

1222·2022
800
ANNI



UNIVERSITÀ
DEGLI STUDI
DI PADOVA

UNIVERSITÀ DEGLI STUDI DI PADOVA

DIPARTIMENTO DI INGEGNERIA INDUSTRIALE

LAUREA MAGISTRALE IN INGEGNERIA AEROSPAZIALE

Stima del moto relativo di un satellite target non cooperativo tramite un sistema di stereo visione

Relatore: *Dott. Andrea Valmorbida*

Correlatore: *Prof. Enrico Lorenzini*

Correlatore: *Prof. Marco Pertile*

Laureando: *Sabino Doriguzzi Toia*

Matricola: *1233450*

Anno Accademico 2021/2022

Sommario

Negli ultimi decenni si è assistito ad un crescente interesse allo sviluppo di algoritmi per la navigazione ed il controllo di prossimità tra satelliti in orbita, usati ad esempio per eseguire manovre di rendez-vous e docking, rimozione dei detriti spaziali o ispezione di satelliti non più operativi o che hanno subito un malfunzionamento. Questo lavoro di tesi si basa sulla stima del moto relativo tramite un sistema di stereo visione tra due satelliti, un chaser controllato ed un target non operativo. Lo scenario scelto è di non cooperazione, ovvero il satellite target non comunica al chaser la propria dinamica in maniera diretta, in un'ottica riconducibile a manovre di rendez-vous con un satellite dismesso.

L'attività è svolta in via sperimentale nel laboratorio di Misure Meccaniche e Termiche presso il Dipartimento di Ingegneria Industriale dell'Università di Padova. La stima del moto, traslazionale e rotazionale, è fatta attraverso l'utilizzo di una stereo camera che acquisisce le immagini del satellite target, in questo caso un mock-up di un cubesat 2U, libero di ruotare attorno ad un asse tramite l'utilizzo di uno stadio rotativo motorizzato. Sfruttando certe componenti della facility SPARTANS, sono stati analizzati diversi scenari sperimentali: due stazionari e quattro rotazionali, con diverso assetto e diversa velocità angolare relativa. Le equazioni del moto relativo sono non lineari, pertanto per la stima viene usato un filtro di Kalman esteso (EKF) che rappresenta una soluzione molto utilizzata in stime basate su misure affette da rumore, in questo caso provenienti dall'analisi delle immagini effettuata in post-processing attraverso il software MATLAB. L'analisi delle immagini stereo permette innanzitutto di estrarre delle features comuni ad entrambe che sono poi utilizzate dal filtro di Kalman esteso per ottenere una stima del moto rela-

tivo. Quest'ultima viene poi confrontata con un riferimento fiduciario fornito da un sistema di Motion Capture per valutarne l'errore. In particolare, l'errore di stima è stato valutato considerando l'influenza di diversi fattori, quali: numero e distribuzione delle features sul target, frequenza di acquisizione delle immagini e condizioni iniziali del vettore di stato più o meno prossime al valore di riferimento. I risultati ottenuti portano a dedurre che per poter avere una buona velocità computazionale e allo stesso tempo una buona stima del moto relativo, la ricerca delle features deve essere fatta acquisendo immagini con una frequenza più bassa e le features non devono essere di numero eccessivo, ma di un numero ridotto purché siano distribuite uniformemente sul target.

Abstract

In the last decade we have assisted to a growing interest in the development of advanced navigation and control algorithms for proximity operations between satellites in orbit, used for instance to perform rendez-vous e docking maneuvers, removing of space debris or inspection maneuver of no more operative satellites or satellites with a malfunction.

This thesis focuses on the estimation of the relative motion between two satellites, a controlled chaser and a non-operative target, based on a stereo-vision system. The scenario analyzed is a non-cooperating scenario, i.e. the target satellite does not transmit its motion to the chaser, in a context of rendezvous maneuvers with a decommissioned satellite.

The experimental activity was carried out at the Mechanical and Thermal Measurements laboratory of the Department of Industrial Engineering of the University of Padua. The estimation of the translational and rotational motion is done by using a stereo camera that acquires images of the target satellite, in this case a mock-up of a 2U cubesat, that is free to rotate around one axis through the use of a motorized rotary stage.

By exploiting certain components of the SPARTANS facility, different experimental scenarios were analyzed: two stationary scenarios and four rotational ones with different attitude and angular velocities. The equations of relative motion are non-linear and therefore to have an estimation of the relative motion an Extended Kalman filter (EKF) is used. The last one is the most widely used solution in estimations based on noise-affected measurements, which in this case are coming from a post-processing image analysis carried out using MATLAB. At first,

the analysis of the stereo images allows to identify some features that are common to both images and that are then employed in the Extended Kalman Filter to have an estimation of the relative motion. After that, the last one is compared with a fiducial reference provided by a Motion Capture system to evaluate the estimation error. Specifically, the estimation error was evaluated considering the influence of different elements, as the number and the distribution of the features on the target, the image acquisition frequency and initial conditions for the state vector more or less close to the reference ones.

The experimental results showed that, to obtain a reduced computational cost and, at the same time, a good estimation of the relative motion, the image acquisition frequency has to be not so high and the detected features must not be excessive in number, but a small number of features are enough as long as they are uniformly distributed over the target.

Indice

1	Introduzione	1
1.1	Obiettivi e struttura della tesi	5
2	Background teorico	9
2.1	Dinamica traslazionale relativa	9
2.2	Dinamica rotazionale relativa	13
2.2.1	Dinamica rotazionale	14
2.2.2	Cinematica rotazionale	16
2.3	Sistema di stereo visione	18
2.3.1	Ricostruzione di un immagine digitale	19
2.3.2	Modello Pinhole camera ideale	20
2.3.3	Stereo camera	24
3	Analisi delle immagini	27
3.1	Calibrazione della stereo camera	28
3.2	Ricerca delle features	32
3.2.1	Harris-Stephens	33
3.2.2	SURF (Speeded up robust features)	34
3.3	Matching delle features	35
3.4	Rimozione degli outliers	37
3.5	Selezione delle features	39
3.6	Triangolazione delle features	40
4	Set-up sperimentale e sistemi di riferimento	43
4.1	Set-up sperimentale	43
4.1.1	Motion Capture	45
4.1.2	Satellite target	46
4.1.3	Satellite chaser e stereo camera	47
4.1.4	Stadio rotativo motorizzato	48
4.2	Sistemi di riferimento	49
4.2.1	Rototraslazione I-W e T-W	50
4.2.2	Rototraslazione C-W e I-C	53
5	Stima del moto relativo: Extended Kalman Filter	61
5.1	Struttura dell'EKF	62
5.1.1	Inizializzazione	63
5.1.2	Predizione	66

5.1.3	Aggiornamento	66
5.2	Modello dinamico	68
5.2.1	Modello dinamico della traslazione relativa	68
5.2.2	Modello dinamico della rotazione relativa	69
5.2.3	Modello dinamico delle features	72
5.2.4	Modello dinamico dei rapporti del tensore di inerzia	73
5.3	Modello di misura	73
6	Risultati dei test sperimentali	77
6.1	Scenario 1	79
6.2	Scenario 2	81
6.3	Scenario 3	82
6.4	Scenario 4	83
6.5	Scenario 5	84
6.6	Grafici dei risultati sperimentali	86
6.7	Risultati del moto traslazionale relativo	102
6.8	Risultati del moto rotazionale relativo	103
7	Conclusioni	107

Elenco delle figure

1.1	Manovra di docking con la ISS [1]	3
1.2	Fiducial markers utilizzati da SPHERES [2] [3].	4
1.3	Manovra di berthing della missione DEOS [4]	5
2.1	Rappresentazione (non in scala) del moto relativo.	11
2.2	Rappresentazione di un corpo rigido.	13
2.3	Angoli di Eulero secondo una rotazione 3-2-1.	17
2.4	Immagine digitale di un satellite target acquisita in laboratorio.	19
2.5	Modello camera Pinhole.	22
2.6	Modello stereo camera.	24
3.1	Schema procedurale dell'analisi delle features.	28
3.2	Rappresentazione dei diversi tipi di distorsione.	29
3.3	Rappresentazione della scacchiera per la calibrazione ripresa in più pose.	30
3.4	Scacchiera distorta pre-calibrazione.	32
3.5	Scacchiera rettificata post-calibrazione.	32
3.6	Features trovate dal detector Harris-Stephens.	34
3.7	Features trovate dal detector SURF.	35
3.8	Matching tra descrittori trovati con Harris-Stephens.	36
3.9	Matching tra descrittori trovati con SURF.	37
3.10	Inliers stimati con Harris-Stephens.	38
3.11	Inliers stimati con SURF.	39
4.1	Setup sperimentale.	44
4.2	Bacchette per la calibrazione del motion capture.	45
4.3	Satellite target.	46
4.4	Satellite chaser.	47
4.5	Stadio rotativo motorizzato.	49
4.6	Sistemi di riferimento e posizione dei marker sferici.	50
4.7	Dimensioni L_x , L_y e L_z del target.	53
4.8	Punti visti nel sistema di riferimento C e W nell'algorithm PnP.	54
4.9	Disposizione dei marker nella posa 1.	57
4.10	Disposizione dei marker nella posa 2.	58
4.11	Errori di riproiezione dei marker con l'algorithm PnP per entrambe le camere per le due pose.	59
5.1	Schematizzazione dell'algorithm EKF.	62

5.2	Vertici del target per l'inizializzazione della rotazione dell'EKF trovati usando la funzione <i>bwboundaries</i> di MATLAB [27].	63
6.1	Scenario 1: selezione di tutte le features trovate con l'algoritmo SURF.	80
6.2	Scenario 3: selezione delle features ben distribuite sul target.	83
6.3	Selezione delle features concentrate sul target.	84
6.4	Errori nello scenario 1 del test 01.	87
6.5	Errori nello scenario 1 del test 03.	88
6.6	Errori nello scenario 1 del test 06.	89
6.7	Errori nello scenario 2 del test 01.	90
6.8	Errori nello scenario 2 del test 03.	91
6.9	Errori nello scenario 2 del test 06.	92
6.10	Errori nello scenario 3 del test 01.	93
6.11	Errori nello scenario 3 del test 03.	94
6.12	Errori nello scenario 3 del test 06.	95
6.13	Errori nello scenario 4 del test 01.	96
6.14	Errori nello scenario 4 del test 03.	97
6.15	Errori nello scenario 4 del test 06.	98
6.16	Errori nello scenario 5 del test 01.	99
6.17	Errori nello scenario 5 del test 03.	100
6.18	Errori nello scenario 5 del test 06.	101
6.19	Errori rms della traslazione per tutti gli scenari dei test considerati. .	102
6.20	Errori rms della velocità traslazionale per tutti gli scenari dei test considerati.	103
6.21	Errori rms della rotazione per tutti gli scenari dei test considerati. . .	104
6.22	Errori rms della velocità rotazionale per tutti gli scenari dei test considerati.	105

Elenco delle tabelle

2.1	Valori di intensità luminosa dell'immagine dei primi 10×10 pixel della (Fig. 2.4).	20
4.1	Dimensione del target.	47
4.2	Principali caratteristiche della stereo camera.	48
4.3	Errori di riproiezione dei marker con l'algoritmo PnP per entrambe le camere.	58
4.4	Confronto tra la rototraslazione tra le due camere ottenuta con l'algoritmo di Zhang e quella ottenuta con l'algoritmo PnP.	59
6.1	Caratteristiche dei due test stazionari. La posa è espressa nel sistema di riferimento del chaser e calcolata usando il Motion Capture.	78
6.2	Caratteristiche dei quattro test rotazionali. La posa è espressa nel sistema di riferimento del chaser e calcolata usando il Motion Capture.	78
6.3	Matrici iniziali dell'EKF per lo scenario 1 per tutti i test.	81
6.4	Stato iniziale dello scenario 1.	81
6.5	Stato iniziale dello scenario 5.	85
6.6	Errori rms nello scenario 1 del test 01.	87
6.7	Errori rms nello scenario 1 del test 03.	88
6.8	Errori rms nello scenario 1 del test 06.	89
6.9	Errori rms nello scenario 2 del test 01.	90
6.10	Errori rms nello scenario 2 del test 03.	91
6.11	Errori rms nello scenario 2 del test 06.	92
6.12	Errori rms nello scenario 3 del test 01.	93
6.13	Errori rms nello scenario 3 del test 03.	94
6.14	Errori rms nello scenario 3 del test 06.	95
6.15	Errori rms nello scenario 4 del test 01.	96
6.16	Errori rms nello scenario 4 del test 03.	97
6.17	Errori rms nello scenario 4 del test 06.	98
6.18	Errori rms nello scenario 5 del test 01.	99
6.19	Errori rms nello scenario 5 del test 03.	100
6.20	Errori rms nello scenario 5 del test 06.	101

Capitolo 1

Introduzione

L'inizio dell'era spaziale a partire dalla metà del XX secolo ha rappresentato una di quelle rivoluzioni che hanno modificato a dismisura le abitudini dell'uomo. Probabilmente, senza un periodo buio come la Guerra Fredda, l'evoluzione dell'attività spaziale non sarebbe stata così verticale. Se inizialmente l'obiettivo principale fu di rimarcare la supremazia tra Unione Sovietica e Stati Uniti d'America, ben presto il fine principale diventò quello della scoperta del cosmo e di comprensione dell'universo. Infatti, dopo essere riusciti a mandare l'uomo dapprima nello spazio, successivamente in orbita ed infine sulla superficie lunare, l'interesse nello sviluppo dell'osservazione dell'universo portò alla realizzazione di una tipologia di missioni spaziali attinenti a questo fine. Ma non solo: le abitudini che oggi consideriamo normali non sarebbero le stesse senza le missioni spaziali. Telefonia, internet, navigazione, media: sono alcune delle applicazioni necessarie al giorno d'oggi.

Tra le molteplici varietà di missioni spaziali, il moto relativo tra satelliti negli ultimi decenni è stato utilizzato in maniera maggiore per le diverse necessità. Per moto relativo si intende due o più corpi orbitanti, la cui mutua distanza è molto più piccola rispetto a quella con il centro della Terra, o rispetto a qualsiasi esso sia il centro di massa. Il rendez-vous con successivo docking o berthing è sicuramente l'applicazione più emblematica del moto relativo, come ad esempio, il rendez-vous e docking con la Stazione Spaziale Internazionale per portare gli astronauti o per portare rifornimenti.

Un'altra applicazione è il monitoraggio dei detriti spaziali, le cui proprietà orbitali non sono facilmente determinabili da Terra. Per detrito spaziale si intende un oggetto che non è più controllato direttamente da Terra, quindi i satelliti dismessi, stadi esausti, coperture di telescopi, polveri, scaglie di vernici o componenti di satelliti. Per le alte velocità orbitali in gioco, una collisione tra un satellite attivo ed un detrito può arrecare danni anche irreversibili al primo, quali modifiche dell'orbita irreversibili o peggio ancora la completa distruzione del satellite, e quindi il monitoraggio con eventuale rimozione ricopre un aspetto importante.

Tutte queste tipologie di applicazioni del moto relativo necessitano di sensori atti a stimare il moto. Esistono diverse tipologie di sensori che possono essere utilizzati, ma non tutti si prestano al caso del moto relativo. Il GPS, in particolare modo il GPS differenziale, può essere usato, in particolare modo in manovre di rendez-vous e docking con la ISS. Una soluzione che può essere attuata e che risulta molto interessante e promettente è l'utilizzo di sistemi di visione, in configurazione sia monoculare (una camera) e sia stereo (due camere). L'interesse principale è dato dalla relativa semplicità di integrazione di un sistema di visione sul satellite e, non avendo parti in moto, l'usura risulta essere molto limitata. Inoltre, essendo componenti molto leggeri, si prestano molto all'utilizzo in ambito spaziale. Il problema principale è dovuto alla qualità delle immagini, che deve essere migliore possibile, e al fatto che tutti gli algoritmi necessari per l'analisi delle immagini sono piuttosto costosi dal punto di vista computazionale. Tutti questi aspetti saranno analizzati nel corso dei capitoli successivi.

L'utilizzo di un sistema di visione va contestualizzato con il livello di cooperazione che intercorre con il satellite target, in quanto le soluzioni sono diversificate in funzione dello scenario. Si possono descrivere quattro diverse tipologie di cooperazione con il target:

- *Target cooperativo attivo.*

Questo scenario riguarda manovre di rendez-vous e docking con un target noto ed ancora operativo. In questa conformazione, il satellite target comunica sempre in maniera attiva la sua dinamica ad un altro satellite detto chaser, ovvero inseguitore, tramite per esempio antenne RF. Nella Fig. 1.1 è possibile

osservare il docking con la Stazione Spaziale Internazionale, un chiaro esempio di target cooperativo attivo. Questo è il caso cooperativo più semplice ma non è quello più ricorrente.



Figura 1.1: Manovra di docking con la ISS [1]

- *Target cooperativo passivo.*

In questo scenario il target non comunica direttamente la dinamica al chaser, ma il tutto viene fatto in maniera passiva. La dinamica relativa viene determinata tramite dei *fiducial marker* attaccati alla superficie del target, che possono avere forme e dimensioni diverse in funzione del caso. La posizione dei marker fiduciali sul target è ben nota dal chaser e la loro identificazione in modo univoco attraverso un sistema di visione abbinata ad opportuni algoritmi permette di ottenere una stima del moto relativo tra i due satelliti. Questi marker, facendo un parallelismo con un concetto di tutti i giorni, consistono in una specie di QR-code, la cui scansione restituisce le informazioni cercate. Nella Fig. 1.2 si osservano i marker fiduciali utilizzati dalla missione SPHERES, la quale identificazione restituisce informazioni sul moto relativo tra due

unità.

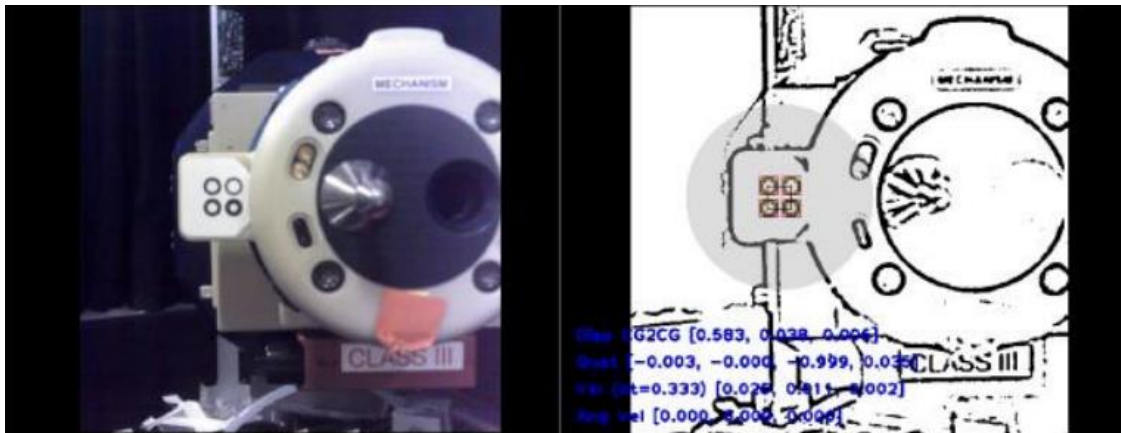


Figura 1.2: Fiducial markers utilizzati da SPHERES [2] [3].

- *Target non cooperativo noto.*

Questo è lo scenario riguardante satelliti dismessi e privati dei fiducial marker, che quindi non possono comunicare in maniera attiva o passiva la propria dinamica. Tuttavia, una stima delle proprietà geometriche e fisiche è nota dal lancio. Anche in questo scenario possono essere utilizzati dei sistemi di visione, ma la non cooperazione tra i satelliti rende gli algoritmi per la stima del moto più complessi. Due esempi di missioni in fase di sviluppo che hanno utilizzato un sistema di stereo visione per la stima del moto relativo in questo scenario sono DEOS (*Deutsche Orbitale Servicing Mission*) e OLEV (*Orbit Life Extension Vehicle*) [4].

DEOS è una missione basata su manovre di rendez-vous e berthing tramite l'utilizzo un braccio robotico. Viene utilizzato un sistema di stereo visione per eseguire le manovre, fissato sul braccio robotico e dotato anche di un dispositivo di illuminazione. Mantenendosi in prossimità del target, la stereo camera acquisisce inizialmente immagini per 2 minuti e poter stimare il moto relativo in modo da comandare il manipolatore a seguire la traiettoria corretta. Una volta iniziata la manovra di berthing, la stereo camera continua ad acquisire immagini per poter, nel caso, apportare correzioni alla traiettoria prefissata. Le manovre per il berthing eseguite da DEOS sono visibili nella Fig. 1.3.

OLEV è una missione basata su manovre di rendez-vous e docking per satelliti

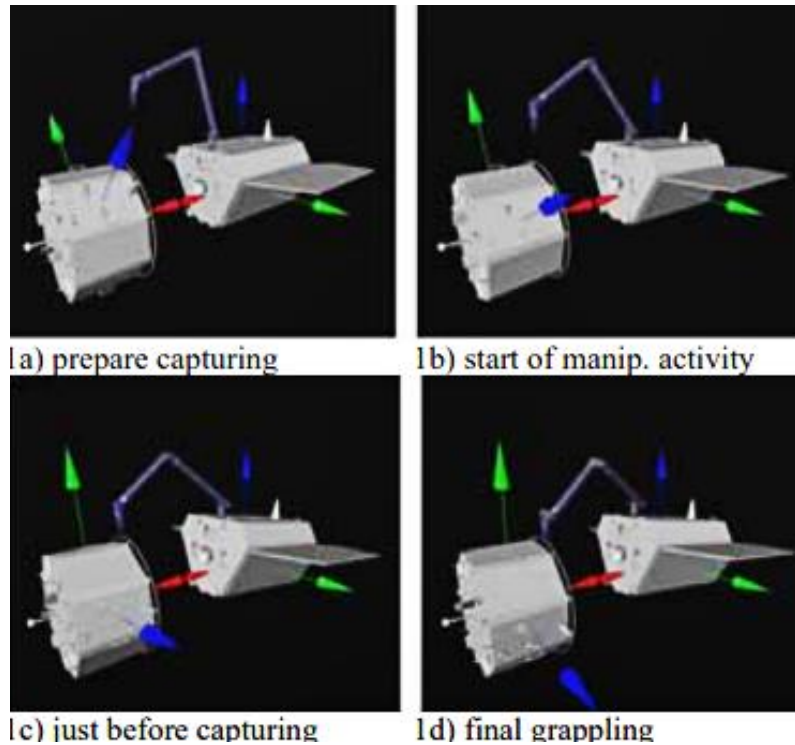


Figura 1.3: Manovra di berthing della missione DEOS [4]

in orbita GEO, il quale attracco viene attuato tramite un sistema propulsivo elettrico. Anche in questo caso viene utilizzato un sistema di visione per la stima del moto relativo, che viene attivato negli ultimi 5 metri di manovra.

- *Target non cooperativo non noto.*

Questo è lo scenario più complesso riguardante i detriti spaziali e asteroidi, le cui proprietà fisiche e dinamiche non sono note. Anche in questo caso vengono utilizzati algoritmi di visione, che sono ancora più complessi, come spiegato in [5] e [6].

1.1 Obiettivi e struttura della tesi

Lo scenario analizzato in questo lavoro di tesi riguarda un target non cooperativo noto; quindi, l'esempio di un satellite dismesso che non comunica più la dinamica al chaser. Dopo aver eseguito alcune simulazioni per riprodurre il moto rotazionale del target, vengono studiati diverse tipologie di analisi della immagini

per poter valutare quali siano le migliori da un punto di vista applicativo sulla base dei risultati ottenuti.

Il secondo capitolo è prettamente teorico e vengono illustrate le equazioni che descrivono il moto relativo. Si parte dal moto relativo traslazionale, introducendo le equazioni del moto linearizzate di Clohessy–Wiltshire (CW) che sono state utilizzate. Successivamente, si tratta il moto relativo rotazionale. Dapprima si introducono le equazioni di Eulero per la dinamica rotazionale e successivamente i quaternioni per la cinematica rotazionale: si vedrà che i quaternioni servono ad evitare problemi di singolarità che possono avvenire se si utilizzano direttamente gli angoli di Eulero. Si introduce poi il modello matematico utilizzato per descrivere il sistema di visione, partendo dal modello pinhole ideale per una singola camera e applicandolo poi ad un sistema stereo.

Nel terzo capitolo è illustrato il procedimento utilizzato per l'analisi delle immagini, necessario per la stima del moto relativo. Partendo dalla calibrazione della stereo camera, sono descritte le tecniche per la ricerca di features utili, selezionando solamente quelle di interesse.

Nel quarto capitolo è descritto il set-up sperimentale utilizzato nel corso dell'attività di tesi in tutte le sue componenti, descrivendone le peculiarità e dando anche informazioni di natura tecnica. Successivamente, sono descritti i sistemi di riferimento presi in considerazione e come calcolare tutte le matrici di rototraslazione. Viene spiegato anche come creare questi sistemi di riferimento tramite il set-up sperimentale a disposizione.

