinect Sensor Fusion for Human Skeleton Tracking Using Kalman Filtering» *International Journal of Advanced Robotic Systems*, 2016.

- [10] B. Siciliano, L. Sciavicco, L. Villani e G. Oriolo, *Robotica: Modellistica, pianificazione e controllo*, Terza Edizione a cura di, McGraw-Hill, 2008.
- [11] E. Machida, M. Cao, T. Murao e H. Hashimoto, «Human Motion Tracking of Mobile Robot with Kinect 3D Sensor» *SICE Annual Conference*, 2012.
- [12] H. Liu e L. Wang, «Gesture recognition for human-robot collaboration: A review» *International Journal of Industrial Ergonomics*, pp. 1-13, 2017.
- [13] G. Nagymáté, T. Tuchband e R. M. Kiss, «A novel validation and calibration method for motion capture systems based on micro-triangulation» *Journal of Biomechanics*, vol. 74, pp. 16-22, 2018.
- [14] A. Al-Naji, K. Gibson, S.-H. Lee e J. Chahl, «Real Time Apnoea Monitoring of Children Using the Microsoft Kinect Sensor: A Pilot Study» *Sensors*, vol. 17, p. 286, 2017.
- [15] T. Mallick, P. P. Das e A. K. Majumdar, «Characterizations of Noise in Kinect Depth Images: A Review» *SENSORS JOURNAL*, vol. 14, n. 6, 2014.
- [16] G. Biondi, «Estimation of attitude, angular rate and inertia ratios» in *Fault-tolerant feature-based estimation of space debris motion and inertial properties*, 2017, pp. 72-85.
- [17] J. V. Candy, *Bayesian signal processing : classical, modern, and particle filtering methods*, Seconda edizione a cura di, John Wiley & Sons Inc., 2016.
- [18] D. Dimitrov, «Dynamics of free-flying manipulator systems» in *Dynamics and Control of Space Manipulators During a Satellite Capturing Operation*, Tohoku University, 2005.
- [19] A. Jain e G. Rodriguez, «Linearization of Manipulator Dynamics Using Spatial Operators» *Transactions on System □ Man and Cybernetics*, vol. 22, pp. 239-248, 1993.
- [20] S. Pan, H. Su, J. Chu e H. Wang, «Applying a novel extended Kalman filter to missile–target interception with APN guidance law: A benchmark case study» *Control Engineering Practice*, vol. 18, pp. 159-167, 2010.
- [21] Y. Bar-Shalom, X. R. Li e T. Kirubarajan, «Estimation for Kinematic Models» in *Estimation with Applications to Tracking and Navigation*, John Wiley & Sons, Inc., 2001.
- [22] K. Lai, L. Bo, X. Ren e D. F. Fox, «A Large-Scale Hierarchical Multi-View RGB-D Object Dataset» *IEEE International Conference on Robotics and Automation (ICRA)*, 2011.
- [23] J. R. Terven, «Kinect 2 Interface for Matlab» [Online]. Available: <https://it.mathworks.com/matlabcentral/fileexchange/53439-kinect-2-interface-for-matlab>.