Nel quinto capitolo viene spiegato approfonditamente l'algoritmo utilizzato per la stima del moto, ovvero l'Extended Kalman Filter (EKF). L'algoritmo è descritto nella sua completezza, sottolineando i processi utilizzati ed i modelli dinamici e di misura presi in considerazione.

Nel sesto capitolo sono dapprima definiti gli scenari ed i test sperimentali eseguiti in laboratorio e descritte le diverse tecniche utilizzate per l'analisi delle immagini. Queste ultime poi vengono confrontate per valutare quale sia la soluzione migliore dal punto di vista della precisione di stima del moto relativo e del costo computazionale dell'algoritmo usato. Successivamente sono presenti i grafici

dei risultati ottenuti per tutti gli scenari sperimentali presi in considerazione. Infine, è fatto un confronto tra tutti i risultati, evidenziando vantaggi e svantaggi dei diversi scenari studiati durante l'attività sperimentale.

Nel settimo ed ultimo capitolo è descritta quella che si ritiene essere la soluzione migliore in base ai risultati sperimentali analizzati. In particolare, si vedrà che la soluzione migliore riguarderà una combinazione di diverse tecniche atte a diminuire il costo computazionale e a minimizzare l'errore nella stima. Si spiega infine perché un sistema di stereo visione è un buon sensore per la stima del moto relativo.

Capitolo 2

Background teorico

Nel corso del capitolo 1 è stato introdotto il moto relativo ed il grado di cooperazione che può intercorrere. È stato poi sottolineato che un sensore adatto alla stima del moto è un sistema di stereo visione, che verrà utilizzato nel corso di questa tesi, le quali caratteristiche verranno descritte nei capitoli e nei paragrafi successivi.

Risulta prima necessaria una spiegazione approfondita su quanto concerne il moto relativo, spiegando le varie equazioni che verranno poi utilizzate nel corso della stima del moto.

2.1 Dinamica traslazionale relativa

Prima di spiegare approfonditamente cos'è il moto relativo e le equazioni che lo descrivono, è necessario introdurre l'equazione della gravitazione universale di Newton [7]. Siano due corpi, uno di massa M e l'altro di massa m , sia \mathbf{r} il vettore tra i due corpi, sia G la costante di gravitazione universale. La forza gravitazionale è pari a:

$$\mathbf{F}_g = -\frac{GMm}{\|\mathbf{r}\|^3}\mathbf{r} \quad (2.1)$$

L'equazione del moto del corpo m può essere scritta dividendo (2.1) per m . Denotando infine con $\mu = GM$, essa diventa:

$$\ddot{\mathbf{r}} + \frac{\mu}{\|\mathbf{r}\|^3} \mathbf{r} = \mathbf{0} \quad (2.2)$$

Se m rappresenta un satellite artificiale e M la Terra, l'equazione differenziale (2.2) descrive il moto del satellite attorno alla Terra, con μ che denota la costante planetaria terrestre e \mathbf{r} il vettore che parte dal centro alla Terra e diretto verso il satellite.

A partire da questa equazione, di fondamentale importanza, è possibile analizzare il caso di moto relativo, ovvero del moto tra due satelliti, il chaser ed il target. Sia \mathbf{r}_0 il vettore tra il centro della Terra ed il chaser, sia $\delta\mathbf{r}$ il vettore tra il chaser ed il target e sia \mathbf{r} il vettore tra il centro della Terra ed il target, (2.2) può essere quindi espansa:

$$\delta\ddot{\mathbf{r}} = -\ddot{\mathbf{r}}_0 - \mu \frac{\mathbf{r}_0 + \delta\mathbf{r}}{\|\mathbf{r}_0 + \delta\mathbf{r}\|^3} \quad (2.3)$$

L'equazione (2.3) è fortemente non lineare, quindi talvolta non semplice dal punto di vista applicativo. Per ovviare a questa problematica, conviene linearizzare le equazioni del moto, e questo può essere fatto nel caso di moto relativo di prossimità, ovvero quando $\|\delta\mathbf{r}\| \ll \|\mathbf{r}_0\|$, portando all'introduzione delle equazioni Clohessy-Wiltshire (CW) [8]. Saltando qualche passaggio matematico e sfruttando l'ipotesi di moto relativo di prossimità, (2.3) può essere scritta come:

$$\delta\ddot{\mathbf{r}} = -\ddot{\mathbf{r}}_0 - \mu \frac{\mathbf{r}_0}{\|\mathbf{r}_0\|^3} - \frac{\mu}{\|\mathbf{r}_0\|^3} \left[\delta\mathbf{r} - \frac{3}{\|\mathbf{r}_0\|^2} (\mathbf{r}_0 \cdot \delta\mathbf{r}) \mathbf{r}_0 \right] \quad (2.4)$$

La prima parte di (2.4) rappresenta l'equazione del moto del chaser, come descritto in (2.2), e pertanto diventa:

$$\delta\ddot{\mathbf{r}} = -\frac{\mu}{\|\mathbf{r}_0\|^3} \left[\delta\mathbf{r} - \frac{3}{\|\mathbf{r}_0\|^2} (\mathbf{r}_0 \cdot \delta\mathbf{r}) \mathbf{r}_0 \right] \quad (2.5)$$

Per arrivare alla formulazione finale delle equazioni CW, occorre effettuare ancora qualche passaggio. Per semplificare i conti, si ipotizza che il chaser orbiti con moto circolare uniforme attorno alla Terra, in maniera che il vettore velocità angolare del chaser $\boldsymbol{\omega}_C$ sia costante ($\dot{\boldsymbol{\omega}}_C = \mathbf{0}$). Sotto questa ipotesi, è possibile sviluppare il

termine $\delta\ddot{\mathbf{r}}$ presente in (2.5):

$$\delta\ddot{\mathbf{r}} = \boldsymbol{\omega}_C \times (\boldsymbol{\omega}_C \times \delta\mathbf{r}) + 2\boldsymbol{\omega}_C \times \delta\mathbf{v}_{rel} + \delta\mathbf{a}_{rel} \quad (2.6)$$

dove il doppio prodotto vettoriale diventa:

$$\boldsymbol{\omega}_C \times (\boldsymbol{\omega}_C \times \delta\mathbf{r}) = \boldsymbol{\omega}_C (\boldsymbol{\omega}_C \cdot \delta\mathbf{r}) - \|\boldsymbol{\omega}_C\|^2 \delta\mathbf{r} \quad (2.7)$$

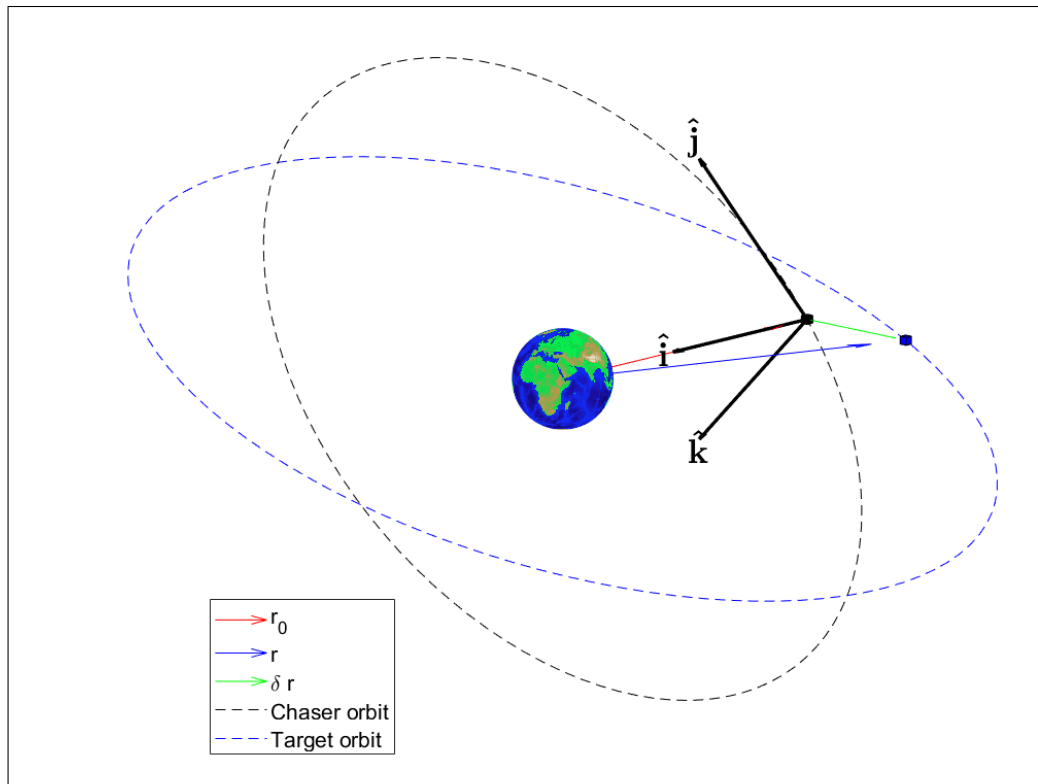


Figura 2.1: Rappresentazione (non in scala) del moto relativo.

Per proiettare i vettori lungo gli assi occorre definire un sistema di riferimento. Il sistema di riferimento scelto è quello classico costruito con il versore $\hat{\mathbf{i}}$ lungo il vettore congiungente chaser-Terra, il versore $\hat{\mathbf{j}}$ lungo il vettore velocità ed il versore $\hat{\mathbf{k}}$ lungo il vettore momento angolare, come visibile nella Fig. 2.1. È possibile quindi proiettare tutti i vettori presenti nelle equazioni di moto relativo:

$$\begin{aligned}
\delta \mathbf{r} &= \delta x \hat{\mathbf{i}} + \delta y \hat{\mathbf{j}} + \delta z \hat{\mathbf{k}} \\
\delta \mathbf{v}_{rel} &= \delta \dot{x} \hat{\mathbf{i}} + \delta \dot{y} \hat{\mathbf{j}} + \delta \dot{z} \hat{\mathbf{k}} \\
\delta \mathbf{a}_{rel} &= \delta \ddot{x} \hat{\mathbf{i}} + \delta \ddot{y} \hat{\mathbf{j}} + \delta \ddot{z} \hat{\mathbf{k}} \\
\boldsymbol{\omega}_C &= n \hat{\mathbf{k}} \\
n &= \sqrt{\frac{\mu}{\|\mathbf{r}_0\|^3}}
\end{aligned} \tag{2.8}$$

Mentre il prodotto scalare all'interno di (2.5) è:

$$(\mathbf{r}_0 \cdot \delta \mathbf{r}) \mathbf{r}_0 = \left[\|\mathbf{r}_0\| \hat{\mathbf{i}} \cdot (\delta x \hat{\mathbf{i}} + \delta y \hat{\mathbf{j}} + \delta z \hat{\mathbf{k}}) \right] \mathbf{r}_0 = (\|\mathbf{r}_0\|^2 \delta x) \hat{\mathbf{i}} \tag{2.9}$$

Infine, è possibile inserire tutte queste equazioni all'interno di (2.5) per ottenere le equazioni CW proiettate lungo gli assi cartesiani definiti dalla terna introdotta, e svolgendo qualche calcolo risulta:

$$\begin{cases}
\delta \ddot{x} - 3n^2 \delta x - 2n \delta \dot{y} = \eta_x \\
\delta \ddot{y} + 2n \delta \dot{x} = \eta_y \\
\delta \ddot{z} + n^2 \delta z = \eta_z
\end{cases} \tag{2.10}$$

dove $\boldsymbol{\eta} = (\eta_x, \eta_y, \eta_z)^T$ sono le forze esterne agenti sul chaser. Esprimendo il sistema di equazioni (2.10) in forma matriciale, ponendo $\ddot{\boldsymbol{\rho}} = (\delta \ddot{x}, \delta \ddot{y}, \delta \ddot{z})^T$, $\dot{\boldsymbol{\rho}} = (\delta \dot{x}, \delta \dot{y}, \delta \dot{z})^T$ e $\boldsymbol{\rho} = (\delta x, \delta y, \delta z)^T$, risulta:

$$\ddot{\boldsymbol{\rho}} = \begin{bmatrix} 3n^2 & 0 & 0 & 0 & 2n & 0 \\ 0 & 0 & 0 & -2n & 0 & 0 \\ 0 & 0 & -n^2 & 0 & 0 & 0 \end{bmatrix} \begin{pmatrix} \boldsymbol{\rho} \\ \dot{\boldsymbol{\rho}} \end{pmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \boldsymbol{\eta} \tag{2.11}$$

Le equazioni CW ricavate sono molto potenti in quanto descrivono il moto

relativo tra due satelliti in maniera lineare, semplificando quindi di molto il calcolo. Come si vedrà nel capitolo 5, questo sarà fondamentale per la stima del moto, in quanto il filtro di Kalman esteso (EKF) che sarà utilizzato, avrà equazioni molto più semplici. Questa linearizzazione va bene, come detto, quando $\|\delta\mathbf{r}\| \ll \|\mathbf{r}_0\|$ che in uno scenario di manovre di rendez-vous è più che sufficiente. È inoltre facile notare come le componenti all'interno del piano orbitale siano accoppiate, mentre quella lungo il momento angolare è disaccoppiata. Queste equazioni differenziali possono essere facilmente integrate e risolte in forma chiusa, ma questo non è il fine di questa tesi, in quanto come si vedrà è solo necessario avere un modello dinamico.

2.2 Dinamica rotazionale relativa

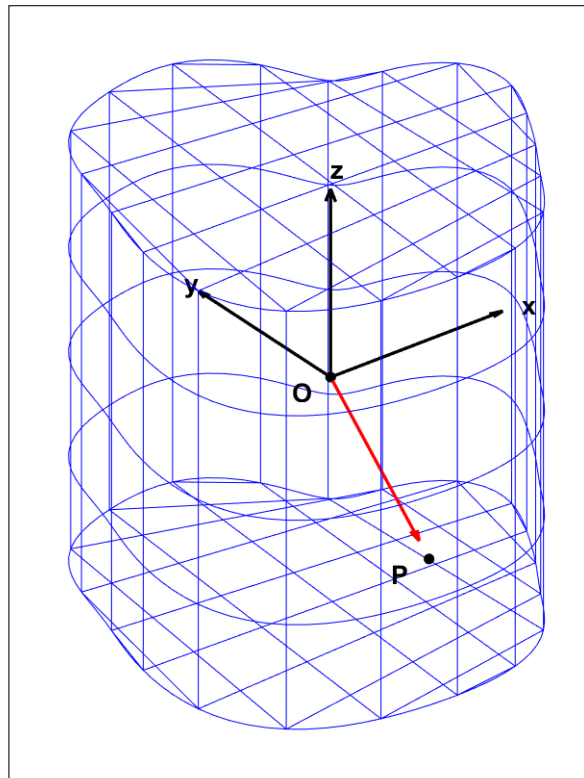


Figura 2.2: Rappresentazione di un corpo rigido.

Nel paragrafo 2.1 è stata spiegata la dinamica traslazionale relativa, ma le tre componenti non sono sufficienti per la descrizione completa di un satellite. Infatti, un satellite può essere considerato come un corpo rigido a sei gradi di libertà: tre

di traslazione e tre di rotazione. Nel seguente paragrafo verrà data una spiegazione dettagliata della dinamica rotazionale. Inoltre, come si vedrà, lo studio della dinamica rotazionale relativa deve essere diviso in due sezioni: la dinamica rotazionale e la cinematica rotazionale, la quale cinematica deriva direttamente dall'integrazione delle equazioni della dinamica.

2.2.1 Dinamica rotazionale

Sia un punto P appartenente al corpo rigido. Sia O il centro del sistema di riferimento solidale al corpo rigido e posto in corrispondenza del baricentro la cui rappresentazione è visibile nella Fig. 2.2. La proprietà del corpo rigido è data dal fatto che le coordinate del punto P rimangono fisse rispetto al sistema di riferimento solidale. In parole povere, il corpo è indeformabile. È pertanto possibile calcolare il momento angolare rispetto al polo O nel caso generale:

$$\mathbf{h} = \int_C \vec{OP} \times \mathbf{v}_P dm \quad (2.12)$$

La velocità \mathbf{v}_P del punto P può essere espressa nelle coordinate del punto O :

$$\mathbf{v}_P = \mathbf{v}_O + \mathbf{v}_{P,rel} + \boldsymbol{\omega} \times \vec{OP} \quad (2.13)$$

ove $\boldsymbol{\omega}$ è la velocità angolare del corpo, mentre $\mathbf{v}_{P,rel} = \mathbf{0}$ in quanto il corpo è rigido. Sostituendo (2.13) all'interno di (2.12):

$$\mathbf{h} = \int_C \vec{OP} dm \times \mathbf{v}_O + \int_C \vec{OP} \times (\boldsymbol{\omega} \times \vec{OP}) dm \quad (2.14)$$

Se O coincide con il baricentro, il primo integrale è nullo, pertanto il momento angolare rispetto al polo O risulta semplicemente:

$$\mathbf{h} = \int_C \vec{OP} \times (\boldsymbol{\omega} \times \vec{OP}) dm \quad (2.15)$$

Considerando $P = (x, y, z)^T$ con O posto nell'origine, è possibile definire il tensore simmetrico di inerzia 3×3 :

$$\mathbf{I} = \int_C \begin{bmatrix} y^2 + z^2 & -xy & -xz \\ -xy & x^2 + z^2 & -yz \\ -xz & -yz & x^2 + y^2 \end{bmatrix} dm = \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{xy} & I_{yy} & I_{yz} \\ I_{xz} & I_{yz} & I_{zz} \end{bmatrix} \quad (2.16)$$

Sostituendo (2.16) all'interno di (2.15) e svolgendo qualche conto, il momento angolare risulta infine essere:

$$\mathbf{h} = \mathbf{I}\boldsymbol{\omega} \quad (2.17)$$

Nella dinamica rotazionale il momento angolare è fondamentale e servirà per descriverne il comportamento, arrivando all'importante derivazione delle equazioni di Eulero [9]. Infatti, derivando il momento angolare definito rispetto al tempo e considerando che il corpo rigido stia ruotando, si ottiene che:

$$\frac{d\mathbf{h}}{dt} = \dot{\mathbf{h}} + \boldsymbol{\omega} \times \mathbf{h} \quad (2.18)$$

Dalla seconda equazione cardinale della dinamica, si ha che:

$$\frac{d\mathbf{h}}{dt} = -\mathbf{v}_O \times M\mathbf{v}_G + \mathbf{N} \quad (2.19)$$

ove M è la massa del corpo rigido, \mathbf{v}_G è la velocità del centro di massa e \mathbf{N} è la coppia esterna agente su sistema. Siccome per ipotesi il polo O è considerato coincidente con il baricentro, il prodotto vettoriale tra \mathbf{v}_O e \mathbf{v}_G restituisce il vettore nullo. Pertanto, combinando (2.19) e (2.18) si ottiene:

$$\dot{\mathbf{h}} + \boldsymbol{\omega} \times \mathbf{h} = \mathbf{N} \quad (2.20)$$

Sostituendo l'espressione per il momento angolare ricavata in (2.17) nelle ipotesi che il corpo rigido non subisca variazioni temporali delle proprietà di inerzia, si possono finalmente scrivere le equazioni di Eulero in forma vettoriale:

$$\mathbf{I}\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times \mathbf{I}\boldsymbol{\omega} = \mathbf{N} \quad (2.21)$$

Se il sistema di riferimento scelto è anche sistema principale di inerzia, il tensore di inerzia è diagonale. Quindi risulta semplice proiettare lungo gli assi cartesiani di (2.21):

$$\begin{cases} I_{xx}\dot{\omega}_x + \omega_y\omega_z(I_{zz} - I_{yy}) = N_x \\ I_{yy}\dot{\omega}_y + \omega_x\omega_z(I_{xx} - I_{zz}) = N_y \\ I_{zz}\dot{\omega}_z + \omega_x\omega_y(I_{yy} - I_{xx}) = N_z \end{cases} \quad (2.22)$$

Se ciò non fosse, la proiezione di (2.21) lungo gli assi cartesiani risulterebbe essere:

$$\begin{cases} I_{xx}\dot{\omega}_x + I_{xy}\dot{\omega}_y + I_{xz}\dot{\omega}_z + \omega_y\omega_z(I_{zz} - I_{yy}) + \\ \quad (\omega_y^2 - \omega_z^2)I_{yz} + \omega_x(\omega_y I_{xz} - \omega_z I_{xy}) = N_x \\ I_{xy}\dot{\omega}_x + I_{yy}\dot{\omega}_y + I_{yz}\dot{\omega}_z + \omega_x\omega_z(I_{xx} - I_{zz}) + \\ \quad (\omega_x^2 - \omega_y^2)I_{xz} + \omega_y(\omega_z I_{xy} - \omega_x I_{xz}) = N_y \\ I_{xz}\dot{\omega}_x + I_{yz}\dot{\omega}_y + I_{zz}\dot{\omega}_z + \omega_x\omega_y(I_{yy} - I_{xx}) + \\ \quad (\omega_x^2 - \omega_y^2)I_{xy} + \omega_x(\omega_x I_{yz} - \omega_y I_{xz}) = N_z \end{cases} \quad (2.23)$$

L'integrazione di (2.22) o (2.23) restituisce il vettore velocità angolare. Una seconda integrazione è necessaria per trovare gli angoli, detti angoli di Eulero.

2.2.2 Cinematica rotazionale

Nel paragrafo 2.2.1 si è visto come le integrazioni delle equazioni di Eulero porti a trovare il vettore velocità angolare, che a sua volta può essere integrato nel tempo per trovare gli angoli di Eulero. Essi consistono in tre rotazioni successive attorno ad assi successivi. Ne esistono di diverse, le cui più usate sono le rotazioni 3-2-1 per satelliti stabilizzati a tre assi, visibile nella Fig. 2.3, o le rotazioni 3-1-3 per satelliti spinnanti. Gli angoli sono comunemente detti ϕ per la rotazione attorno all'asse x , θ per la rotazione attorno all'asse y e ψ per la rotazione attorno all'asse z .

Sia $\boldsymbol{\alpha} = (\phi, \theta, \psi)^T$ il vettore degli angoli di Eulero, siano $\mathbf{M}_1(\boldsymbol{\alpha})$ e $\mathbf{M}_2(\boldsymbol{\alpha})$ due matrici che sono funzione degli angoli di Eulero definite in base al tipo di

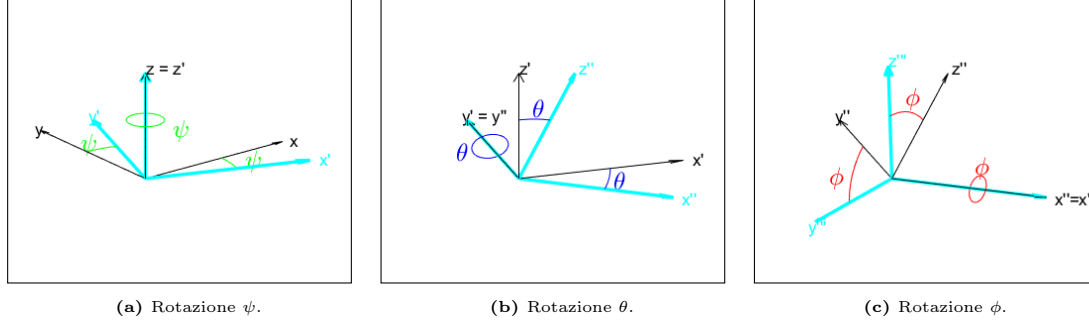


Figura 2.3: Angoli di Eulero secondo una rotazione 3-2-1.

rotazione e sia $\boldsymbol{\omega}_O$ il vettore velocità orbitale. Ricordando che il vettore $\boldsymbol{\omega}$ si trova integrando (2.21), le equazioni della cinematica sono:

$$\boldsymbol{\omega} = \mathbf{M}_1(\boldsymbol{\alpha}) \dot{\boldsymbol{\alpha}} + \mathbf{M}_2(\boldsymbol{\alpha}) \boldsymbol{\omega}_O \quad (2.24)$$

La quale invertita risulta:

$$\dot{\boldsymbol{\alpha}} = \mathbf{M}_1^{-1}(\boldsymbol{\alpha}) [\boldsymbol{\omega} - \mathbf{M}_2(\boldsymbol{\alpha}) \boldsymbol{\omega}_O] \quad (2.25)$$

È possibile dimostrare che per grandi rotazioni, il determinante della matrice $\mathbf{M}_1(\boldsymbol{\alpha})$ possa diventare nullo, e pertanto l'inversione di (2.24) risulti impossibile, portando a non poter risolvere (2.25) per trovare gli angoli di Eulero.

Per risolvere questa singolarità, risulta più semplice e sicuro scrivere le equazioni della cinematica utilizzando la struttura algebrica dei quaternioni. Un quaternion è definito come:

$$\mathbf{q} = \begin{pmatrix} q_0 \\ \vec{q} \end{pmatrix} \quad (2.26)$$

dove $\vec{q} = (q_1, q_2, q_3)^T$ e \mathbf{q} è un vettore a norma unitaria tale per cui $q_0^2 + \vec{q} \cdot \vec{q} = 1$. L'equazione della cinematica scritta in funzione dei quaternioni è priva di singolarità ed è definita nel seguente modo:

$$\dot{\mathbf{q}} = \frac{1}{2} \mathbf{Q}(\boldsymbol{\omega}) \mathbf{q} = \frac{1}{2} \begin{bmatrix} 0 & -\omega_x & -\omega_y & -\omega_z \\ \omega_x & 0 & \omega_z & -\omega_y \\ \omega_y & -\omega_z & 0 & \omega_x \\ \omega_z & \omega_y & -\omega_x & 0 \end{bmatrix} \mathbf{q} \quad (2.27)$$

Integrando queste equazioni, si trova la matrice di rotazione e i conseguenti angoli di Eulero. Definita $\mathbf{A}(\mathbf{q}) = \begin{bmatrix} 0 & -q_3 & q_2 \\ q_3 & 0 & -q_1 \\ -q_2 & q_1 & 0 \end{bmatrix}$, è possibile dimostrare che la matrice di rotazione è pari a:

$$\mathbf{R} = \mathbf{1}_{3 \times 3} + 2\mathbf{A}(\mathbf{q}) + 2q_0\mathbf{A}(\mathbf{q}) \quad (2.28)$$

Mentre gli angoli di Eulero sono:

$$\begin{cases} \phi = \arctan \frac{2(q_0q_1 + q_2q_3)}{1 - 2(q_1^2 + q_2^2)} \\ \theta = \arcsin 2(q_0q_2 + q_1q_3) \\ \psi = \arctan \frac{2(q_0q_3 + q_1q_2)}{1 - 2(q_2^2 + q_3^2)} \end{cases} \quad (2.29)$$

2.3 Sistema di stereo visione

Come anticipato, il sensore utilizzato per la stima del moto è una stereo camera. Una stereo camera è la composizione di due camere, una destra ed una sinistra: l'utilizzo di queste due camere rende possibile la rilevazione della terza dimensione, e quindi è possibile ricostruire una scena. Facendo il parallelismo con l'essere umano, una stereo camera è il sistema oculare, dove le due camere sono i due occhi. Nel corso dei successivi paragrafi verranno dati i concetti di base su come si forma un'immagine digitale e come modellare dal punto di vista matematico il modello camera e successivamente il modello stereo camera.

2.3.1 Ricostruzione di un immagine digitale

Come primo aspetto, bisogna capire come effettivamente funzioni una camera, ovvero cosa rileva. Le camere utilizzate oggi sono prettamente di natura digitale. Esse utilizzano dei sensori chiamati CCD (Charge-Coupled Device) [10] [11], ovvero circuiti formati da griglie o righe di elementi semiconduttori che accumulano carica elettrica proporzionale all'intensità di radiazione luminosa irradiata. Questi elementi sono accoppiati in maniera tale che la carica elettrica possa essere trasferita all'elemento adiacente. Gli impulsi elettrici vengono poi inviati al dispositivo che è in grado di ricostruire la matrice dei pixel, la quale origine è posta convenzionalmente nell'angolo in alto a sinistra, e dunque a ricostruire l'immagine.

Per quanto riguarda l'uomo, l'occhio riceve degli input di radiazione luminosa che vengono elaborati dal cervello. Le camere digitali funzionano allo stesso modo, solo che il sensore occhio viene sostituito dal sensore CCD e il cervello viene sostituito da una matrice che elabora l'input di radiazione luminosa.



Figura 2.4: Immagine digitale di un satellite target acquisita in laboratorio.

Le matrici che caratterizzano le immagini digitali in generale sono tre, in cui ogni matrice fornisce i valori di intensità luminosa per i tre colori primari: rosso,

verde e blu. Queste sono dette *matrici RGB* che combinate insieme producono le immagini. La dimensione di ogni matrice è strettamente legata alla dimensione dell'immagine digitale, ovvero all'altezza e alla larghezza in pixel. Se si ha un'immagine in scala di grigi, la matrice è soltanto una. Il valore di intensità luminosa che viene attribuito ad un pixel varia convenzionalmente tra 0 e 255, dove 0 corrisponde ad un'immagine completamente nera mentre 255 corrisponde ad un'immagine completamente bianca. Nella Fig. 2.4 è possibile osservare un'immagine acquisita durante l'attività sperimentale in laboratorio secondo una scala di grigi. I valori di intensità luminosa dell'immagine dei primi 10×10 pixel sono rappresentati nella Tab. 2.1.

12	12	11	11	11	11	11	11	12	12	...
10	10	12	12	11	11	11	11	13	12	...
8	10	10	9	11	11	11	10	13	12	...
9	9	11	11	11	11	11	11	11	11	...
15	15	15	15	15	15	15	15	15	15	...
16	16	17	17	17	17	17	17	17	17	...
18	17	16	16	16	16	16	16	16	16	...
18	18	15	16	16	16	16	16	16	15	...
16	16	17	17	17	17	17	17	17	17	...
16	16	18	17	17	17	17	17	17	17	...
...

Tabella 2.1: Valori di intensità luminosa dell'immagine dei primi 10×10 pixel della (Fig. 2.4).

Le immagini in input vengono quindi rappresentate da questa tipologia di matrici. Nel corso dei successivi paragrafi e capitoli verranno spiegate le relazioni matematiche utilizzate per descrivere i modelli della camera e come queste matrici vengano utilizzate per riconoscere delle features interessanti del target, come per esempio nella Fig. 2.4.

2.3.2 Modello Pinhole camera ideale

Il modello di camera utilizzato è quello del "Pinhole Camera Model" [12]. L'ipotesi fondamentale che sta a monte è che la lente che forma la camera sia molto sottile, dunque, le equazioni che ne descrivono il comportamento sono notevolmente più semplici. È importante sottolineare come questo modello non tenga conto di

fenomeni ottici come la distorsione: essa verrà eliminata tramite la calibrazione, i quali algoritmi saranno introdotti nel capitolo 3.

Si prenda la Fig. 2.5 rappresentante la schematizzazione del modello Pin-hole camera. Siano:

- (X, Y, Z) il sistema di riferimento centrato nella camera, in cui l'asse Z coincide con l'asse ottico, mentre il piano (X, Y) è parallelo al piano immagine;
- (u, v) il sistema di riferimento, espresso in pixel, la cui origine è per convenzione l'estremo in alto a sinistra;
- (x, y) il sistema di riferimento centrato nel piano immagine le cui coordinate pixel sono $(c_x, c_y)^T$, distanziato della lunghezza focale f dal centro del sistema di riferimento della camera.

Si prenda il punto $P = (X, Y, Z)^T$ la cui proiezione sul piano immagine è $p = (u, v)^T$. Tramite la semplice similitudine di triangoli rettangoli è possibile scrivere:

$$\begin{pmatrix} x \\ y \end{pmatrix} = -\frac{f}{Z} \begin{pmatrix} X \\ Y \end{pmatrix} \quad (2.30)$$

Che può essere scritta in coordinate omogenee:

$$Z \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{bmatrix} -f & 0 & 0 \\ 0 & -f & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \quad (2.31)$$

Il legame tra le coordinate espresse nel sistema di riferimento camera (X, Y, Z) e le coordinate espresse nel sistema di riferimento mondo (X_0, Y_0, Z_0) , è dato da:

$$\begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix} \begin{pmatrix} X_0 \\ Y_0 \\ Z_0 \\ 1 \end{pmatrix} \quad (2.32)$$

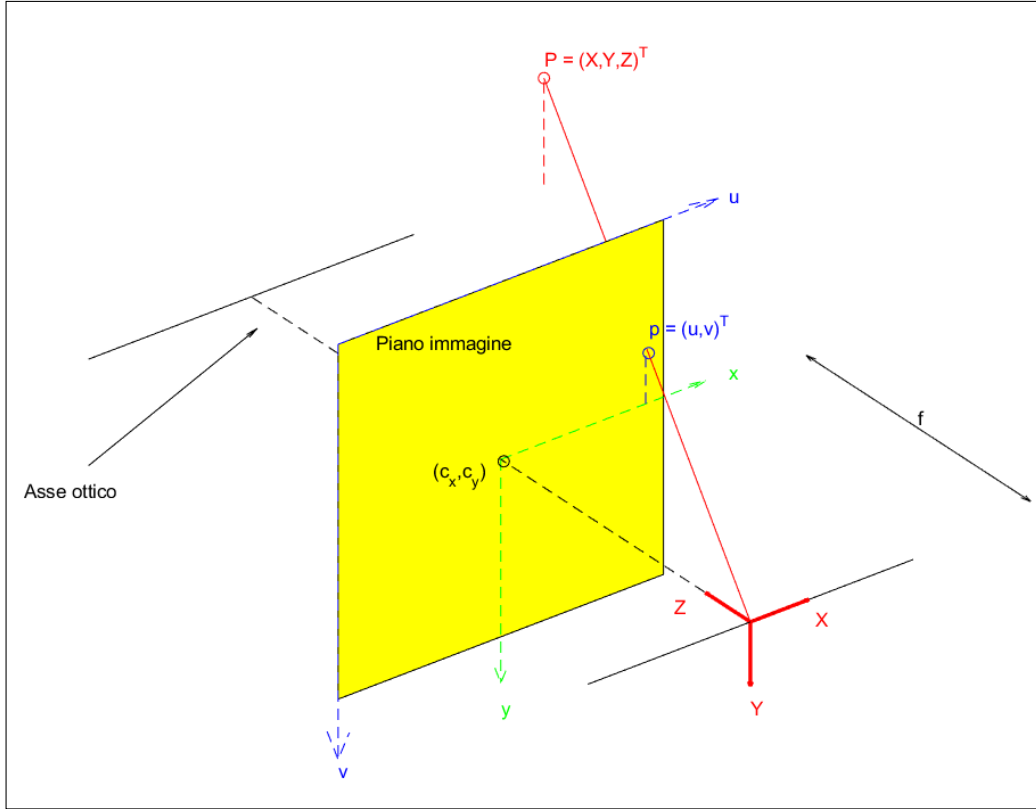


Figura 2.5: Modello camera Pinhole.

dove \mathbf{R} è la matrice di rotazione e \mathbf{t} è il vettore traslazione, mentre la matrice completa è detta matrice di rototraslazione tra coordinate mondo e coordinate camera. Sostituendo (2.32) in (2.31):

$$Z \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{bmatrix} -f & 0 & 0 \\ 0 & -f & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix} \begin{pmatrix} X_0 \\ Y_0 \\ Z_0 \\ 1 \end{pmatrix} \quad (2.33)$$

Il sistema di visione restituisce le coordinate dei punti di interesse nel sistema coordinate pixel (u, v) , e pertanto bisogna fare un ultimo passaggio. È possibile introdurre la matrice dei parametri intrinseci \mathbf{K} definita come:

$$\mathbf{K} = \begin{bmatrix} -S_x & -S_\theta & c_x \\ 0 & -S_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \quad (2.34)$$

dove S_x , S_y e S_θ sono fattori di scala espressi in $[\frac{pixel}{m}]$, c_x e c_y sono i centri ottici espressi in $[pixel]$ e f è la distanza focale espressa in $[m]$. Tutti questi parametri possono essere estrapolati attraverso un'opportuna calibrazione. La matrice dei parametri intrinseci è tale per cui:

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \mathbf{K} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \quad (2.35)$$

Data l'introduzione della matrice dei parametri intrinseci (2.34) ed il legame con le coordinate metriche espresso in (2.35), è possibile sostituire tutto all'interno di (2.33). Per dare una notazione più compatta, si introducono prima le seguenti matrici:

$$\begin{aligned} \mathbf{K} &= \begin{bmatrix} f_x & f_\theta & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \\ \mathbf{A} &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \\ \mathbf{T} &= \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix} \end{aligned} \quad (2.36)$$

La coordinata Z in un sistema monoculare è ignota: è possibile porre $\lambda = Z$ come fattore di scala. Infine, sostituendo il tutto le matrici (2.36) in (2.33), può essere finalmente definita l'equazione fondamentale del modello pinhole:

$$\lambda \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \mathbf{KAT} \begin{pmatrix} X_0 \\ Y_0 \\ Z_0 \\ 1 \end{pmatrix} \quad (2.37)$$

Si osserva come (2.37) sia definito a meno di una costante λ , che tramite un'unica

camera o con un'unica osservazione risulta sconosciuta e deve essere calcolata.

2.3.3 Stereo camera

Nel paragrafo 2.3.2 è stato introdotto il modello pinhole camera ideale che viene utilizzato, con tutte le equazioni che lo riguardano. Dopo aver ricavato l'equazione fondamentale (2.37), si è riscontrata la problematica data dall'incognita λ correlata alla terza dimensione. Per risolvere questo può essere utilizzato un sistema a due camere, ovvero una stereo camera.

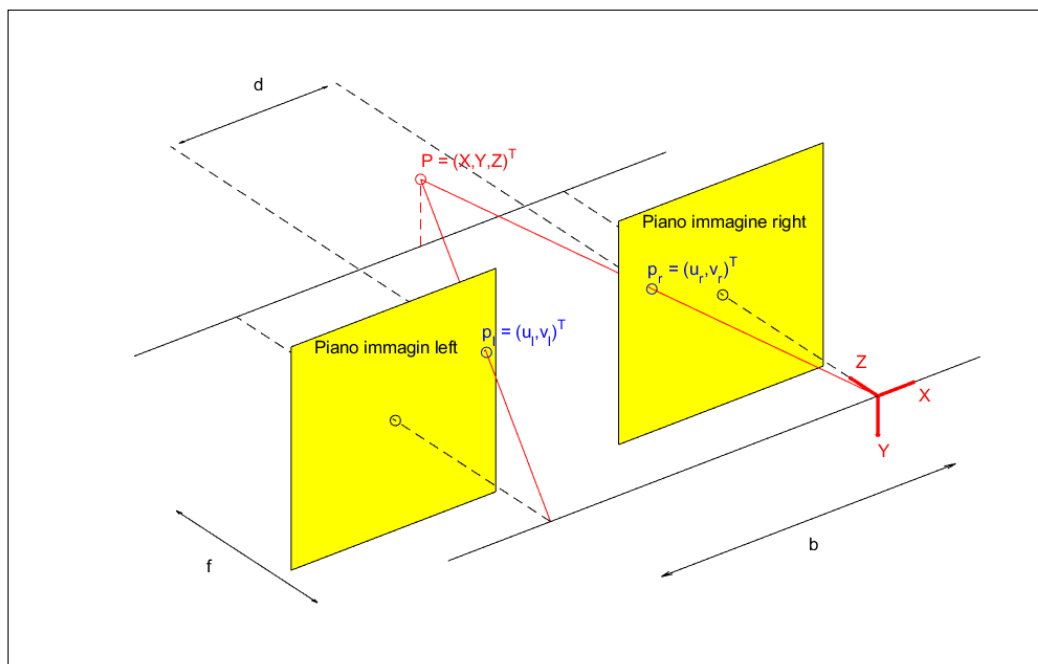


Figura 2.6: Modello stereo camera.

Sia la schematizzazione del modello di una stereo camera ad assi paralleli osservabile nella Fig. 2.6. Sia un punto $P = (X, Y, Z)^T$, ove le coordinate sono espresse nel sistema di riferimento di una camera. In questo caso, il sistema di coordinate è espresso nel seguente modo:

- sistema di riferimento centrato nella camera destra;
- asse X lungo la congiungente tra la due camera, diretto dalla camera sinistra a quella destra e centrato nel pinhole della camera destra;
- asse Z lungo l'asse ottico;

- asse Y chiude la terna.

I centri ottici sono separati solamente lungo l'asse X da una quantità b (baseline). I piani immagini delle due camere si formano alla stessa distanza focale f dal centro della camera. In questa maniera, la proiezione del punto P sui piani immagini ha la stessa coordinata v ($v_l = v_r$), ma differente coordinata u ($u_l \neq u_r$). Questa differenza d è detta disparity ($d = u_l - u_r$), ed è fondamentale la sua conoscenza per la stima della terza dimensione. Per semplice similitudine di triangoli rettangoli, nota f dalla calibrazione, nota b dai parametri costruttivi e calcolata d , è possibile scrivere:

$$Z = \frac{bf}{d} \tag{2.38}$$

Da (2.38), Siccome non è facile calcolare d molto piccoli, se si vuole avere una profondità Z massima calcolata piuttosto alta, occorre aumentare la b o f .

Capitolo 3

Analisi delle immagini

Nei paragrafi 2.3.2 e 2.3.3 sono stati illustrati rispettivamente i modelli matematici di Pinhole di una camera monoculare e di un sistema di stereo visione. In questo capitolo si analizzerà come applicare tale sistema alla determinazione del moto, in particolare quali siano gli output da estrarre. A prescindere da quale sia l'obiettivo di un sistema di stereo visione, le operazioni che devono essere svolte seguono uno schema standard visibile nella Fig. 3.1:

- *Calibrazione della stereo camera*: ottenimento dei parametri intrinseci introdotti nel paragrafo 2.3.2 per entrambe le camere. Questo passaggio permette anche di ottenere immagini rettificate, ovvero depurarle dai fenomeni di distorsione;
- *Ricerca delle features*: camera sinistra e destra ricercano in maniera indipendente dei particolari tipi di features, ovvero delle figure di interesse (spigoli, superfici, bordi, etc.), sul proprio piano immagine;
- *Matching delle features*: le features indipendenti trovate dalla camera sinistra e da quella destra vengono accoppiate;
- *Rimozione degli outliers*: possono esserci delle features al di fuori del cosiddetto piano epipolare, che devono essere eliminate;
- *Selezione delle features*: in funzione del tipo di studio, vengono selezionate solo alcuni tipi di features. Per esempio, nel caso di moto relativo stazionario,

si tende a voler avere sempre le stesse features in tutta la scansione e quindi bisogna selezionare solo le features che vengono osservate in ogni frame. Per lo scenario rotazionali ciò non avviene e si utilizzano altre tecniche che verranno spiegate in seguito;

- *Triangolazione delle features*: le features rimanenti vengono triangolate per ricavare le coordinate 3D espresse nel sistema di riferimento (X, Y, Z) introdotto nel paragrafo 2.3.3.

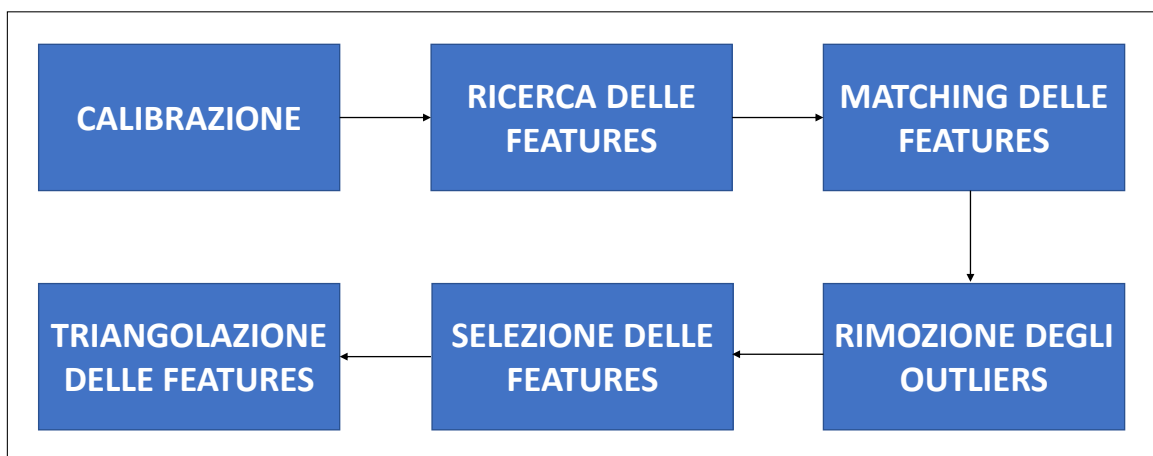


Figura 3.1: Schema procedurale dell'analisi delle features.

3.1 Calibrazione della stereo camera

La calibrazione ricopre un ruolo fondamentale nel corso dell'analisi delle immagini. Senza la corretta conoscenza dei parametri intrinseci, ovvero la lunghezza focale f , la baseline b , e la matrice \mathbf{K} introdotta nel paragrafo 2.3.2 non è possibile scrivere in maniera completa le equazioni. In più, non risulterebbe pure possibile la rettificazione delle immagini, ovvero la rimozione degli effetti dovuti a distorsioni ottiche, che pregiudica la corretta applicazione degli step successivi.

La distorsione, osservabile nella Fig. 3.2, può essere rappresentata dal punto di vista matematico in diversi modi. I più utilizzati sono polinomi, trovabili in letteratura [13]. Definito (x, y) il sistema di riferimento ideale, il sistema di riferimento $(x_{d,r}, y_{d,r})$ dovuto alla distorsione radiale può essere scritto come:

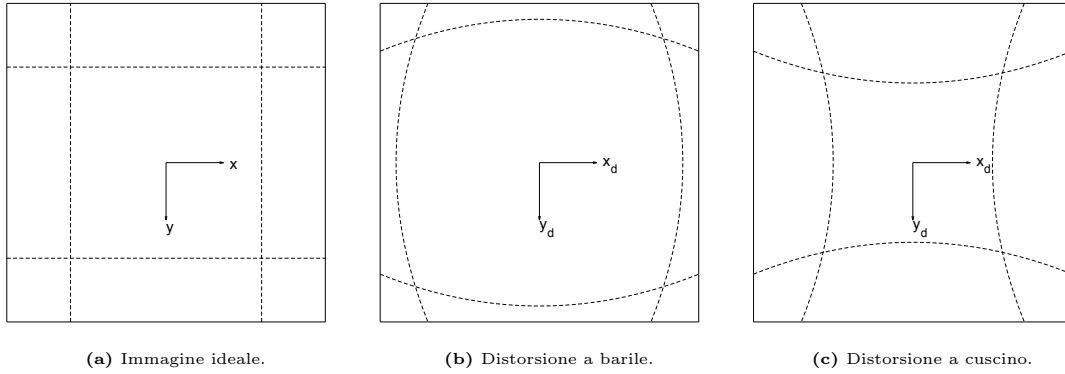


Figura 3.2: Rappresentazione dei diversi tipi di distorsione.

$$\begin{cases} x_{d,r} = x \left[1 + k_1 (x^2 + y^2) + k_2 (x^2 + y^2)^2 + k_5 (x^2 + y^2)^3 \right] \\ y_{d,r} = y \left[1 + k_1 (x^2 + y^2) + k_2 (x^2 + y^2)^2 + k_5 (x^2 + y^2)^3 \right] \end{cases} \quad (3.1)$$

Mentre il sistema di riferimento $(x_{d,t}, y_{d,t})$ dovuto alla distorsione tangenziale può essere scritto come:

$$\begin{cases} x_{d,t} = 2k_3xy + k_4(3x^2 + y^2) \\ y_{d,t} = k_3(x^2 + 3y^2) + 2k_4xy \end{cases} \quad (3.2)$$

Il sistema di riferimento distorto complessivo (x_d, y_d) è quindi dato da:

$$\begin{cases} x_d = x_{d,r} + x_{d,t} \\ y_d = y_{d,r} + y_{d,t} \end{cases} \quad (3.3)$$

ove $\mathbf{k} = (k_1, k_2, k_3, k_4, k_5)$ è il vettore dei coefficienti di distorsione.

Esistono diversi metodi per eseguire la calibrazione. In MATLAB è esistente un toolbox che permette di eseguirla in maniera automatica tramite l'algoritmo di Zhang [14]. Questo algoritmo si basa sull'analisi di una scacchiera disposta a diverse distanze ed angolazioni rispetto alla camera, il cui sistema di riferimento ed un esempio di diversa disposizione della scacchiera sono presenti nella Fig. 3.3.

Siano date n immagini di una scacchiera ripresa da posizioni ed angolazioni diverse. La calibrazione con l'algoritmo di Zhang prevede, quindi, il calcolo di n

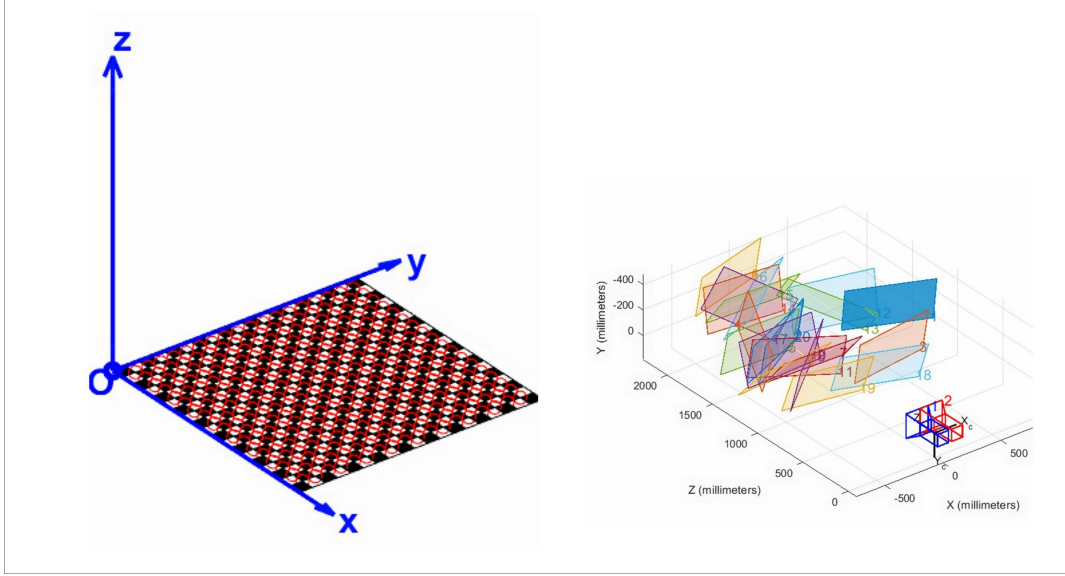


Figura 3.3: Rappresentazione della scacchiera per la calibrazione ripresa in più pose.

matrici omografiche \mathbf{H} ottenute dalle n immagini di osservazione del piano. Le matrici \mathbf{H} possono essere scritte come:

$$\mathbf{H} = \begin{bmatrix} \mathbf{h}_1 & \mathbf{h}_2 & \mathbf{h}_3 \end{bmatrix} = \lambda \mathbf{K} \begin{bmatrix} \mathbf{r}_1 & \mathbf{r}_2 & \mathbf{t} \end{bmatrix} \quad (3.4)$$

Nonostante il fattore λ non sia noto, siccome i vettori \mathbf{r}_1 ed \mathbf{r}_2 devono essere ortogonali, (3.4) deve rispettare seguente il vincolo:

$$\begin{cases} \mathbf{r}_1^T \mathbf{r}_2 = 0 \\ \mathbf{r}_1^T \mathbf{r}_1 = \mathbf{r}_2^T \mathbf{r}_2 \end{cases} \implies \begin{cases} \mathbf{h}_1^T \mathbf{B} \mathbf{h}_2 = 0 \\ \mathbf{h}_1^T \mathbf{B} \mathbf{h}_1 = \mathbf{h}_2^T \mathbf{B} \mathbf{h}_2 \end{cases} \quad (3.5)$$

ove la matrice \mathbf{B} è definita come:

$$\mathbf{B} = (\mathbf{K}^{-1})^T \mathbf{K}^{-1} \quad (3.6)$$

Se per semplicità di trattazione viene tralasciato lo skew (S_θ) e ponendo gli altri parametri $fS_x = f_x$ e $fS_y = f_y$, (3.6) può essere riscritta:

$$\mathbf{B} = \begin{bmatrix} \frac{1}{f_x^2} & 0 & -\frac{c_x}{f_x^2} \\ 0 & \frac{1}{f_y^2} & -\frac{c_y}{f_y^2} \\ -\frac{c_x}{f_x^2} & -\frac{c_y}{f_y^2} & \left(\frac{c_x}{f_x}\right)^2 + \left(\frac{c_y}{f_y}\right)^2 + 1 \end{bmatrix} \quad (3.7)$$

Le incognite della matrice \mathbf{B} possono essere risolte mediante l'osservazione di minimo due piani, oppure tre se viene considerato anche lo skew. Si osserva che la matrice \mathbf{B} è simmetrica: nel caso di (3.7) le incognite sono 5, mentre se si tenesse conto anche dello skew le incognite sarebbero 6. Pertanto, la matrice \mathbf{B} può essere espressa come un vettore di incognite:

$$\mathbf{b} = (B_{11}, B_{12}, B_{22}, B_{13}, B_{23}, B_{33})^T \quad (3.8)$$

L' i -esimo vettore \mathbf{h}_i è:

$$\mathbf{h}_i = (h_{i1}, h_{i2}, h_{i3})^T \quad (3.9)$$

Introducendo un vettore ausiliario \mathbf{v}_{ij} tale per cui:

$$\mathbf{v}_{ij} = \begin{pmatrix} h_{i1}h_{j1} \\ h_{i1}h_{j2} + h_{i2}h_{j1} \\ h_{i2}h_{j2} \\ h_{i3}h_{j1} + h_{i1}h_{j3} \\ h_{i3}h_{j2} + h_{i2}h_{j3} \\ h_{i3}h_{j3} \end{pmatrix} \quad (3.10)$$

e sostituendo (3.8), (3.9) e (3.10) all'interno dei due vincoli esplicitati in (3.5), dopo qualche passaggio risulta che:

$$(\mathbf{v}_{11} - \mathbf{v}_{22})^T \mathbf{b} = 0 \quad (3.11)$$

Pertanto, ogni immagine restituisce 2 equazioni nei 6 parametri indipendenti di \mathbf{B} , la cui soluzione del sistema lineare è risolvibile attraverso un metodo ai minimi quadrati. Una volta determinate le incognite di \mathbf{B} , quindi determinata la matrice dei parametri intrinseci \mathbf{K} , invertendo (3.4) è possibile ricavare i parametri della rototraslazione tra le due camere, in cui il versore \mathbf{r}_3 si trova dall'ortogonalità tra i versori \mathbf{r}_1 e \mathbf{r}_2 .

Calcolati tutti questi parametri è possibile trovare anche quelli riguardanti la distorsione attraverso la minimizzazione dell'errore di riproiezione:

$$\min \sum_i \sum_j \| \mathbf{x}'_{d_{i,j}} - \mathbf{x}'_{d_{i,j}} (\mathbf{K}, \mathbf{R}, \mathbf{t}, \mathbf{k}, \mathbf{W}_i^j) \|^2 \quad (3.12)$$

ove $\mathbf{x}'_{d_{i,j}}$ viene misurata sulle immagini, mentre \mathbf{W}_i^j viene misurata sui quadrati della scacchiera.

Una volta calibrata la stereo camera, è possibile rettificare le immagini, ovvero depurarle dalla distorsione: in tal senso un nuovo sistema di riferimento viene introdotto.

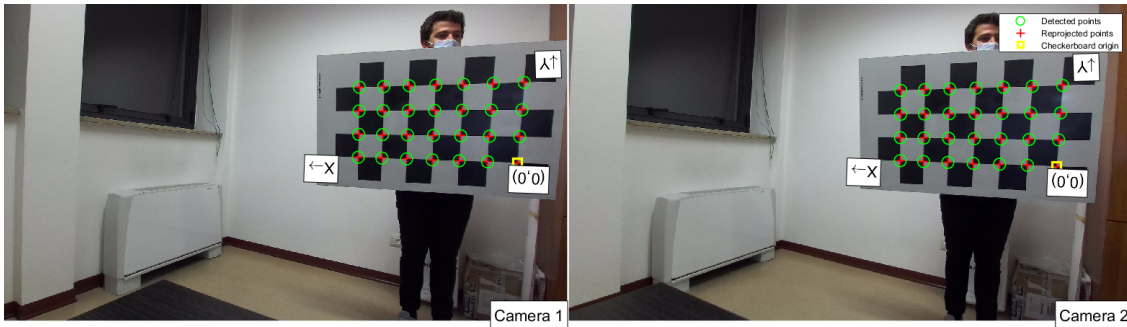


Figura 3.4: Scacchiera distorta pre-calibrazione.

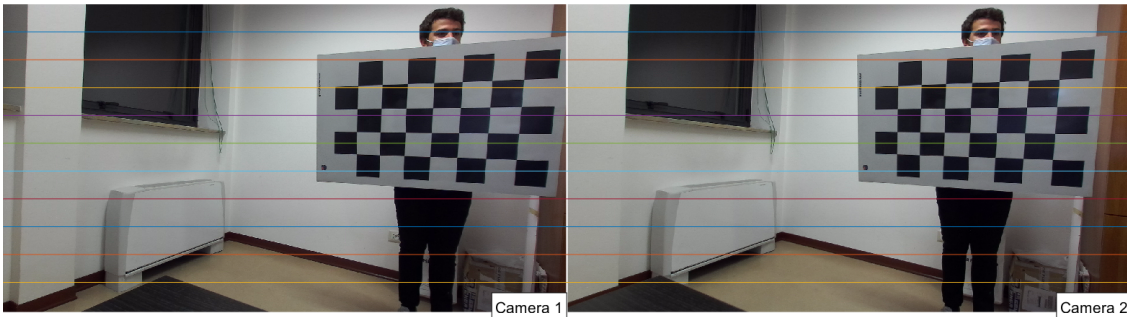


Figura 3.5: Scacchiera rettificata post-calibrazione.

3.2 Ricerca delle features

Una volta calibrato il sistema di visione e rettificato le immagini, è possibile procedere con la ricerca delle features, per entrambe le camere. Nel paragrafo 1.1 è stato spiegato che l'obiettivo della tesi è la determinazione del moto relativo di un satellite non cooperativo, in cui la posizione delle features trovate non è nota in coordinate target, e quindi non è possibile dedurre direttamente il moto relativo.

A ragione di ciò, la stereo camera deve ricercare delle features che possano essere note, quali spigoli, bordi, cerchi o antenne.

Esistono molteplici *detector* e *descriptor* per la ricerca delle features di interesse nei piani immagini delle camere. È bene sottolineare come l'immagine che ogni detector o descriptor deve analizzare deve essere espresso in una gradazione di grigi e non in RGB: questo è dato dal fatto che tutti questi si basano sulla valutazione dell'intensità luminosa (come visto nel paragrafo 2.3.1), e pertanto la conversione risulta necessaria. Inoltre, non tutto il piano immagine risulta di interesse. Per questo motivo, se i pixel da considerare per la ricerca delle features sono noti, è molto utile selezionare solo quella zona del piano immagine detta ROI (*Region of Interest*).

Tra i diversi detector e descriptor verranno citati quelli che sono più utilizzati.

3.2.1 Harris-Stephens

Questo è un *corner detector* [15], ovvero un detector adibito a rilevare spigoli. Esso si basa sulla variazione di intensità luminosa $I(x, y)$ in una determinata area di pixel. La funzione gradiente $E(\Delta x, \Delta y)$ può essere scritta come:

$$E(\Delta x, \Delta y) = \sum_{x,y} [I(x + \Delta x, y + \Delta y) - I(x, y)]^2 \quad (3.13)$$

Se Δx e Δy sono molto piccoli, la funzione incrementale all'interno della sommatoria può essere approssimata in serie di Taylor come:

$$I(x + \Delta x, y + \Delta y) \approx I(x, y) + \frac{\partial I(x, y)}{\partial x} \Delta x + \frac{\partial I(x, y)}{\partial y} \Delta y \quad (3.14)$$

Sostituendo (3.14) in (3.13) risulta che:

$$E(\Delta x, \Delta y) = \begin{pmatrix} \Delta x & \Delta y \end{pmatrix} \mathbf{M} \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} \quad (3.15)$$

ove la matrice \mathbf{M} è definita come:

$$\mathbf{M} = \sum_{x,y} \begin{bmatrix} \frac{\partial^2 I(x,y)}{\partial x^2} & \frac{\partial^2 I(x,y)}{\partial x \partial y} \\ \frac{\partial^2 I(x,y)}{\partial x \partial y} & \frac{\partial^2 I(x,y)}{\partial y^2} \end{bmatrix} \quad (3.16)$$

La natura degli autovalori α e β della matrice \mathbf{M} rappresentano il comportamento della feature trovata. In particolare:

- $\alpha \gg \beta$ o $\alpha \ll \beta$: si è in corrispondenza di uno spigolo;
- $\alpha \approx \beta \approx 0$: si è in corrispondenza di una regione piatta;
- $\alpha, \beta \ll 0a$: si è in corrispondenza di uno spigolo.

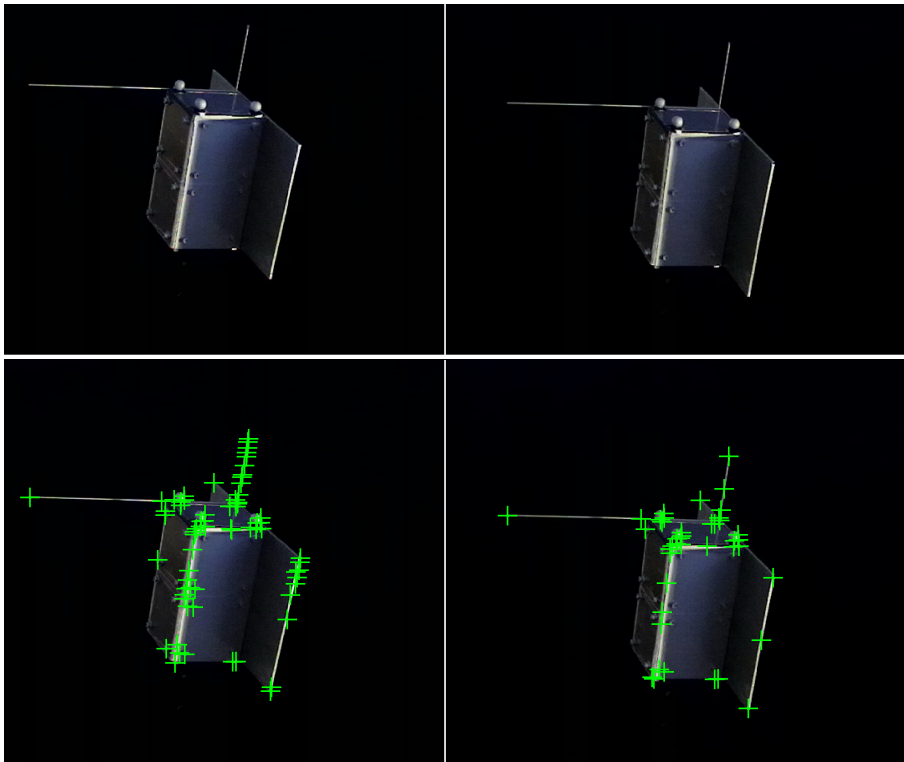


Figura 3.6: Features trovate dal detector Harris-Stephens.

3.2.2 SURF (Speeded up robust features)

Questo è un *blob detector* [16], ovvero un detector adibito a rilevare regioni quando vengono apprezzate variazioni dell'intensità luminosa tra regioni limitrofe.

Il vantaggio di questo detector è la invarianza per scala e rotazione, rendendolo molto robusto ed affidabile, quindi idoneo ad applicazioni real time. Si basa sul Fast Hessian detector. La matrice Hessiana \mathbf{H} per ogni punto $\mathbf{x} = (x, y)$ è:

$$\mathbf{H} = \sum_{x,y} \begin{bmatrix} L_{xx}(\mathbf{x}, \boldsymbol{\sigma}) & L_{xy}(\mathbf{x}, \boldsymbol{\sigma}) \\ L_{xy}(\mathbf{x}, \boldsymbol{\sigma}) & L_{yy}(\mathbf{x}, \boldsymbol{\sigma}) \end{bmatrix} \quad (3.17)$$

ove $L_{xx}(\mathbf{x}, \boldsymbol{\sigma})$ è la convoluzione della derivata seconda $\frac{\partial^2 g(\boldsymbol{\sigma})}{\partial x^2}$ della gaussiana di intensità I in ogni punto \mathbf{x} , con $L_{xy}(\mathbf{x}, \boldsymbol{\sigma})$ e $L_{yy}(\mathbf{x}, \boldsymbol{\sigma})$ definiti in maniera uguale. Il fine della convoluzione è quello di minimizzare il rumore, rendendo quindi questo detector robusto ed affidabile.

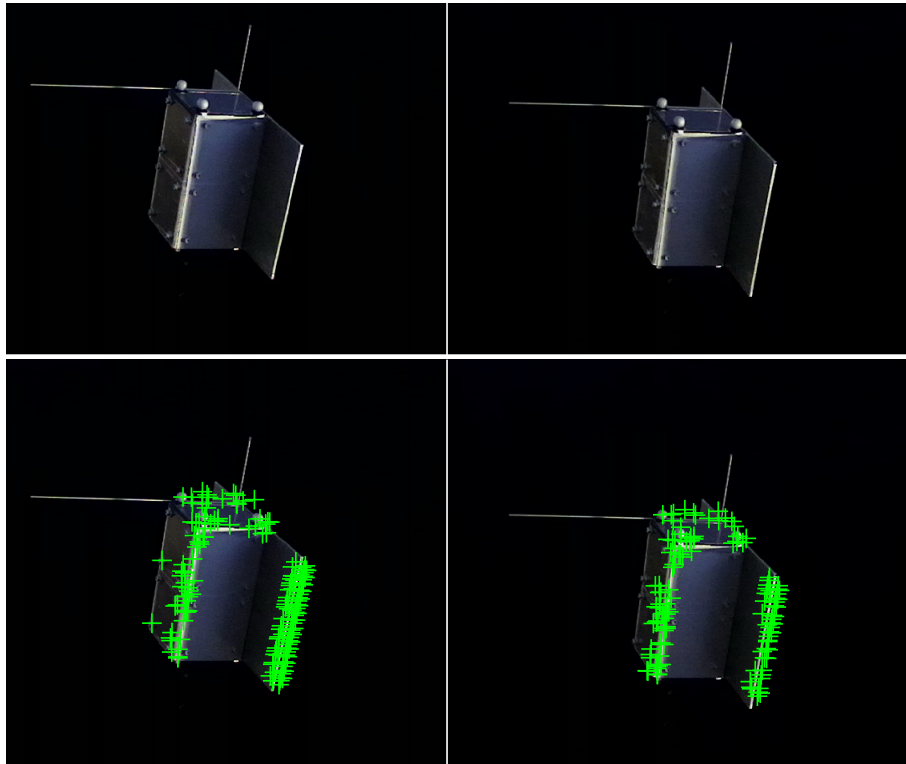


Figura 3.7: Features trovate dal detector SURF.

3.3 Matching delle features

Una volta trovate le features per entrambe le camere, occorre eseguire il matching per selezionare solamente le features comuni [17]. Infatti, in generale,

camera destra e camera sinistra troveranno un certo numero di features di diverse coordinate (x, y) sui due piani immagine, ma non tutte sono comuni ad entrambe le camere. È buona norma utilizzare la stessa tipologia di detector per entrambe le camere, perché altrimenti il matching non porta a risultati oppure porta a risultati sbagliati.

Il primo passo per il matching è valutare la distanza tra tutti gli n descrittori. Indicando come \mathbf{f}_1 il descrittore della camera sinistra e \mathbf{f}_2 il descrittore della camera destra, esistono due diverse tipologie di distanza:

$$SDA = \sum_{i=1}^n |\mathbf{f}_1(i) - \mathbf{f}_2(i)| \quad (3.18)$$

$$SDQ = \sum_{i=1}^n [\mathbf{f}_1(i) - \mathbf{f}_2(i)]^2 \quad (3.19)$$

dove (3.18) rappresenta la somma delle differenze assolute mentre (3.19) rappresenta la somma delle differenze al quadrato. Per ogni coppia di descrittori, viene calcolata la distanza con tutti gli altri. La coppia di descrittori più vicina viene considerata come corrispondenza se viene soddisfatta una certa soglia, come per esempio un valore numerico su massima distanza tollerabile. Se un descrittore viene associato a più coppie di descrittori, questo genera un'ambiguità che viene risolta rifiutando quelle corrispondenze. Nella Fig. 3.8 e nella Fig. 3.9 è possibile osservare il matching tra le features trovate rispettivamente con Harris-Stephens e con SURF. Si può notare la selezione di una ROI nella ricerca per selezionare solo le features appartenenti al target.

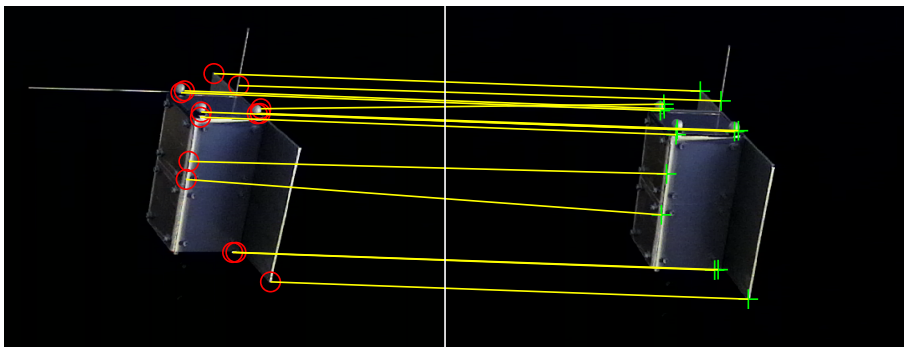


Figura 3.8: Matching tra descrittori trovati con Harris-Stephens.

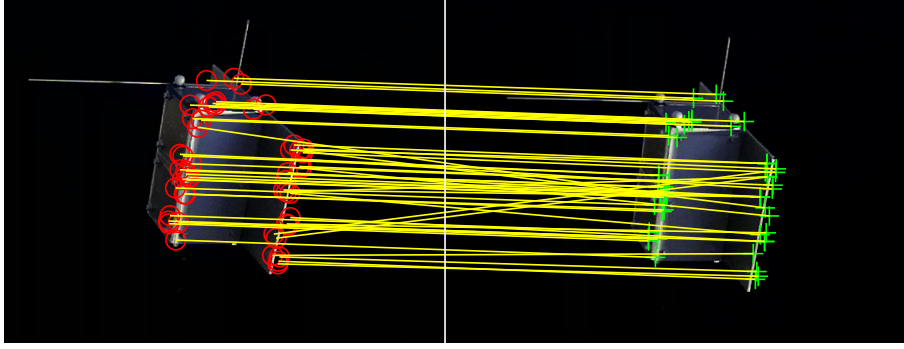


Figura 3.9: Matching tra descrittori trovati con SURF.

3.4 Rimozione degli outliers

Nonostante il matching inteso come valutazione delle distanze porti a rifiutare delle corrispondenze che sono fonte di errori ed ambiguità, un altro passo è necessario per correggere ulteriori corrispondenze di features che non rispettano il vincolo epipolare, ovvero features che non rispettano la riproiezione del punto nell'ambiente. Esse possono generare falsi matching come osservabile nelle Fig.3.8 e Fig.3.9. Gli outliers vengono stimati applicando il filtraggio RANSAC (RANDOM SAMPLE CONSENSUS) [18]. Essendo noti i parametri intrinseci della matrice \mathbf{K} , occorre stimare la matrice fondamentale \mathbf{F} che rispetta la seguente equazione:

$$\mathbf{x}_2^T \mathbf{F} \mathbf{x}_1 = 0 \quad (3.20)$$

ove $\mathbf{x}_1 = (u_1, v_1, 1)^T$ e $\mathbf{x}_2 = (u_2, v_2, 1)^T$, mentre la matrice \mathbf{F} è:

$$\mathbf{F} = \begin{bmatrix} f_1 & f_2 & f_3 \\ f_4 & f_5 & f_6 \\ f_7 & f_8 & f_9 \end{bmatrix} \quad (3.21)$$

\mathbf{F} viene stimata attraverso il filtraggio RANSAC a partire da una serie di inliers casuali su cui vengono testate le corrispondenze e restituendo il modello di fitting più adatto. Essendo \mathbf{F} una matrice con 7 (e non 9) gradi di libertà, è necessaria la presenza di minimo 7 corrispondenze per risolvere il sistema lineare (3.20), che esplicitato diventa:

$$f_1 u_1 u_2 + f_2 u_1 v_2 + f_3 u_1 + f_4 v_1 u_2 + f_5 v_1 v_2 + f_6 v_1 + f_7 u_2 + f_8 v_2 + f_9 = 0 \quad (3.22)$$

Assumendo di interpolare il vettore di incognite \mathbf{f}_i con un insieme di punti \mathbf{z}_i , il best fitting viene ottenuto minimizzando la seguente funzione costo:

$$\min \sum_{i=1}^n (\mathbf{f}_i^T \mathbf{z}_i)^2 \quad (3.23)$$

in cui l'iesimo vettore \mathbf{z}_i è:

$$\mathbf{z}_i = (u_1 u_2, u_1 v_2, u_1, v_1 u_2, v_1 v_2, v_1, u_2, v_2, 1) \quad (3.24)$$

Da quest'ultima equazione è possibile definire la matrice \mathbf{Z} le cui righe sono le \mathbf{z}_i , quindi una matrice la cui dimensione dipende dal numero di corrispondenze prese in considerazione. La matrice \mathbf{Z} viene successivamente centrata nell'origine, sottraendo ad ogni riga il valore medio. Quindi, è possibile ricavare la matrice \mathbf{M} :

$$\mathbf{M} = \mathbf{Z}^T \mathbf{Z} \quad (3.25)$$

Si possono calcolare gli autovalori λ_i e i corrispondenti autovettori \mathbf{w}_i della matrice \mathbf{M} : il best fitting viene dato dai più piccoli λ_i e quindi \mathbf{w}_i risolvono (3.23). Nelle Fig. 3.10 e Fig. 3.11 si osservano gli inliers stimati rispettivamente con Harris-Stephens e SURF.

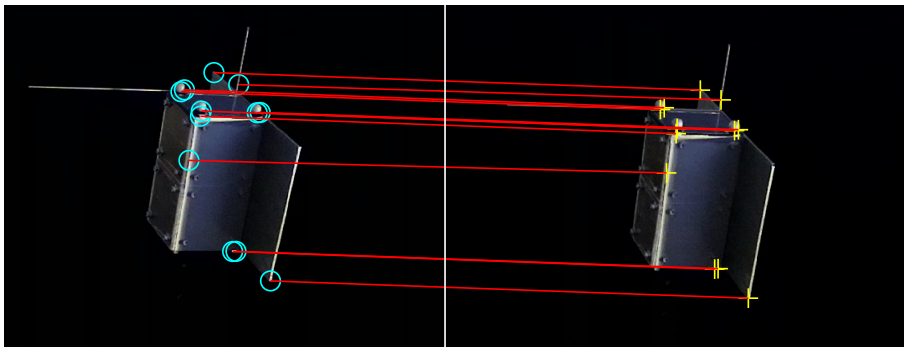


Figura 3.10: Inliers stimati con Harris-Stephens.

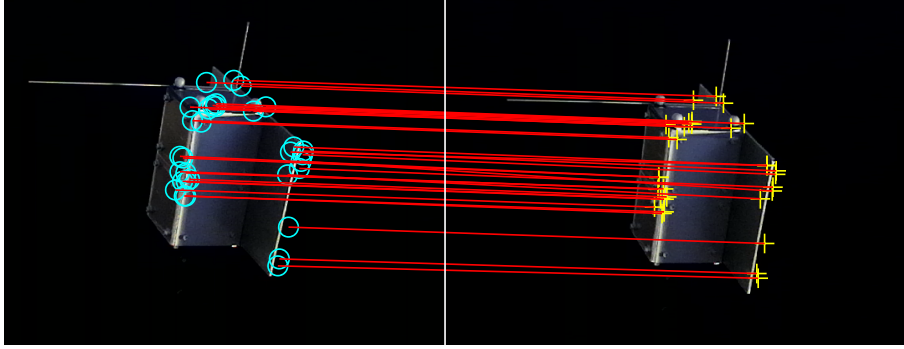


Figura 3.11: Inliers stimati con SURF.

3.5 Selezione delle features

Una volta trovate tutte le corrispondenze tra le features e una volta rimossi gli outliers, è possibile procedere con la selezione delle features. In tal senso, per gli scopi della tesi e per applicare il filtro di Kalman che sarà introdotto nel capitolo 5, è possibile avere diversi scenari.

Come si vedrà meglio più avanti, per un satellite non cooperativo, la posizione dell' i -esima feature sul target non è nota in coordinate target, e pertanto deve essere stimata. Per fare questo in ogni frame di acquisizione devono essere presenti sempre le stesse features. Se nel caso stazionario ciò è possibile, in uno scenario rotazionale non lo è e quindi bisogna adottare soluzioni diverse. I processi introdotti nei paragrafi 3.3 e 3.4 portano a trovare delle features comuni alle quali vengono associati degli indici, per la camera sinistra e destra. Possono essere distinti ora due approcci diversi in funzione dello scenario preso in considerazione:

- *Scenario stazionario.*

Questo è lo scenario in cui chaser e target mantengono sempre la stessa posa relativa, con velocità traslazionale ed angolare nulla, per tutta la durata dell'acquisizione. In questo caso, è possibile avere le stesse features in ogni frame, in quanto il target è fermo. Per fare ciò, la coppia di indici deve rimanere sempre la stessa, riducendo talvolta di molto il numero complessivo di features.

- *Scenario rotazionale.*

Questo è lo scenario in cui chaser e target mantengono sempre la stessa posi-

zione relativa, con velocità traslazionale nulla, ma varia l'assetto relativo con una certa velocità angolare. Per grandi rotazioni, come ad esempio una rivoluzione completa del target, risulta impossibile avere le stesse features per tutta la durata dell'acquisizione. In questo caso, vengono selezionate le coppie di indici comuni al frame i e al frame $i + 1$, che, come si vedrà più avanti, servirà a stimare la velocità delle features.

Il numero di features trovate può essere molto alto o molto basso, in funzione delle condizioni di luminosità e della tipologia di selezione effettuata. Avere un numero molto elevato di features può rallentare gli algoritmi risolutivi, ma un numero troppo basso può portare ad una stima del moto sbagliata; quindi, bisogna trovare un compromesso tra numero di features e velocità di calcolo.

3.6 Triangolazione delle features

Ottenute e selezionate le features di interesse, è possibile procedere con la triangolazione, ovvero ottenere le coordinate 3D delle features stesse espresse in coordinate camera.

In (2.38) si è visto come, note la baseline, la distanza focale e la disparity, è possibile calcolare la profondità. Quando si ha a che fare con più punti, questo approccio non è conveniente, e si devono utilizzare altri algoritmi come, ad esempio, la triangolazione tramite il Middle Point Algorithm [18].

Si suppone che il sistema stereo sia come quello della Fig. 2.6, con il sistema di riferimento (X, Y, Z) centrato nel centro della camera destra. Il matching permette di trovare le features in coordinate pixel: per passare in un sistema di coordinate metriche basta invertire (2.35), essendo note le matrici dei parametri intrinseci \mathbf{K} . Questi punti in coordinate metriche sono rispettivamente \mathbf{x}_1 e \mathbf{x}_2 per la camera sinistra e per la camera destra. Questi punti in coordinate $\mathbf{X} = (X, Y, Z)^T$ sono dati da:

$$\begin{cases} \mathbf{X}_1^1 = \lambda_1 \mathbf{x}_1 \\ \mathbf{X}_2^2 = \lambda_2 \mathbf{x}_2 \end{cases} \quad (3.26)$$

ove λ_1 e λ_2 sono dei coefficienti che devono essere determinati, mentre \mathbf{X}_1^1 e \mathbf{X}_2^2 sono i punti 3D espressi nei sistemi di riferimento della camera sinistra e destra. Portando (3.26), per esempio, nel sistema di riferimento della camera sinistra:

$$\begin{cases} \mathbf{X}_1^1 = \lambda_1 \mathbf{x}_1 \\ \mathbf{X}_2^1 = \lambda_2 \mathbf{R}_2^1 \mathbf{x}_2 + \mathbf{p}_{21}^1 \end{cases} \quad (3.27)$$

ove \mathbf{R}_2^1 è la matrice di rotazione dalla camera destra alla camera sinistra, mentre \mathbf{p}_{21}^1 è l'origine della camera destra espressa nel sistema di riferimento della camera sinistra. Definendo $\mathbf{p}_{12}^1 = -\mathbf{p}_{21}^1$ e $\mathbf{p}_{12}^2 = \mathbf{R}_1^2 \mathbf{p}_{12}^1$, è possibile calcolare i coefficienti:

$$\begin{cases} \lambda_1 = -\frac{(\mathbf{x}_1^T \mathbf{R}_2^1 \mathbf{x}_2) (\mathbf{x}_2^T \mathbf{p}_{12}^2) + (\mathbf{x}_2^T \mathbf{x}_2) (\mathbf{x}_1^T \mathbf{p}_{21}^1)}{(\mathbf{x}_1^T \mathbf{x}_1) (\mathbf{x}_2^T \mathbf{x}_2) - (\mathbf{x}_1^T \mathbf{R}_2^1 \mathbf{x}_2)^2} \\ \lambda_2 = -\frac{(\mathbf{x}_1^T \mathbf{x}_1) (\mathbf{x}_2^T \mathbf{p}_{21}^1) + (\mathbf{x}_1^T \mathbf{R}_2^1 \mathbf{x}_2) (\mathbf{x}_1^T \mathbf{p}_{12}^2)}{(\mathbf{x}_1^T \mathbf{x}_1) (\mathbf{x}_2^T \mathbf{x}_2) - (\mathbf{x}_1^T \mathbf{R}_2^1 \mathbf{x}_2)^2} \end{cases} \quad (3.28)$$

Trovati i coefficienti e sostituendoli all'interno di (3.27), il punto 3D espresso nel sistema di riferimento della camera sinistra è dato dal punto medio:

$$\mathbf{X} = \frac{\mathbf{X}_1^1 + \mathbf{X}_2^1}{2} \quad (3.29)$$

Capitolo 4

Set-up sperimentale e sistemi di riferimento

Nel corso dei capitoli 2 e 3, sono stati affrontati gli aspetti teorici riguardanti il moto relativo tra satelliti, sia traslazionale sia rotazionale, e gli step da effettuare nel corso dell'analisi delle immagini acquisite con il sistema di stereo visione.

In questo capitolo verrà data una descrizione del set-up sperimentale usato in laboratorio [19] e dei sistemi di riferimento presi in considerazione. Per i sistemi di riferimento verranno trattate anche delle equazioni necessarie per eseguire tutte le rototraslazioni necessarie [20].

4.1 Set-up sperimentale

La valutazione del moto relativo tra il chaser ed il target è stata fatta in laboratorio presso il Dipartimento di Ingegneria Industriale (DII) dell'Università di Padova. In questo laboratorio, è presente la facility SPARTANS [19] atta a riprodurre le manovre di prossimità tra satelliti. Tale facility garantisce il moto traslazionale a due gradi di libertà su un tavolo a basso attrito grazie ad un modulo traslazionale o *Translation Module* (TM) e il moto rotazionale grazie ad un modulo rotazionale o *Attitude Module* (AM) ad un satellite chaser. Tuttavia, nel corso dell'attività speri-

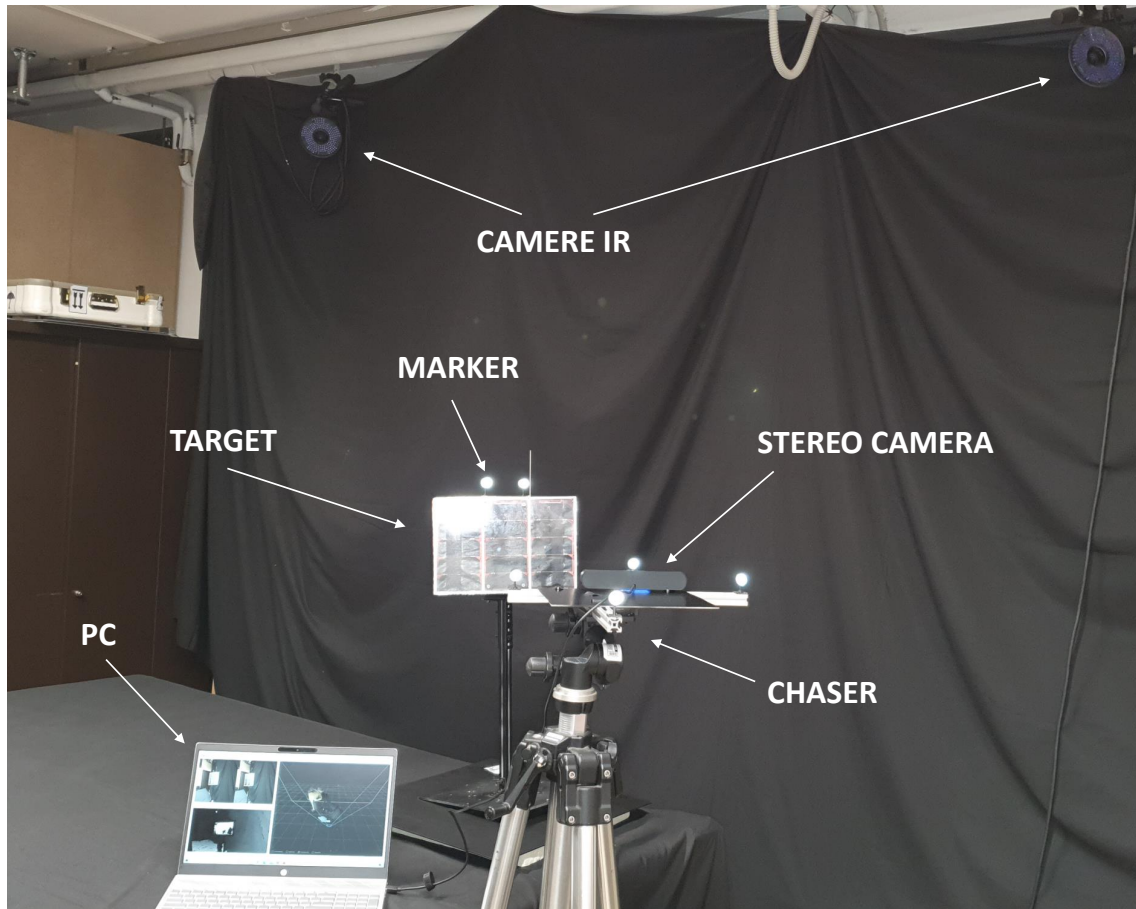


Figura 4.1: Setup sperimentale.

mentale non è stata utilizzata completamente la facility ma qualche sua componente, visibile nella Fig. 4.1. In particolare, il set-up sperimentale utilizzato è formato da:

- Il satellite target, realizzato mediante stampa 3D, con pannelli solari ed antenne;
- Una stereo camera ZED fissata su un telaio d'alluminio su un treppiedi, che rappresenta il chaser;
- Un sistema di Motion Capture (MC) con 6 camere IR, che permette di ottenere un riferimento fiduciario per il moto relativo tra target e chaser;
- 6 camere IR che servono a creare il riferimento fiduciario con il sistema di motion capture;
- Un set di marker, riflettenti nell'infrarosso, posizionati sul chaser e sul target che vengono ripresi in ogni istante dal sistema di motion capture;

- Un PC che acquisisce ed elabora le immagini della stereo camera;
- Un PC che acquisisce i dati del motion capture;
- Uno stadio rotativo motorizzato che impartisce un moto rotazionale al target.

4.1.1 Motion Capture



Figura 4.2: Bacchette per la calibrazione del motion capture.

Il sistema di Motion Capture è quello adibito ad ottenere un sistema di riferimento fiduciario per il moto relativo tra chaser e target. Tale riferimento è necessario per poter verificare che la stima eseguita tramite il sistema di stereo visione sia corretta. Esso è formato da 6 camere IR che riescono a stimare la posizione dei marker, riflettenti nell'infrarosso, in un sistema di riferimento mondo, come definito nel paragrafo 4.2.

Ad ogni nuovo utilizzo del motion capture, un sistema di riferimento mondo deve essere definito. Questo viene fatto attraverso il processo di calibrazione che è cruciale, in quanto senza un riferimento affidabile tutti i confronti per valutare la bontà dell'algoritmo di stima risultano vani. La calibrazione viene fatta tramite tre bacchette, ortogonali tra di loro, che definiscono il sistema di riferimento LVLH come visibile nella Fig.4.2. Le bacchette vengono visualizzate da tutte le 6 camere in un certo intervallo di tempo, con il vincolo che le bacchette siano tutte visibili e non ci siano sovrapposizioni. Ogni asse ha un diverso numero di marker riflettenti, quindi il software adibito alla calibrazione del sistema attribuisce l'asse in funzione del numero di marker. Un volta eseguito questo primo step, una semplice bacchetta

deve essere mossa all'interno del volume di lavoro, in maniera tale che il sistema di motion capture riesca ad attribuire una coordinata $(x_W, y_W, z_W)^T$ ad ogni marker all'interno di quel volume.

Ottenuto il sistema di riferimento mondo, è possibile attribuire ai marker posti sulla base superiore del chaser e del target le coordinate, che, come si vedrà nel dettaglio nel paragrafo 4.2, sono necessarie per calcolare il moto relativo tra i due. La frequenza di acquisizione è pari a 50 Hz, differente da quella delle stereo camera, e pertanto il vettore delle coordinate del Motion Capture deve essere uniformato a quello della stereo camera in termini di frequenza per poter valutare gli errori di stima. Per fare ciò, è stata eseguita un'interpolazione tramite una funzione spline delle coordinate acquisite dal Motion Capture.

4.1.2 Satellite target

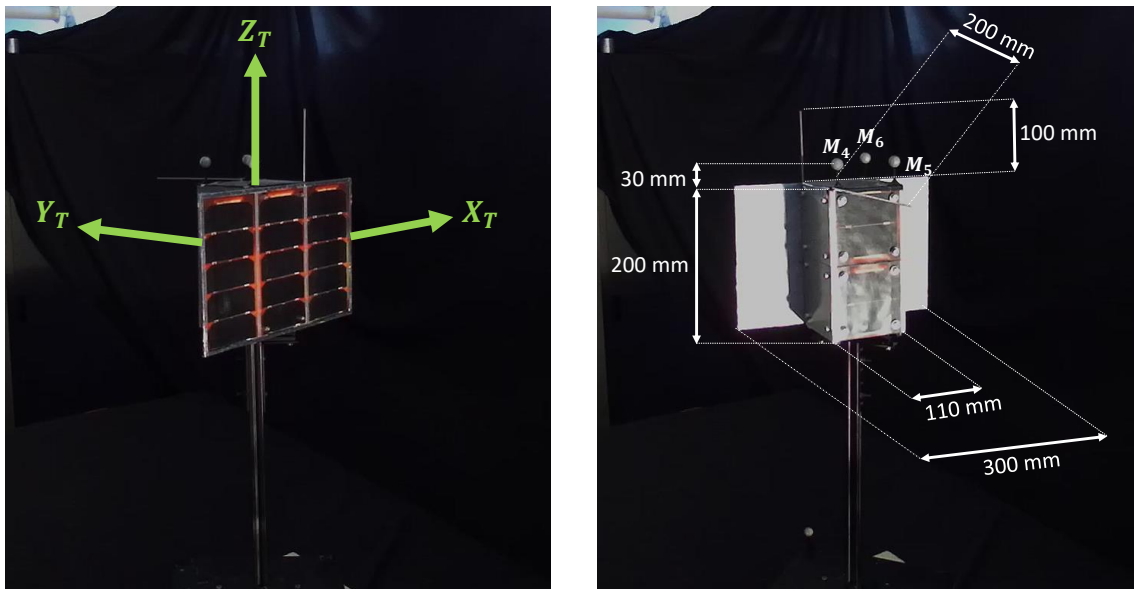


Figura 4.3: Satellite target.

Nella Fig. 4.3 è possibile osservare il satellite target utilizzato da una prospettiva "anteriore" ed una "posteriore". Esso è il mock-up di un cubesat 2U comprendente un pannello solare e due antenne ortogonali tra di loro poste sulla faccia superiore. Le celle del pannello solare sono molto utili per la ricerca di features, così come tutti gli spigoli del satellite in cui c'è variazione di intensità luminosa,

	Dimensione [mm]
Cubesat	110 × 200 × 110
Pannello solare	300 × 200
Antenna "verticale"	100
Antenna "orizzontale"	200
Altezza marker	30

Tabella 4.1: Dimensione del target.

come spiegato nel paragrafo 3.2. Inoltre, per poter localizzare il target all'interno del sistema di riferimento mondo tramite il sistema di Motion Capture, tre marker sferici sono posizionati sulla base superiore del target. Il target è fissato su uno stadio rotativo motorizzato per ruotare il target di un determinato angolo e con una determinata velocità angolare. Le caratteristiche geometriche del target sono riassunte nella Tab. 4.1.

4.1.3 Satellite chaser e stereo camera

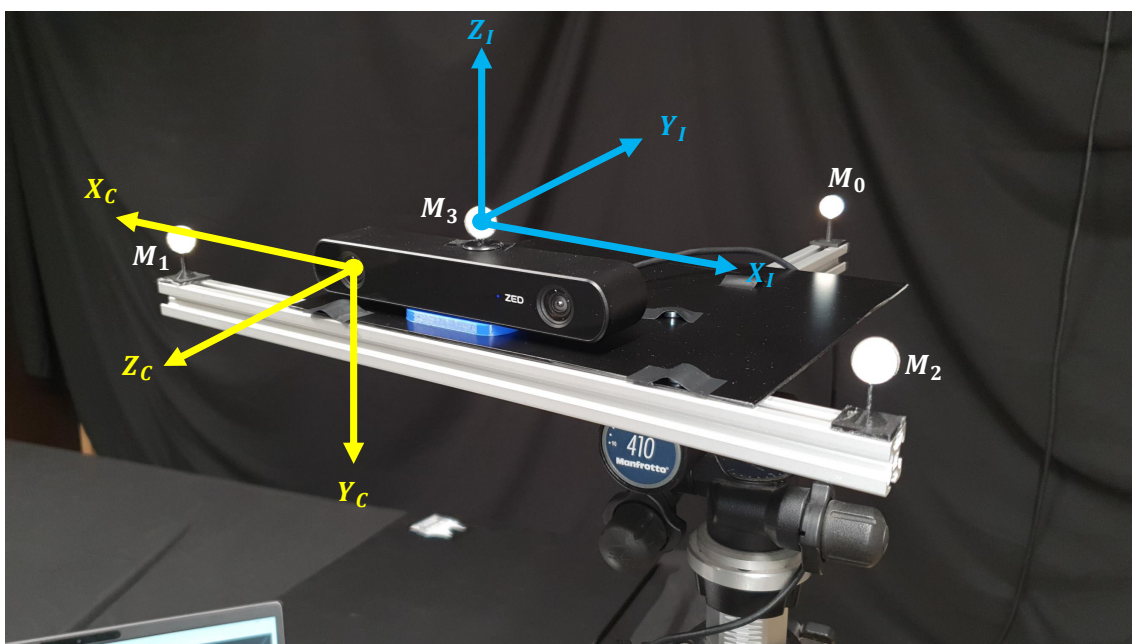


Figura 4.4: Satellite chaser.

Come visibile nella Fig. 4.4, diversamente dal target, il chaser non è rappresentato da un cubesat, ma semplicemente dalla stereo camera supportata da un pannello di alluminio e poggiata su un pannello colorato di nero. Siccome il chaser è

Risoluzione [pixel]	2208 × 1242
Field of view [deg]	110(H) × 70(V) × 120(D)
Dimensioni [mm]	175 × 30 × 33
Peso [g]	166
Alimentazione elettrica	380mA / 5V
Dimensione pixel [μm]	2 × 2
Baseline [mm]	120
Lunghezza focale [mm]	2.12

Tabella 4.2: Principali caratteristiche della stereo camera.

tenuto fermo, come si vedrà nel capitolo 5, la sue proprietà geometriche e di massa non sono importanti. Sui vertici del pannello di alluminio e sopra la stereo camera sono poggiati dei marker che vengono visualizzati dal sistema di Motion Capture per localizzare il chaser nel sistema di riferimento mondo.

La stereo camera è una ZED2 prodotta da Stereolabs [21], che acquisisce immagini con una frequenza di 15 Hz. A circa il centro della stereo camera è posizionato un marker che serve a fissare il centro del sistema di riferimento chaser come si vedrà nel paragrafo 4.2.1. Le proprietà della stereo camera sono visibili nella Tab. 4.2.

La stereo camera viene collegata tramite una porta USB ad un PC che permette l'analisi delle immagini, che è stata fatta in post-processing. La registrazione di un set di frame di immagini a 15 Hz e con una risoluzione di 2K è resa possibile dal software ZED-SDK. I video vengono salvati in formato .svo e successivamente convertiti in diversi frame in formato .png. Le immagini sono state poi importate dal software MATLAB e, tramite l'*Image processing toolbox*, si è proceduto con l'analisi delle immagini per la ricerca delle features. Inoltre, per velocizzare la stima, le coordinate delle features sono state salvate in un file per poi essere utilizzate dagli algoritmi successivi.

4.1.4 Stadio rotativo motorizzato

La base inferiore del target è fissata tramite quattro viti su di uno stadio rotativo motorizzato, visibile nella Fig. 4.5, per imprimere al target un moto rotatorio. L'attuatore della slitta è il driver 8SMC1-USBhF [22], che a sua volta viene

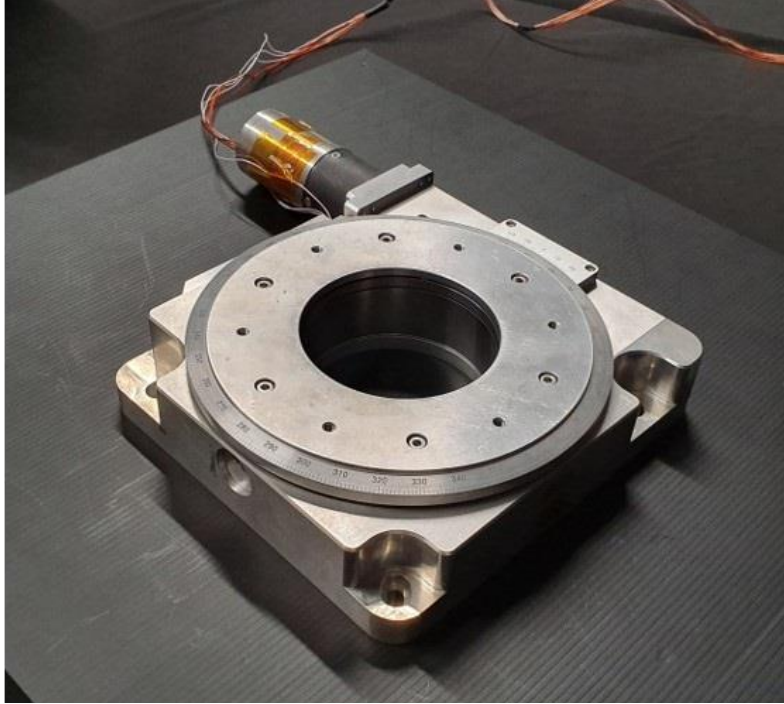


Figura 4.5: Stadio rotativo motorizzato.

comandato da un PC esterno. Il driver rende possibile la rotazione dello stadio non in termini di angoli bensì in termini di step, dove uno step è pari a 0.0025 deg. La velocità rotazionale massima permessa è di 5000 step/s, che corrisponde a 12.5 deg/s.

4.2 Sistemi di riferimento

Nel paragrafo 4.1 è stata illustrata la metodologia per ottenere un riferimento fiduciario tramite i marker sferici posizionati sul chaser e sul target. Questi marker sferici, le cui coordinate $(x_{W_i}, y_{W_i}, z_{W_i})$ sono note per ogni step di acquisizione del motion capture, servono per generare il sistema di riferimento del chaser e del target, da cui è ottenuto il moto relativo tra i due mock-up che va confrontato con quello in output dalla stima con il sistema di visione. I sistemi di riferimento complessivi introdotti sono 4:

- W : sistema di riferimento mondo generato dalla calibrazione del motion capture come spiegato nel paragrafo 4.1. Siccome la calibrazione ha l'asse Y lungo

la verticale locale, una rotazione del sistema di riferimento risulta necessaria per riportarsi nel sistema di riferimento LVLH classico con asse Z verticale;

- C : sistema di riferimento stereo camera, descritto nel paragrafo 2.3.3;
- I : sistema di riferimento inspector (chaser) che viene generato dai marker sferici disposti nella faccia superiore del supporto della stereo camera, come visto nel paragrafo 4.1;
- T : sistema di riferimento target, anch'esso generato similmente a I .

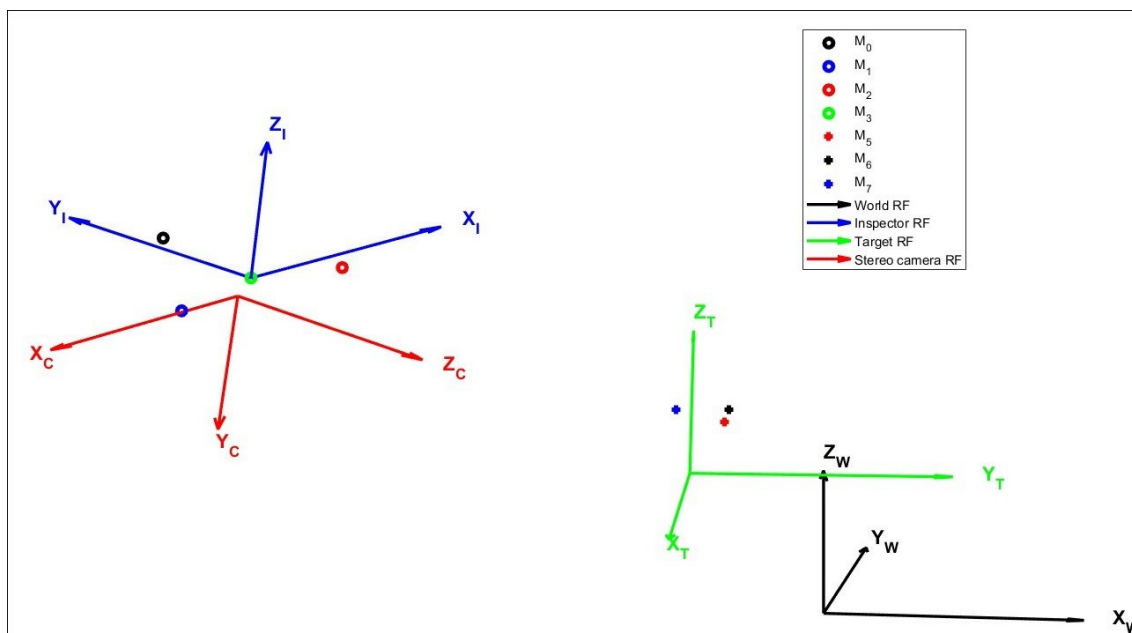


Figura 4.6: Sistemi di riferimento e posizione dei marker sferici.

4.2.1 Rototraslazione I-W e T-W

Come primo aspetto bisogna vedere come vengono generati i sistemi di riferimento del chaser e del target, espressi nelle coordinate mondo del Motion Capture. In particolare, in questa sezione verrà calcolata sia la rototraslazione tra sistema di riferimento mondo e sistema di riferimento chaser T_W^I sia la rototraslazione tra sistema di riferimento mondo e sistema di riferimento target T_W^T . Come detto, i marker sferici sono posizionati nelle facce superiori dei satelliti, che come visto nel paragrafo 4.1 sono 4 per il chaser (M_0, M_1, M_2, M_3) e 3 per il target (M_4, M_5, M_6). Per

generare un sistema di riferimento è necessaria la conoscenza di 3 punti (marker): un eventuale quarto punto può essere utilizzato per fissare il centro.

I marker che generano il sistema di riferimento I sono M_0 , M_1 e M_2 . Dalle coordinate dei marker \mathbf{M}_0 , \mathbf{M}_1 e \mathbf{M}_2 è possibile costruire i seguenti versori:

$$\begin{aligned}\mathbf{p}_I &= \frac{\mathbf{M}_2 - \mathbf{M}_1}{\|\mathbf{M}_2 - \mathbf{M}_1\|} \\ \mathbf{q}_I &= \frac{\mathbf{M}_0 - \mathbf{M}_1}{\|\mathbf{M}_0 - \mathbf{M}_1\|}\end{aligned}\tag{4.1}$$

Da (4.1) si è posto inizialmente il marker M_1 come l'origine della terna. Conoscendo i versori \mathbf{p}_I e \mathbf{q}_I è possibile ricostruire la terna:

$$\begin{aligned}\mathbf{x}_I &= \mathbf{p}_I \\ \mathbf{z}_I &= \frac{\mathbf{p}_I \times \mathbf{q}_I}{\|\mathbf{p}_I \times \mathbf{q}_I\|} \\ \mathbf{y}_I &= \mathbf{z}_I \times \mathbf{x}_I\end{aligned}\tag{4.2}$$

I versori (4.2) corrispondono a quelli della matrice di rotazione tra il sistema di riferimento I e il sistema di riferimento W . Il centro viene traslato in corrispondenza del marker M_3 di coordinate \mathbf{M}_3 , per centrarlo in quel punto. Quindi la matrice di rototraslazione finale risulta essere:

$$\begin{aligned}\mathbf{R}_I^W &= \begin{bmatrix} \mathbf{x}_I & \mathbf{y}_I & \mathbf{z}_I \end{bmatrix} \\ \mathbf{t}_I^W &= \mathbf{M}_3 - \mathbf{M}_1 \\ \mathbf{T}_I^W &= \begin{bmatrix} \mathbf{R}_I^W & \mathbf{t}_I^W \\ \mathbf{0}^T & 1 \end{bmatrix}\end{aligned}\tag{4.3}$$

Il procedimento per il calcolo della matrice di rototraslazione tra T e W è simile a quello tra I e W . I marker che generano il sistema di riferimento I sono M_4 , M_5 e M_6 . Dalle coordinate dei marker \mathbf{M}_4 , \mathbf{M}_5 e \mathbf{M}_6 è possibile costruire i seguenti versori:

$$\begin{aligned}
\mathbf{p}_T &= \frac{\mathbf{M}_4 - \mathbf{M}_5}{\|\mathbf{M}_4 - \mathbf{M}_5\|} \\
\mathbf{q}_T &= -\frac{\mathbf{M}_6 - \mathbf{M}_5}{\|\mathbf{M}_6 - \mathbf{M}_5\|}
\end{aligned} \tag{4.4}$$

Da (4.4) si è posto inizialmente il marker M_5 come l'origine della terna. Il segno negativo per il calcolo del versore \mathbf{q}_T è stato scelto in modo tale che il verso di \mathbf{z}_T sia lo stesso di \mathbf{z}_I , in base alla disposizione dei marker sul piano superiore del target. Conoscendo i versori \mathbf{p}_T e \mathbf{q}_T è possibile costruire la terna:

$$\begin{aligned}
\mathbf{x}_T &= \mathbf{p}_T \\
\mathbf{z}_T &= \frac{\mathbf{p}_T \times \mathbf{q}_T}{\|\mathbf{p}_T \times \mathbf{q}_T\|} \\
\mathbf{y}_T &= \mathbf{z}_T \times \mathbf{x}_T
\end{aligned} \tag{4.5}$$

I versori (4.5) corrispondono a quelli della matrice di rotazione tra il sistema di riferimento T e il sistema di riferimento W , analogamente ai versori (4.2). Ciò che cambia è la posizione del centro del sistema di riferimento, in quanto non è presente un quarto marker per eseguire una traslazione, come fatto in (4.3). Esso è stato posto nel centro geometrico del target, che essendo un corpo simmetrico è facile da calcolare. Nel piano (x_T, y_T) , la coordinata coincide al punto medio tra i marker M_4 e M_6 . Nota la dimensione del target (L_x, L_y, L_z) con misure geometriche come visibile nella Fig. 4.7, è possibile calcolare eseguire la traslazione finale. Pertanto, la matrice di rototraslazione finale risulta essere:

$$\begin{aligned}
\mathbf{R}_T^W &= \begin{bmatrix} \mathbf{x}_T & \mathbf{y}_T & \mathbf{z}_T \end{bmatrix} \\
\mathbf{t}_T^W &= \frac{1}{2} (\mathbf{M}_4 + \mathbf{M}_6) + (0, 0, -L_z)^T \\
\mathbf{T}_T^W &= \begin{bmatrix} \mathbf{R}_T^W & \mathbf{t}_T^W \\ \mathbf{0}^T & 1 \end{bmatrix}
\end{aligned} \tag{4.6}$$

Calcolate le matrici di rototraslazione \mathbf{T}_I^W e \mathbf{T}_T^W è possibile calcolare la matrice di rototraslazione relativa tra I e T , che corrisponde a posa ed assetto relativo, che deve essere confrontata con l'output del sistema di visione:

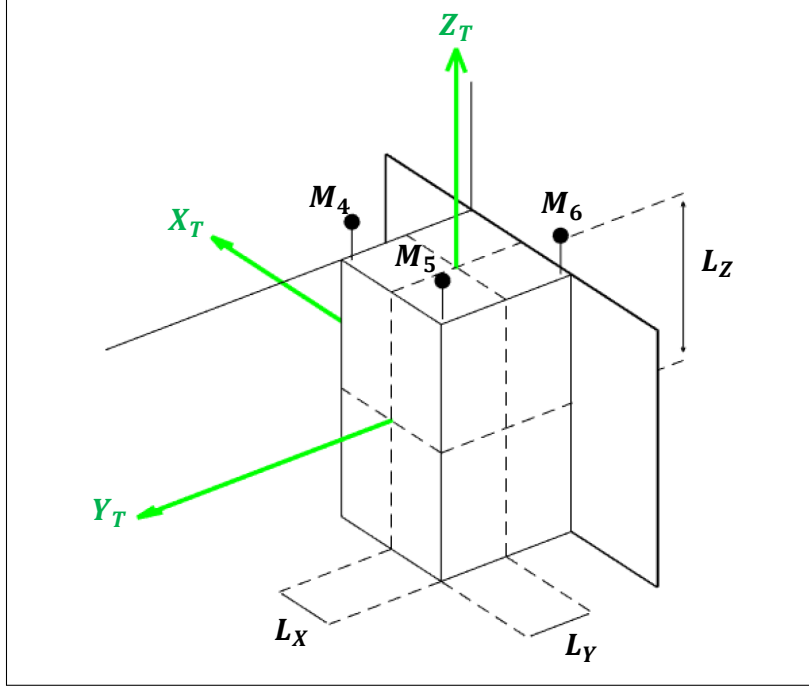


Figura 4.7: Dimensioni L_x , L_y e L_z del target.

$$\mathbf{T}_I^T = \mathbf{T}_I^W [\mathbf{T}_T^W]^{-1} \quad (4.7)$$

4.2.2 Rototraslazione C-W e I-C

Per poter trovare la rototraslazione \mathbf{T}_C^I che intercorre tra C e I deve essere dapprima trovata la rototraslazione \mathbf{T}_C^W tra C e W . La matrice \mathbf{T}_C^I deve essere calcolata per trasformare la stima del moto in coordinate camera in coordinate chaser, in quanto gli assi tra i due sistemi di riferimento hanno direzioni diverse.

Per fare questo, viene utilizzata una procedura simile a quella spiegata in [20]. Un set di 10 marker sferici sono disposti casualmente sul tavolo e su un pannello nero, venendo ripresi sia dalla stereo camera sia dal sistema di motion capture. I marker M_0 , M_1 , M_2 e M_3 , disposti sul chaser, definiscono come descritto nel paragrafo 4.2.1, la rototraslazione \mathbf{T}_I^W e vengono anch'essi ripresi insieme ai 10 marker sferici sul tavolo, come visibile nelle Fig. 4.9 e Fig. 4.10. Questi ultimi vengono ripresi in più istanti temporali, con pose diverse, da stereo camera e motion capture in maniera tale che le loro coordinate sono note sia nel sistema di riferimento C sia nel sistema di riferimento W . Per trovare quindi \mathbf{T}_C^W ci si serve dell'algoritmo

PnP (Perspective-n-Points) [23] [24].

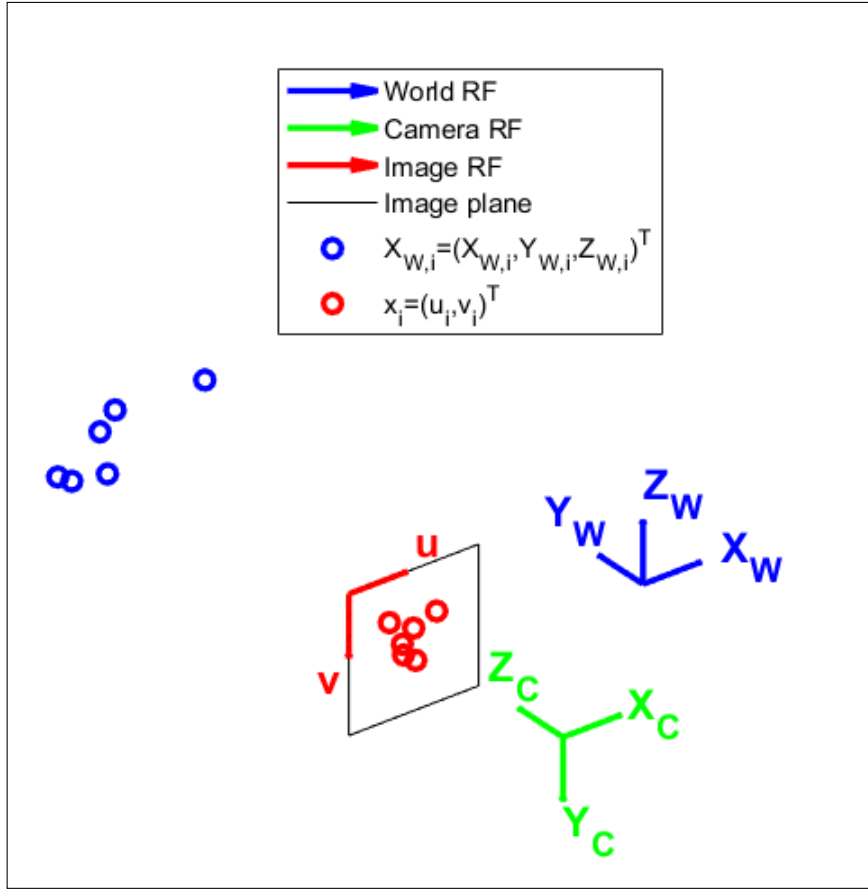


Figura 4.8: Punti visti nel sistema di riferimento C e W nell'algoritmo PnP.

Sia $\mathbf{X} = (X, Y, Z)^T$ la posizione dell' i esimo marker nel sistema di riferimento W , la cui rispettiva proiezione sul piano immagine nel sistema di riferimento C è $\mathbf{x} = (u, v)^T$, come rappresentato nella Fig. 4.8. Le coordinate dei centri del marker sul piano immagine vengono calcolati rilevando dei cerchi sul piano immagine tramite la Trasformata Circolare di Hough, come spiegato in [25]. Note le coordinate, riprendendo l'equazione di pinhole (2.37) e ponendo $\alpha = \frac{1}{\lambda}$ è possibile scrivere:

$$\begin{pmatrix} \mathbf{x} \\ 1 \end{pmatrix} = \alpha \mathbf{KAT} \begin{pmatrix} \mathbf{X} \\ 1 \end{pmatrix} \quad (4.8)$$

Il prodotto tra la matrici \mathbf{KAT} definisce la matrice di proiezione prospettica \mathbf{P} , e pertanto (4.8) può essere riscritta come:

$$\begin{pmatrix} \mathbf{x} \\ 1 \end{pmatrix} = \alpha \mathbf{P} \begin{pmatrix} \mathbf{X} \\ 1 \end{pmatrix} = \alpha \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix} \begin{pmatrix} \mathbf{X} \\ 1 \end{pmatrix} \quad (4.9)$$

Le incognite di questo problema lineare sono i 12 coefficienti che definiscono la matrice di proiezione prospettica \mathbf{P} del modello lineare pinhole camera. Per poter scrivere (4.9) come un sistema lineare nelle incognite $\mathbf{p} = (p_{11}, \dots, p_{34})^T$ è necessaria una manipolazione. Sia il vettore ausiliario $\mathbf{M} = \mathbf{P}(X, Y, Z, 1)^T$; eseguendo a destra e sinistra di (4.9) il prodotto vettoriale per \mathbf{M} risulta:

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} \times \mathbf{M} = \alpha \mathbf{M} \times \mathbf{M} = \mathbf{0} \quad (4.10)$$

Sostituendo all'interno di (4.10) l'espressione di \mathbf{M} :

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} \times \mathbf{P} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} = \begin{bmatrix} 0 & -1 & v \\ 1 & 0 & -u \\ -v & u & 0 \end{bmatrix} \mathbf{P} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} = \mathbf{0} \quad (4.11)$$

Espandendo la matrice \mathbf{P} nelle sue 12 componenti e saltando qualche passaggio intermedio, (4.11) può essere riscritta come:

$$\begin{pmatrix} -(p_{21}X + p_{22}Y + p_{23}Z + p_{24}) + v(p_{31}X + p_{32}Y + p_{33}Z + p_{34}) \\ (p_{11}X + p_{12}Y + p_{13}Z + p_{14}) - u(p_{31}X + p_{32}Y + p_{33}Z + p_{34}) \\ -v(p_{11}X + p_{12}Y + p_{13}Z + p_{14}) + u(p_{21}X + p_{22}Y + p_{23}Z + p_{24}) \end{pmatrix} = \mathbf{0} \quad (4.12)$$

Ora è possibile scrivere (4.12) come un sistema lineare nelle incognite del vettore \mathbf{p} delle componenti della matrice prospettica \mathbf{P} . Considerando $n = 1, 2, \dots, m$ il numero di punti, è possibile scrivere:

$$\begin{bmatrix}
X_1(1-v_1) & \dots & X_m(1-v_m) \\
Y_1(1-v_1) & \dots & Y_m(1-v_m) \\
Z_1(1-v_1) & \dots & Z_m(1-v_m) \\
1-v_1 & \dots & 1-v_m \\
X_1(u_1-1) & \dots & X_m(u_m-1) \\
Y_1(u_1-1) & \dots & Y_m(u_m-1) \\
Z_1(u_1-1) & \dots & Z_m(u_m-1) \\
u_1-1 & \dots & u_m-1 \\
X_1(v_1-u_1) & \dots & X_m(v_m-u_m) \\
Y_1(v_1-u_1) & \dots & Y_m(v_m-u_m) \\
Z_1(v_1-u_1) & \dots & Z_m(v_m-u_m) \\
v_1-u_1 & \dots & v_m-u_m
\end{bmatrix}^T \begin{pmatrix} p_{11} \\ p_{12} \\ p_{13} \\ p_{14} \\ p_{21} \\ p_{22} \\ p_{23} \\ p_{24} \\ p_{31} \\ p_{32} \\ p_{33} \\ p_{34} \end{pmatrix} = \mathbf{B}\mathbf{p} = \mathbf{0} \quad (4.13)$$

ove \mathbf{p} è un vettore 12×1 mentre \mathbf{B} è una matrice $m \times 12$. Siccome ogni punto identificato sul piano immagine restituisce la coordinata u e la coordinata v , si evince facilmente come il numero minimo per risolvere il sistema lineare (4.13) in forma chiusa sia $m = 6$. Per minimizzare gli errori ed avere una stima più precisa, è preferibile avere $m > 6$, la cui risoluzione avviene ai minimi quadrati oppure tramite la scomposizione al valore singolare della matrice \mathbf{B} . Infatti, la matrice \mathbf{B} può essere scomposta come:

$$\mathbf{B} = \mathbf{U}\mathbf{D}\mathbf{V}^T \quad (4.14)$$

con il vettore \mathbf{p} che è rappresentato dall'ultima colonna della matrice \mathbf{V} , da cui è possibile ricavare la matrice di proiezione prospettica \mathbf{P} . Riscrivendo (4.8) senza le coordinate omogenee:

$$\mathbf{K} \begin{bmatrix} \mathbf{R} & \mathbf{t} \end{bmatrix} = \lambda \begin{bmatrix} \mathbf{p}_1 & \mathbf{p}_2 & \mathbf{p}_3 & \mathbf{p}_4 \end{bmatrix} \quad (4.15)$$

Invertendo le prime tre colonne di (4.15):

$$\lambda \mathbf{R} = \mathbf{K}^{-1} \begin{bmatrix} \mathbf{p}_1 & \mathbf{p}_2 & \mathbf{p}_3 \end{bmatrix} \quad (4.16)$$

Poiché la matrice dei parametri intrinseci \mathbf{K} e la matrice di proiezione prospettica \mathbf{P} sono note, anche (4.16) può essere risolta attraverso la scomposizione singolare:

$$\mathbf{K}^{-1} \begin{bmatrix} p_1 & p_2 & p_3 \end{bmatrix} = \mathbf{U} \mathbf{D} \mathbf{V}^T \quad (4.17)$$

A partire dall'eq. (4.17) si può finalmente ricavare la matrice di rototraslazione tra coordinate camera e coordinate mondo:

$$\begin{aligned} \mathbf{R}_C^W &= \mathbf{U} \mathbf{V}^T \\ t_C^W &= \frac{\mathbf{K}^{-1} p_4}{d_{11}} \\ \mathbf{T}_C^W &= \begin{bmatrix} \mathbf{R}_C^W & t_C^W \\ \mathbf{0}^T & 1 \end{bmatrix} \end{aligned} \quad (4.18)$$

Ora si hanno tutti gli ingredienti per poter calcolare la matrice di rototraslazione tra sistema di riferimento camera e sistema di riferimento chaser:

$$\mathbf{T}_C^I = \mathbf{T}_C^W [\mathbf{T}_I^W]^{-1} \quad (4.19)$$

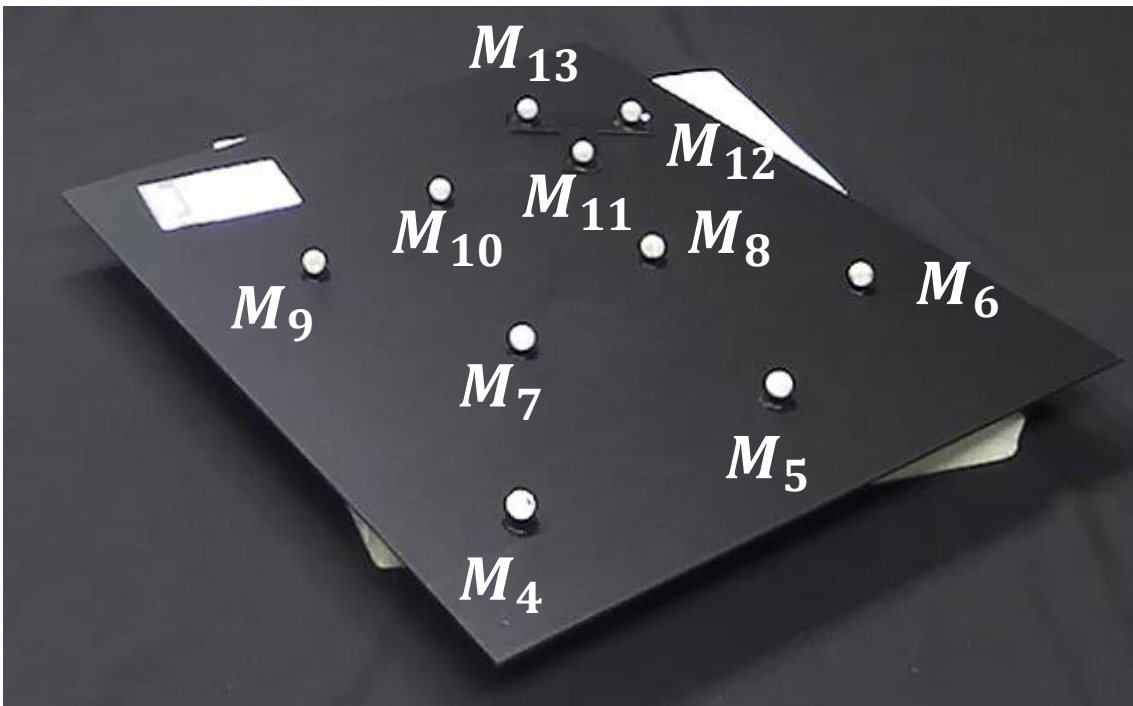


Figura 4.9: Disposizione dei marker nella posa 1.

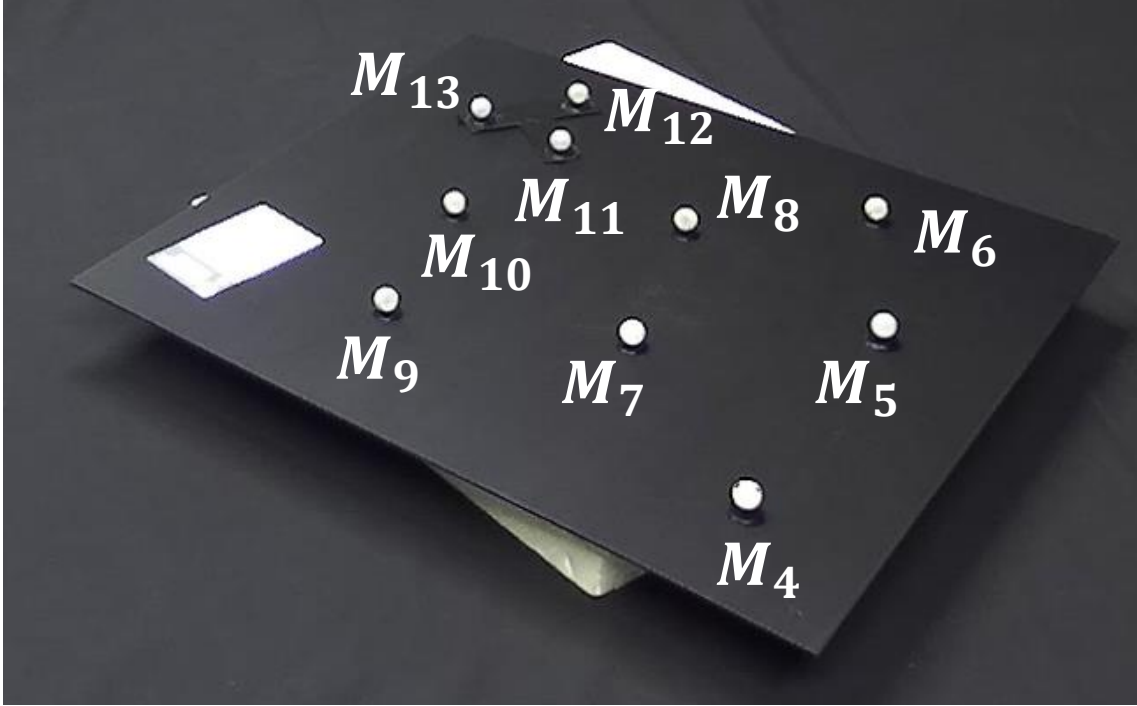


Figura 4.10: Disposizione dei marker nella posa 2.

Per il calcolo della matrice di rototraslazione tra C e I del set-up descritto nel paragrafo 4.1, sono stati presi in considerazione $n = 10$ marker (M_4, \dots, M_{13}) in due pose differenti, come visibile nelle Fig. 4.9 e Fig. 4.10. I valori degli errori di riproiezione, valutati applicando (2.37), sono visibili nella Tab. 4.3 e rappresentati nella Fig. 4.11.

	POSA 1		POSA 2	
	LEFT	RIGHT	LEFT	RIGHT
	$\epsilon_{rms}[px]$	$\epsilon_{rms}[px]$	$\epsilon_{rms}[px]$	$\epsilon_{rms}[px]$
M_4	0.0000	1.9376	1.7965	1.2928
M_5	1.2457	0.7906	1.7600	1.4637
M_6	1.9107	2.1553	2.6561	2.6668
M_7	1.3318	0.4176	0.8084	0.0000
M_8	0.1170	0.2356	0.3062	0.1186
M_9	0.0000	0.8248	0.8990	0.0000
M_{10}	0.3826	0.5049	0.5668	0.1064
M_{11}	0.0000	0.0000	0.1620	0.1426
M_{12}	0.4648	0.4798	0.4373	0.3027
M_{13}	0.3421	0.3039	0.1321	0.1497

Tabella 4.3: Errori di riproiezione dei marker con l'algoritmo PnP per entrambe le camere.

Inoltre, applicando l'algoritmo PnP ad entrambe le camere, possono essere

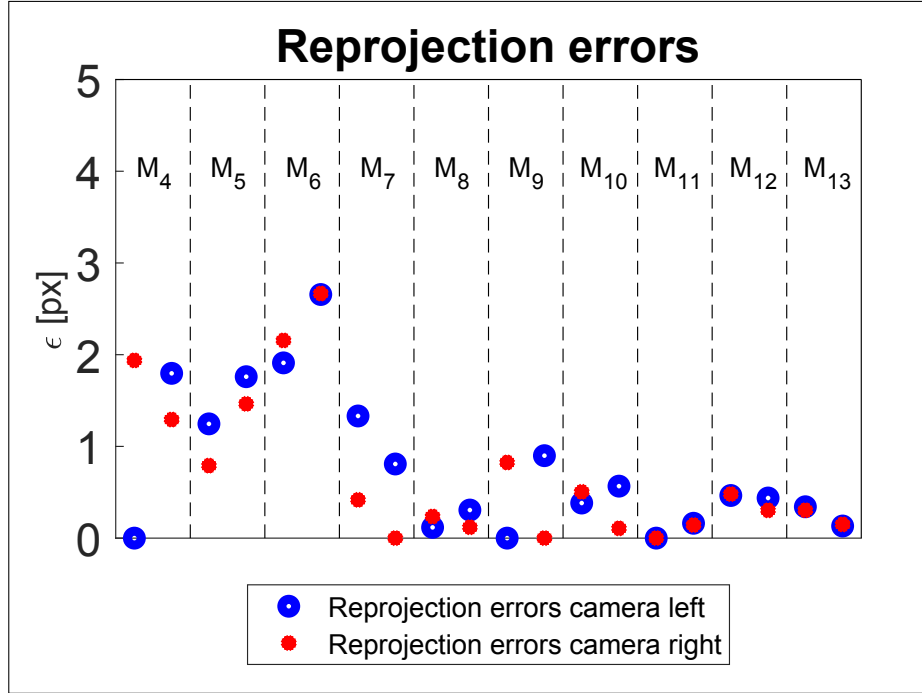


Figura 4.11: Errori di riproiezione dei marker con l’algoritmo PnP per entrambe le camere per le due pose.

calcolate le matrici di rototraslazione $\mathbf{T}_{C,L}^W$ e $\mathbf{T}_{C,R}^W$. Note queste, si può calcolare la rototraslazione tra camera sinistra e destra come $\mathbf{T}_{C,L}^{C,R} = \mathbf{T}_{C,L}^W [\mathbf{T}_{C,R}^W]^{-1}$ che può essere confrontata con quella in output dalla calibrazione con il metodo di Zhang, come visto nel paragrafo 3.1, visibile nella Tab. 4.4.

	Baseline [mm]			Angoli di Eulero [deg]		
	b_x	b_y	b_z	ϕ	θ	ψ
Zhang	-119.3633	-1.3261	-1.2679	-0.0590	0.1582	0,0080
Calibrazione	-119.5565	0.0184	0.0242	0.5265	0.3116	-0.5934
Errore rms	1.1698			0.4927		

Tabella 4.4: Confronto tra la rototraslazione tra le due camere ottenuta con l’algoritmo di Zhang e quella ottenuta con l’algoritmo PnP.

Capitolo 5

Stima del moto relativo: Extended Kalman Filter

Nel seguente capitolo verrà trattata la metodologia utilizzata per la stima del moto relativo tra chaser e target tramite il sistema di stereo visione. Esistono varie tecniche per la stima del moto relativo tramite un sistema di stereo visione. Come introdotto nel paragrafo 4.2.2, l'algoritmo PnP permette di trovare la rototraslazione tra il sistema di riferimento W ed il sistema di riferimento C , purché le coordinate di alcune features sul target siano note in entrambi i sistemi di riferimento. In questo scenario, che è definito cooperante, tutte le matrici di rototraslazione possono essere calcolate (come visto nel paragrafo 4.2) e pertanto la risoluzione sarebbe banale. Ma lo scenario considerato è di non cooperazione: se il chaser è per definizione operativo e controllato; quindi, le sue coordinate nel sistema di riferimento W sono note tramite sensori per la determinazione di assetto e posizione, ciò non si può dire per il target. Pertanto, l'algoritmo PnP non può essere utilizzato per la stima del moto relativo in questo scenario.

Una soluzione per stimare il moto relativo tramite un sistema di stereo visione è data dall'utilizzo di un filtro di Kalman, e più precisamente un filtro di Kalman esteso (EKF) [26]. A partire da una stima iniziale, che può essere quantificata tramite altri sensori o algoritmi, il filtro riesce a predire uno stato dinamico basato su misure affette da rumore, del quali il sistema di stereo visione è affetto.

Ci sono diversi tipologie di filtro di Kalman, i cui principali sono:

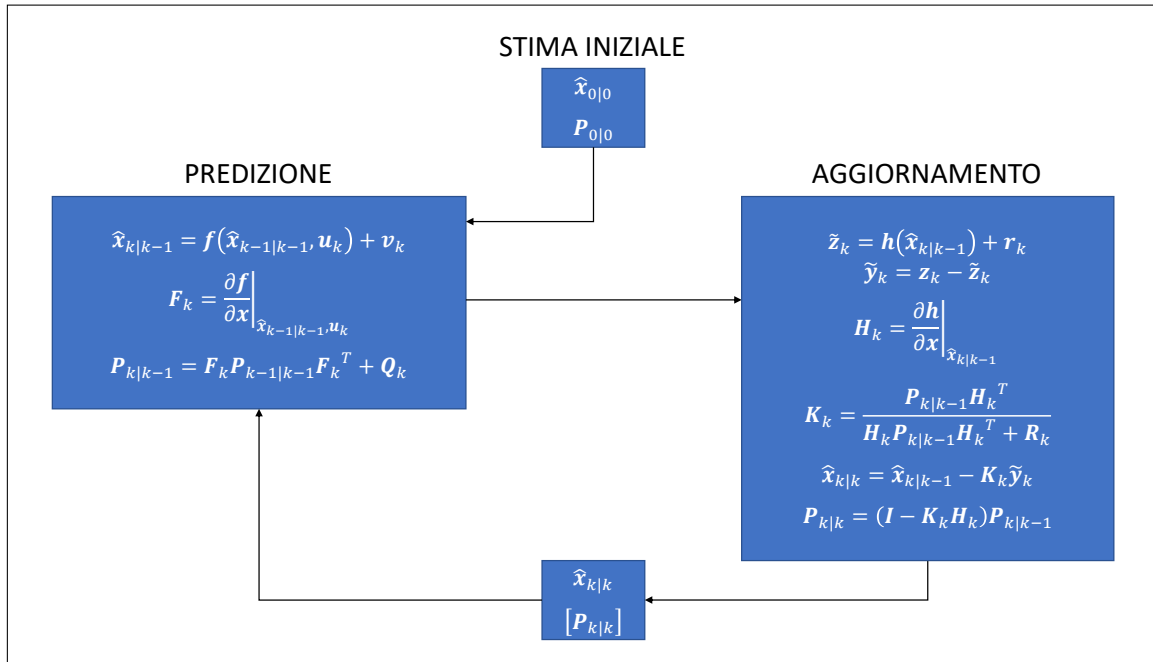


Figura 5.1: Schematizzazione dell'algoritmo EKF.

- *Filtro di Kalman standard (KF)*: filtro utilizzato in caso di sistemi dinamici lineari;
- *Filtro di Kalman esteso (EKF)*: filtro utilizzato in caso di sistemi dinamici non lineari.

Lo scenario di moto orbitale, come visto nel capitolo 2, non è lineare. Se per la dinamica traslazionale relativa è possibile eseguire una linearizzazione attraverso le equazione CW (paragrafo 2.1), ciò non è possibile per la dinamica rotazionale relativa, che è fortemente non lineare (paragrafi 2.2.1 e 2.2.2): deve essere utilizzato quindi un EKF. L'EKF è quindi un filtro utilizzato per linearizzare le equazioni del moto. Ciò ovviamente può portare ad errori, ma che possono essere ridotti se lo step per la linearizzazione è piccolo.

5.1 Struttura dell'EKF

Nella Fig. 5.1 si osservano gli step dell'EKF. Come struttura è molto simile ad un filtro di Kalman standard, con la sola differenza della necessità di linearizzare i

modelli dinamici e di misura, descritti nei paragrafi 5.2 e 5.3 rispettivamente. Nelle seguenti sezioni si spiegherà nel dettaglio la struttura dell'EKF e dei tre macro-blocchi che lo costituiscono.

5.1.1 Inizializzazione

L'inizializzazione dell'EKF è di fondamentale importanza per la convergenza del filtro alle condizioni di moto relativo reale. Infatti, prendendo condizioni iniziali casuali, è difficile che il filtro converga alla soluzione corretta e quindi bisogna porre particolare attenzione a come inizializzare le variabili di moto.

I due elementi da inizializzare sono, come visibile nella Fig. 5.1, $\hat{\mathbf{x}}_{0|0}$ e $\mathbf{P}_{0|0}$, che sono rispettivamente una stima iniziale dello stato del sistema e la matrice di covarianza dell'errore iniziale di stima, i cui elementi diagonali sono le varianze dello stato. Se per la matrice $\mathbf{P}_{0|0}$ possono essere presi valori "ragionevoli" in base all'incertezza della stima iniziale $\mathbf{x}_{0|0}$, per quanto riguarda $\hat{\mathbf{x}}_{0|0}$ devono essere fatte considerazioni differenti.

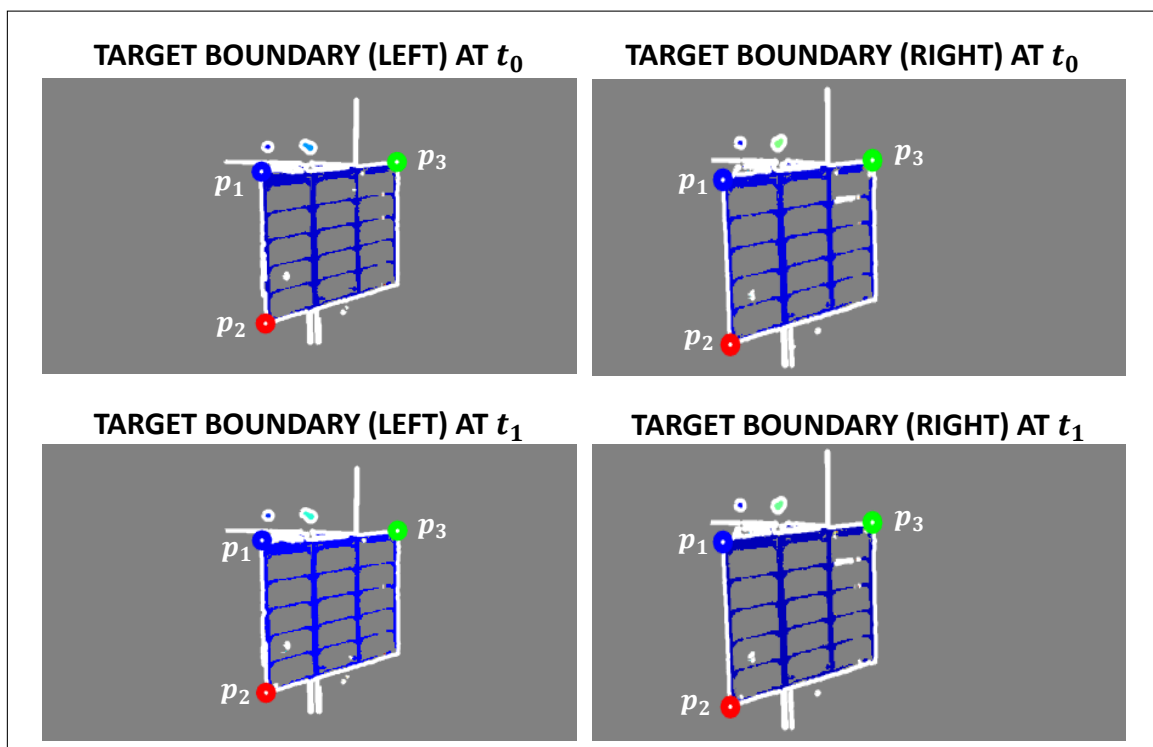


Figura 5.2: Vertici del target per l'inizializzazione della rotazione dell'EKF trovati usando la funzione *bwboundaries* di MATLAB [27].

L'approccio proposto è quello di eseguire una prima manovra di ispezione del satellite target, in cui viene svolta almeno un'intera rivoluzione per permettere al sistema di visione di identificare più features su varie superfici del satellite target. Inoltre, questa manovra è necessaria in quanto permette al chaser di conoscere certe proprietà di moto prima di propagare il filtro.

Per inizializzare la posizione del centro di massa relativo, quindi la traslazione relativa, il chaser compie una rivoluzione intera attorno al target. Nel corso della rotazione, vengono visualizzate le features del target tramite il sistema di visione, cercando quelle che coprono meglio la superficie del target, che vengono poi triangolate per calcolare la coordinata 3D nel sistema di riferimento C . Se questo procedimento viene fatto in maniera ottimale, una buona stima della traslazione relativa iniziale è rappresentata dal baricentro delle features.

Per inizializzare la velocità traslazionale relativa l'approccio è molto simile a quello per la traslazione relativa. In questo caso, la stessa feature deve essere triangolata in due frame successivi. Nota la frequenza di acquisizione è possibile calcolare la velocità tramite un rapporto incrementale.

L'inizializzazione della rotazione è un po' più complessa e delicata, ma ricalca in linea di principio quelle precedenti. Il chaser, durante l'ispezione iniziale, tramite il sistema di visione riesce ad identificare la forma del target. Nel caso in esame il target è un cubesat, quindi la geometria è piuttosto semplice. Come visto nel paragrafo 4.2, un sistema di riferimento viene generato da tre punti. Pertanto, il sistema di visione riconosce tre punti caratteristici del target, ovvero i tre vertici del pannello solare, come visibile nella Fig. 5.2. Al frame iniziale, i vertici $\mathbf{p}_1(t_0)$, $\mathbf{p}_2(t_0)$ e $\mathbf{p}_3(t_0)$, le cui coordinate sono note nei piani immagini, vengono triangolati come visto nel paragrafo 3.6. Note le coordinate 3D nel sistema di riferimento C , è possibile costruire il sistema di riferimento:

$$\begin{aligned}
\mathbf{p}(t_0) &= \frac{\mathbf{p}_2(t_0) - \mathbf{p}_1(t_0)}{\|\mathbf{p}_2(t_0) - \mathbf{p}_1(t_0)\|} \\
\mathbf{q}(t_0) &= \frac{\mathbf{p}_3(t_0) - \mathbf{p}_1(t_0)}{\|\mathbf{p}_3(t_0) - \mathbf{p}_1(t_0)\|} \\
\mathbf{x}(t_0) &= \mathbf{p}(t_0) \\
\mathbf{z}(t_0) &= \frac{\mathbf{p}(t_0) \times \mathbf{q}(t_0)}{\|\mathbf{p}(t_0) \times \mathbf{q}(t_0)\|} \\
\mathbf{y}(t_0) &= \mathbf{z}(t_0) \times \mathbf{x}(t_0)
\end{aligned} \tag{5.1}$$

Si osserva come il sistema di riferimento generato sia ortogonale rispetto a quello prodotto dai marker tramite il motion capture. Pertanto, risulta necessaria un'ultima rotazione per riportarsi nel sistema di riferimento fiduciario:

$$\mathbf{R}(t_0) = \begin{bmatrix} 0 & 0 & -1 \\ 1 & 0 & 0 \\ 0 & -1 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x}(t_0) & \mathbf{y}(t_0) & \mathbf{z}(t_0) \end{bmatrix} \tag{5.2}$$

Nota la matrice di rotazione iniziale, tramite le equazioni descritte nel paragrafo 2.2.2 è possibile ricondursi agli angoli di Eulero iniziali $\boldsymbol{\alpha}(t_0)$ e quindi al quaternione iniziale $\mathbf{q}(t_0)$.

L'inizializzazione della velocità angolare segue i punti precedenti. In due frame successivi, t_0 e t_1 , note le coordinate dei vertici del pannello solare \mathbf{p}_1 , \mathbf{p}_2 e \mathbf{p}_3 in entrambi gli istanti temporali, è possibile calcolare le matrici di rotazione $\mathbf{R}(t_0)$ e $\mathbf{R}(t_1)$ con la stessa metodologia spiegata in (5.1) e (5.2), da cui si estraggono gli angoli di Eulero $\boldsymbol{\alpha}(t_0)$ e $\boldsymbol{\alpha}(t_1)$. La stima della velocità angolare iniziale si trova tramite un semplice rapporto incrementale:

$$\boldsymbol{\omega}(t_0) = \frac{\boldsymbol{\alpha}(t_1) - \boldsymbol{\alpha}(t_0)}{t_1 - t_0} \tag{5.3}$$

5.1.2 Predizione

Come visto nel paragrafo precedente, una buona inizializzazione comporta una buona stima del filtro di Kalman. In particolare, viene inizializzato il vettore di stato $\hat{\mathbf{x}}_{0|0}$ e la matrice covarianza di errore $\mathbf{P}_{0|0}$, che sono detti rispettivamente vettore di *stato a posteriori* e matrice *di covarianza dell'errore di stima a posteriori*. Dallo stato a posteriori è possibile calcolare il vettore di *stato a priori* per l'istante di tempo successivo:

$$\hat{\mathbf{x}}_{k|k-1} = \mathbf{f}(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{u}_k) + \mathbf{v}_k \quad (5.4)$$

dove \mathbf{v}_k è il rumore di processo con matrice di covarianza \mathbf{Q}_k e \mathbf{u}_k è l'accelerazione di controllo. La natura della funzione \mathbf{f} verrà descritta nel paragrafo 5.2, che è non lineare. Pertanto, per andare avanti con l'algoritmo, è necessario calcolare la matrice *Jacobiana dello stato*:

$$\mathbf{F}_k = \left. \frac{\partial \mathbf{f}}{\partial \hat{\mathbf{x}}} \right|_{\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{u}_k} \quad (5.5)$$

calcolata utilizzando lo stato a posteriori. Nota la matrice Jacobiana e nota la matrice di covarianza dell'errore di stima a posteriori, è possibile calcolare la matrice *di covarianza dell'errore di stima a priori* come:

$$\mathbf{P}_{k|k-1} = \mathbf{F}_k \mathbf{P}_{k-1|k-1} \mathbf{F}_k^T + \mathbf{Q}_k \quad (5.6)$$

5.1.3 Aggiornamento

Una volta eseguita la predizione, è possibile eseguire l'aggiornamento sia in termini di stato sia in termini di misura. Le misure di cui necessita l'EKF saranno descritte nel paragrafo 5.3. Come primo aspetto, avviene l'aggiornamento del vettore *modello di misura*:

$$\tilde{\mathbf{z}}_k = \mathbf{h}(\hat{\mathbf{x}}_{k|k-1}) + \mathbf{r}_k \quad (5.7)$$

dove \mathbf{r}_k è il rumore di misura, con matrice di covarianza \mathbf{R}_k . La funzione \mathbf{h} è non lineare e verrà descritta nel paragrafo 5.3. Una stima del vettore delle misure $\tilde{\mathbf{z}}_k$ ottenuta usando 5.7 con $\hat{\mathbf{x}}_{k|k-1}$ viene confrontata poi con il vettore di misura tramite la differenza:

$$\tilde{\mathbf{y}}_k = \mathbf{z}_k - \tilde{\mathbf{z}}_k \quad (5.8)$$

Siccome la funzione \mathbf{h} non è lineare, anche in questo caso risulta necessaria l'introduzione della matrice *Jacobiana delle misure*:

$$\mathbf{H}_k = \left. \frac{\partial \mathbf{h}}{\partial \hat{\mathbf{x}}} \right|_{\hat{\mathbf{x}}_{k|k-1}} \quad (5.9)$$

calcolata utilizzando lo stato a priori. Per stimare lo stato a posteriori dello step successivo, nota la matrice Jacobiana delle misure, la matrice di covarianza dell'errore di stima a priori e la covarianza delle misure, si calcola il *guadagno di Kalman*:

$$\mathbf{K}_k = \frac{\mathbf{P}_{k|k-1} \mathbf{H}_k^T}{\mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^T + \mathbf{R}_k} \quad (5.10)$$

Il vettore di stato a posteriori e la matrice di covarianza dell'errore di stima a posteriori sono calcolati come segue:

$$\begin{aligned} \hat{\mathbf{x}}_{k|k} &= \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k \tilde{\mathbf{y}}_k \\ \mathbf{P}_{k|k} &= (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_{k|k-1} \end{aligned} \quad (5.11)$$

I valori di $\hat{\mathbf{x}}_{k|k}$ e $\mathbf{P}_{k|k}$ in output dall'aggiornamento allo step k sono in input per la predizione allo step $k + 1$ e il ciclo iterativo viene ripetuto n volte, dove n è il numero di step complessivi legato a come è stata eseguita la discretizzazione temporale. Nel corso del lavoro di tesi, n è stato scelto pari al numero di frame dell'acquisizione, eccetto per uno scenario.

Per la convergenza del filtro, è molto importante l'evoluzione della matrice $\mathbf{P}_{k|k}$. Essa è una matrice simmetrica, in cui gli elementi diagonali sono le varianze dell'errore di stima mentre gli elementi non diagonali rappresentano le covarianze. Analizzando il comportamento delle deviazioni standard associate ad ogni elemento

del vettore di stato, è possibile capire se il filtro converge o meno: se l'andamento temporale delle deviazioni standard tende a restringersi vuol dire che si sta andando a convergenza, e viceversa.

5.2 Modello dinamico

Nella struttura dell'algoritmo si è visto come entri in gioco il modello dinamico che deve essere attribuito allo scenario di studio, che è per l'appunto non lineare. Nel caso in questione, il vettore di stato che il filtro deve stimare è il seguente:

$$\mathbf{x} = (\boldsymbol{\rho}^T, \dot{\boldsymbol{\rho}}^T, \mathbf{q}^T, \boldsymbol{\omega}^T, \mathbf{P}_1^T, \dots, \mathbf{P}_N^T, k_1, k_2)^T \in \mathbb{R}^{15+3N} \quad (5.12)$$

dove $\boldsymbol{\rho}$ e $\dot{\boldsymbol{\rho}}$ sono i vettori che identificano la traslazione relativa, \mathbf{q} e $\boldsymbol{\omega}$ sono i vettori che identificano la rotazione relativa, \mathbf{P}_i sono le coordinate dell'iesimo punto identificato dal sistema di visione, k_1 e k_2 sono i rapporti delle componenti del tensore di inerzia. Il modello dinamico non lineare espresso in forma continua è:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{v} \quad (5.13)$$

dove \mathbf{v} è un rumore bianco gaussiano di media nulla con covarianza \mathbf{Q} . Nel caso applicativo, risulta più comodo utilizzare un modello discreto, come visto nel paragrafo 5.1, tramite un rapporto incrementale del tipo $\dot{\mathbf{x}} = \frac{\mathbf{x}_{k+1} - \mathbf{x}_k}{dt}$, dove dt corrisponde al tempo tra gli step di propagazione dello stato. Pertanto, il modello continuo (5.13) può essere riscritto come:

$$\mathbf{x}_{k+1} = \mathbf{x}_k + dt(\mathbf{f}(\mathbf{x}_k) + \mathbf{v}_k) \quad (5.14)$$

5.2.1 Modello dinamico della traslazione relativa

Il modello dinamico per la traslazione relativa ($\boldsymbol{\rho}$) e per la velocità traslazionale relativa ($\dot{\boldsymbol{\rho}}$) riprende le equazioni CW introdotte nel paragrafo 2.1. Riscrivendo

(2.11) e aggiungendo il vettore $\dot{\boldsymbol{\rho}}$, la dinamica traslazionale relativa continua è [28]:

$$\begin{pmatrix} \dot{\boldsymbol{\rho}} \\ \ddot{\boldsymbol{\rho}} \end{pmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 3n^2 & 0 & 0 & 0 & 2n & 0 \\ 0 & 0 & 0 & -2n & 0 & 0 \\ 0 & 0 & -n^2 & 0 & 0 & 0 \end{bmatrix} \begin{pmatrix} \boldsymbol{\rho} \\ \dot{\boldsymbol{\rho}} \end{pmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \boldsymbol{\eta} \quad (5.15)$$

Se ora viene applicata a (5.15) la discretizzazione fatta in (5.14) e trascurando le forze esterne agenti sul chaser in quanto nello scenario studiato esse non sono presenti, risulta che:

$$\begin{pmatrix} \boldsymbol{\rho}_{k+1} \\ \dot{\boldsymbol{\rho}}_{k+1} \end{pmatrix} = \begin{pmatrix} \boldsymbol{\rho}_k \\ \dot{\boldsymbol{\rho}}_k \end{pmatrix} + dt \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 3n^2 & 0 & 0 & 0 & 2n & 0 \\ 0 & 0 & 0 & -2n & 0 & 0 \\ 0 & 0 & -n^2 & 0 & 0 & 0 \end{bmatrix} \begin{pmatrix} \boldsymbol{\rho}_k \\ \dot{\boldsymbol{\rho}}_k \end{pmatrix} \quad (5.16)$$

Come detto in precedenza, questo modello è già di per sé lineare, in cui tutti i coefficienti della matrice di stato che premoltiplica $(\boldsymbol{\rho}_k, \dot{\boldsymbol{\rho}}_k)^T$ sono costanti e quindi riconducibile ad un filtro di Kalman standard. Come si vedrà, ciò non accade per le altre componenti del vettore di stato.

5.2.2 Modello dinamico della rotazione relativa

Il modello dinamico della la rotazione relativa utilizza il quaternione \mathbf{q} e non gli angoli di Eulero $\boldsymbol{\alpha}$ per i problemi legati ad un'eventuale singolarità citati nel paragrafo 2.2.2. Riscrivendo (2.27) e applicando la discretizzazione fatta in (5.14), il quaternione relativo è:

$$\mathbf{q}_{k+1} = \mathbf{q}_k + dt \frac{1}{2} \begin{bmatrix} 0 & -\omega_{x_k} & -\omega_{y_k} & -\omega_{z_k} \\ \omega_{x_k} & 0 & \omega_{z_k} & -\omega_{y_k} \\ \omega_{y_k} & -\omega_{z_k} & 0 & \omega_{x_k} \\ \omega_{z_k} & \omega_{y_k} & -\omega_{x_k} & 0 \end{bmatrix} \mathbf{q}_k \quad (5.17)$$

A differenza degli altri modelli, il modello dinamico della velocità rotazionale relativa non può essere riferito direttamente alle equazioni di Eulero (2.21) ricavate nel paragrafo 2.2.1. Siano i sistemi di riferimento W , C e T introdotti nel paragrafo 4.2. La velocità angolare relativa nel sistema di riferimento W della target rispetto al chaser è [29]:

$$\boldsymbol{\omega} = \boldsymbol{\omega}_{WC} - \boldsymbol{\omega}_{WT} \quad (5.18)$$

dove $\boldsymbol{\omega}_{WC}$ e $\boldsymbol{\omega}_{WT}$ sono le velocità angolari del chaser (in coordinate camera) e del target espresse in coordinate W . Derivando rispetto al tempo (5.18):

$$\dot{\boldsymbol{\omega}}^{(W)} = \dot{\boldsymbol{\omega}}_{WC}^{(W)} - \dot{\boldsymbol{\omega}}_{WT}^{(W)} \quad (5.19)$$

Inoltre, $\dot{\boldsymbol{\omega}}^{(W)}$ può essere calcolata anche utilizzando il teorema di Coriolis:

$$\dot{\boldsymbol{\omega}}^{(W)} = \dot{\boldsymbol{\omega}}^{(C)} + \boldsymbol{\omega}_{WC}^{(C)} \times \boldsymbol{\omega}^{(C)} \quad (5.20)$$

Siccome ciò che interessa è trovare $\dot{\boldsymbol{\omega}}^{(C)}$, è possibile combinare (5.19) e (5.20) isolando il termine $\dot{\boldsymbol{\omega}}^{(C)}$:

$$\dot{\boldsymbol{\omega}}^{(C)} = \dot{\boldsymbol{\omega}}_{WC}^{(W)} - \dot{\boldsymbol{\omega}}_{WT}^{(W)} - \boldsymbol{\omega}_{WC}^{(C)} \times \boldsymbol{\omega}^{(C)} \quad (5.21)$$

Dato che $\dot{\boldsymbol{\omega}}_{WC}^{(W)} = \dot{\boldsymbol{\omega}}_{WC}^{(C)}$ e $\dot{\boldsymbol{\omega}}_{WT}^{(W)} = \dot{\boldsymbol{\omega}}_{WT}^{(T)}$ e sapendo che la rotazione relativa tra i sistemi di riferimento C e T è pari a $\mathbf{R}_T^C(\mathbf{q})$ come visto in (2.28), (5.21) diventa:

$$\dot{\boldsymbol{\omega}}^{(C)} = \dot{\boldsymbol{\omega}}_{WC}^{(C)} - \mathbf{R}_T^C(\mathbf{q}) \dot{\boldsymbol{\omega}}_{WT}^{(T)} - \boldsymbol{\omega}_{WC}^{(C)} \times \boldsymbol{\omega}^{(C)} \quad (5.22)$$

Le equazioni di Eulero (2.21) per il sistema di riferimento C e T sono rispettivamente:

$$\begin{aligned}
\dot{\boldsymbol{\omega}}_{WC}^{(C)} &= \mathbf{I}_C^{-1} \left(\mathbf{N}_C - \boldsymbol{\omega}_{WC}^{(C)} \times \mathbf{I}_C \boldsymbol{\omega}_{WC}^{(C)} \right) \\
\dot{\boldsymbol{\omega}}_{WT}^{(T)} &= \mathbf{I}_T^{-1} \left(\mathbf{N}_T - \boldsymbol{\omega}_{WT}^{(T)} \times \mathbf{I}_T \boldsymbol{\omega}_{WT}^{(T)} \right)
\end{aligned} \tag{5.23}$$

Combinando (5.22) e (5.23), il vettore velocità angolare relativo $\boldsymbol{\omega}^{(C)}$ espresso in coordinate C è pari a:

$$\begin{aligned}
\dot{\boldsymbol{\omega}}^{(C)} &= \mathbf{I}_C^{-1} \left(\mathbf{N}_C - \boldsymbol{\omega}_{WC}^{(C)} \times \mathbf{I}_C \boldsymbol{\omega}_{WC}^{(C)} \right) \\
&\quad - \mathbf{R}_T^C(\mathbf{q}) \mathbf{I}_T^{-1} \left(\mathbf{N}_T - \boldsymbol{\omega}_{WT}^{(T)} \times \mathbf{I}_T \boldsymbol{\omega}_{WT}^{(T)} \right) \\
&\quad - \boldsymbol{\omega}_{WC}^{(C)} \times \boldsymbol{\omega}^{(C)}
\end{aligned} \tag{5.24}$$

Lo scenario in questione però è di non cooperazione, quindi \mathbf{I}_T , \mathbf{N}_T e $\boldsymbol{\omega}_{WT}^{(T)}$ non sono noti. Il tensore di inerzia \mathbf{I}_T è noto a inizio missione, ma nell'ipotesi di satellite dismesso le proprietà di massa sono cambiate. Il filtro deve quindi stimare i rapporti del tensore di inerzia \mathbf{I}_T come si vedrà nel paragrafo 5.2.4. Le coppie esterne sul target \mathbf{N}_T possono essere trattate come un rumore gaussiano a media nulla perché è un satellite passivo. La velocità angolare $\boldsymbol{\omega}_{WT}^{(T)}$ può essere espressa attraverso una rotazione, ovvero $\boldsymbol{\omega}_{WT}^{(T)} = [\mathbf{R}_T^C(\mathbf{q})]^T \boldsymbol{\omega}_{WT}^{(C)}$. Si può pertanto modificare (5.24), omettendo il pedice $^{(C)}$ per brevità di trattazione, in quanto la velocità angolare relativa è espressa in coordinate C :

$$\begin{aligned}
\dot{\boldsymbol{\omega}} &= \mathbf{I}_C^{-1} \left(\mathbf{N}_C - \boldsymbol{\omega}_{WC} \times \mathbf{I}_C \boldsymbol{\omega}_{WC} \right) \\
&\quad - \mathbf{R}_T^C(\mathbf{q}) \mathbf{I}_T^{-1} \left(\mathbf{N}_T - [\mathbf{R}_T^C(\mathbf{q})]^T (\boldsymbol{\omega}_{WC} - \boldsymbol{\omega}) \right. \\
&\quad \left. \times \mathbf{I}_T [\mathbf{R}_T^C(\mathbf{q})]^T (\boldsymbol{\omega}_{WC} - \boldsymbol{\omega}) \right) \\
&\quad - \boldsymbol{\omega}_{WC} \times \boldsymbol{\omega}
\end{aligned} \tag{5.25}$$

Riscrivendo (5.25) e applicando la discretizzazione fatta in (5.14), la velocità angolare relativa è:

$$\begin{aligned}
\boldsymbol{\omega}_{k+1} = & \boldsymbol{\omega}_k + dt \mathbf{I}_C^{-1} (\mathbf{N}_{C_k} - \boldsymbol{\omega}_{WC_k} \times \mathbf{I}_C \boldsymbol{\omega}_{WC_k}) \\
& - \mathbf{R}_T^C(\mathbf{q}_k) \mathbf{I}_T^{-1} \left(\mathbf{N}_{T_k} - [\mathbf{R}_T^C(\mathbf{q}_k)]^T (\boldsymbol{\omega}_{WC_k} - \boldsymbol{\omega}_k) \right. \\
& \times \mathbf{I}_T [\mathbf{R}_T^C(\mathbf{q}_k)]^T (\boldsymbol{\omega}_{WC_k} - \boldsymbol{\omega}_k) \left. \right) \\
& - \boldsymbol{\omega}_{WC_k} \times \boldsymbol{\omega}_k
\end{aligned} \tag{5.26}$$

Si è inoltre ipotizzato che sul chaser non agiscano forze esterne ($\mathbf{N}_C = \mathbf{0}$) perché il moto di osservazione avviene sfruttando delle orbite chiuse naturali attorno al target oppure dei punti stazionari delle equazioni CW e che esso non sia in rotazione ($\boldsymbol{\omega}_{WC} = \mathbf{0}$). Quindi, il modello finale della velocità angolare relativa è:

$$\boldsymbol{\omega}_{k+1} = \boldsymbol{\omega}_k - dt \mathbf{R}_T^C(\mathbf{q}_k) \mathbf{I}_T^{-1} \left(\mathbf{N}_{T_k} - [\mathbf{R}_T^C(\mathbf{q}_k)]^T \boldsymbol{\omega}_k \times \mathbf{I}_T [\mathbf{R}_T^C(\mathbf{q}_k)]^T \boldsymbol{\omega}_k \right) \tag{5.27}$$

I modelli dinamici rotazionali (5.17) e (5.27) sono fortemente non lineare, in quanto la velocità angolare $\boldsymbol{\omega}_{k+1}$ dipende direttamente dal quaternionione \mathbf{q}_k attraverso la matrice di rotazione $\mathbf{R}_T^C(\mathbf{q}_k)$. Pertanto, diversamente dal modello dinamico traslazionale (5.16) che è stato linearizzato dalle equazioni CW, non è possibile legare lo stato allo step k allo step $k+1$ attraverso una matrice di coefficienti slegati dallo stato. A ragione di ciò, il calcolo dello Jacobiano descritto nel paragrafo 5.1 risulta necessaria.

5.2.3 Modello dinamico delle features

Le features sono visualizzate dal sistema di stereo visione a poi triangolate per ottenere le coordinate 3D nel sistema di riferimento C . Tuttavia, come sarà descritto nel paragrafo 5.3, c'è la necessità di esprimerle rispetto al centro di massa del target nel sistema di riferimento T . Nel paragrafo 3.5 si è visto come, in un caso rotazionale, sia impossibile avere sempre lo stesso numero di features per tutta la durata dell'acquisizione. Questo rappresenta un problema nell'applicazione dell'EKF, in quanto le matrici Jacobiane descritte nel paragrafo 5.1 cambierebbero di dimen-

sione ad ogni step. Tuttavia, esiste un modo semplice per affrontare il problema ed esprimere le componenti dello stato $(\mathbf{P}_1, \dots, \mathbf{P}_N)$ in funzione delle altre componenti. Esso consiste nello sfruttare l'inizializzazione descritta nel paragrafo 5.1.1. Infatti, sia $\boldsymbol{\rho}$ la stima del centro di massa, sia $\boldsymbol{\rho}_i$ l' i -esima feature nota in coordinate C , la coordinata \mathbf{P}_i può essere calcolata come [29]:

$$\mathbf{P}_{i_k} = [\mathbf{R}_T^C(\mathbf{q}_k)]^T (\boldsymbol{\rho}_i - \boldsymbol{\rho}) \quad (5.28)$$

5.2.4 Modello dinamico dei rapporti del tensore di inerzia

Essendo il target noto con elevata incertezza, il valore del tensore di inerzia \mathbf{I}_T del target è anch'esso sconosciuto e pertanto deve essere stimato dall'EKF. D'altro canto, non è possibile stimare tutte le sei componenti del tensore simmetrico ma è possibile stimare i rapporti tra le componenti [30]. Ipotizzando che il sistema di riferimento T sia anche principale di inerzia, il tensore di inerzia è diagonale, e quindi le componenti da stimare sono tre. I rapporti del tensore sono [31]:

$$\begin{cases} l_x = \frac{I_{yy} - I_{zz}}{I_{xx}} \\ l_y = \frac{I_{zz} - I_{xx}}{I_{yy}} \\ l_z = \frac{I_{xx} - I_{yy}}{I_{zz}} \end{cases} \quad (5.29)$$

Siccome $\mathbf{l} = (l_x, l_y, l_z)^T$ varia con la predizione e l'aggiornamento del filtro di Kalman, occorre inserire i rapporti all'interno di (5.27) nella propagazione del filtro. Da notare come i rapporti di inerzia si mantengano costanti durante il piccolo periodo di acquisizione, e questo è importante nella scrittura delle equazioni dato che $\dot{\mathbf{l}} = \mathbf{0}$.

5.3 Modello di misura

Nella struttura dell'algoritmo si è visto che è necessaria la formulazione di un modello riguardante le variabili di misura, ovvero correlare lo stato al vettore

delle misure tramite una funzione matematica. Nel caso in questione, le misure che vengono effettuate sono le seguenti [32]:

$$\mathbf{z}_i = (u_{iL}, v_{iL}, u_{iR}, v_{iR}, \dot{u}_{iL}, \dot{v}_{iL}, \dot{u}_{iR}, \dot{v}_{iR}, d_i)^T \in \mathbb{R}^{9N} \quad (5.30)$$

dove $(u_{iL}, v_{iL})^T$ denota la proiezione dell' i -esima feature sul piano immagine della camera sinistra con velocità $(\dot{u}_{iL}, \dot{v}_{iL})^T$, $(u_{iR}, v_{iR})^T$ denota la proiezione dell' i -esima feature sul piano immagine della camera destra con velocità $(\dot{u}_{iR}, \dot{v}_{iR})^T$ e d_i è la disparity pari a $u_{iL} - u_{iR}$. La metodologia per trovare le coordinate sul piano immagine dei punti tramite l'analisi delle immagini è stata spiegata nel capitolo 3. Nota la frequenza di acquisizione della stereo camera, trovare le velocità delle feature è facile a condizione che l' i -esima features sia visibile allo step k e allo step $k + 1$, la cui selezione viene fatta come descritto nel paragrafo 3.5. In particolare:

$$\begin{aligned} \dot{\mathbf{w}}_{iL_k} &= \begin{pmatrix} \dot{u}_{iL_k} \\ \dot{v}_{iL_k} \end{pmatrix} = \frac{1}{dt} \begin{pmatrix} u_{iL_{k+1}} - u_{iL_k} \\ v_{iL_{k+1}} - v_{iL_k} \end{pmatrix} \\ \dot{\mathbf{w}}_{iR_k} &= \begin{pmatrix} \dot{u}_{iR_k} \\ \dot{v}_{iR_k} \end{pmatrix} = \frac{1}{dt} \begin{pmatrix} u_{iR_{k+1}} - u_{iR_k} \\ v_{iR_{k+1}} - v_{iR_k} \end{pmatrix} \end{aligned} \quad (5.31)$$

Definito il vettore \mathbf{z} direttamente dalle misure, occorre definire il vettore $\tilde{\mathbf{z}}$ che è correlato allo stato tramite la funzione \mathbf{h} :

$$\tilde{\mathbf{z}}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{r}_k \quad (5.32)$$

dove \mathbf{r}_k è un rumore bianco gaussiano di media nulla con covarianza \mathbf{R} .

Nel paragrafo 2.3 è stata descritta la stereo camera e il suo sistema di riferimento. Si è supposto che il sistema di riferimento C sia posto in corrispondenza del pinhole della camera destra. Nel paragrafo 4.2 sono state introdotte le matrici di rototraslazione tra tutti i sistemi di riferimento, in particolare quella tra I e C . Supponendo che \mathbf{P}_C sia il vettore tra il centro ottico della camera destra e il centro di massa del chaser, la coordinata dell' i -esima feature nel sistema di riferimento C è:

$$\boldsymbol{\rho}_i = \mathbf{R}_T^C(\mathbf{q}) \mathbf{P}_i + \boldsymbol{\rho} + \mathbf{P}_C \quad (5.33)$$

dove \mathbf{P}_i è la coordinata dell' i -esima feature nel sistema di riferimento T , $\boldsymbol{\rho}$ è il vettore traslazione relativa e $\mathbf{R}_T^C(\mathbf{q})$ è la matrice di rotazione tra T e C . Note dalla calibrazione e dai parametri costruttivi la lunghezza focale f e la baseline b , è possibile proiettare $\boldsymbol{\rho}_i$ sui due piani immagine:

$$\begin{aligned}\tilde{\mathbf{w}}_{iL} &= \begin{pmatrix} \tilde{u}_{iL} \\ \tilde{v}_{iL} \end{pmatrix} = \begin{pmatrix} f \frac{\rho_{ix} + b}{\rho_{iz}} \\ f \frac{\rho_{iy}}{\rho_{iz}} \end{pmatrix} \\ \tilde{\mathbf{w}}_{iR} &= \begin{pmatrix} \tilde{u}_{iR} \\ \tilde{v}_{iR} \end{pmatrix} = \begin{pmatrix} f \frac{\rho_{ix}}{\rho_{iz}} \\ f \frac{\rho_{iy}}{\rho_{iz}} \end{pmatrix}\end{aligned}\tag{5.34}$$

Tramite dei passaggi matematici, è possibile dimostrare che le derivate rispetto al tempo di 5.34 è:

$$\begin{aligned}\dot{\tilde{\mathbf{w}}}_{iL} &= \begin{pmatrix} \dot{\tilde{u}}_{iL} \\ \dot{\tilde{v}}_{iL} \end{pmatrix} = \begin{bmatrix} -\frac{f}{\rho_{iz}} & 0 & \frac{\tilde{u}_{iL}}{\rho_{iz}} & \frac{\tilde{u}_{iL}\tilde{v}_{iL}}{f} & -f - \frac{\tilde{u}_{iL}^2}{f} & \tilde{v}_{iL} \\ 0 & -\frac{f}{\rho_{iz}} & \frac{\tilde{v}_{iL}}{\rho_{iz}} & f + \frac{\tilde{v}_{iL}^2}{f} & -\frac{\tilde{u}_{iL}\tilde{v}_{iL}}{f} & -\tilde{u}_{iL} \end{bmatrix} \begin{pmatrix} \dot{\boldsymbol{\rho}}^{(L)} \\ \boldsymbol{\omega}^{(L)} \end{pmatrix} \\ \dot{\tilde{\mathbf{w}}}_{iR} &= \begin{pmatrix} \dot{\tilde{u}}_{iR} \\ \dot{\tilde{v}}_{iR} \end{pmatrix} = \begin{bmatrix} -\frac{f}{\rho_{iz}} & 0 & \frac{\tilde{u}_{iR}}{\rho_{iz}} & \frac{\tilde{u}_{iR}\tilde{v}_{iR}}{f} & -f - \frac{\tilde{u}_{iR}^2}{f} & \tilde{v}_{iR} \\ 0 & -\frac{f}{\rho_{iz}} & \frac{\tilde{v}_{iR}}{\rho_{iz}} & f + \frac{\tilde{v}_{iR}^2}{f} & -\frac{\tilde{u}_{iR}\tilde{v}_{iR}}{f} & -\tilde{u}_{iR} \end{bmatrix} \begin{pmatrix} \dot{\boldsymbol{\rho}}^{(R)} \\ \boldsymbol{\omega}^{(R)} \end{pmatrix}\end{aligned}\tag{5.35}$$

dove $\dot{\boldsymbol{\rho}}^{(L)}$ e $\boldsymbol{\omega}^{(L)}$ sono le componenti di velocità traslazionale e rotazionale rispetto alla camera sinistra, mentre $\dot{\boldsymbol{\rho}}^{(R)}$ e $\boldsymbol{\omega}^{(R)}$ sono le componenti di velocità traslazionale e rotazionale rispetto alla camera destra. Siccome il centro del sistema di riferimento è stato scelto coincidente con il pinhole della camera destra, le componenti di velocità della camera sinistra devono essere correlate a quelle della camera destra. Dato che le camere sono allineate, la matrice di rotazione è uguale alla matrice identità. Inoltre, essendo $\dot{\boldsymbol{\rho}}$ e $\boldsymbol{\omega}$ le derivate della posizione e dell'assetto, poiché \mathbf{R}_{CL}^{CR} è la matrice identità, è possibile affermare che $\dot{\boldsymbol{\rho}}^{(L)} = \dot{\boldsymbol{\rho}}^{(R)}$ e $\boldsymbol{\omega}^{(L)} = \boldsymbol{\omega}^{(R)}$, quindi (5.35)

può essere riscritta:

$$\begin{aligned}
\dot{\mathbf{w}}_{iL} = \begin{pmatrix} \dot{\tilde{u}}_{iL} \\ \dot{\tilde{v}}_{iL} \end{pmatrix} &= \begin{bmatrix} -\frac{f}{\rho_{iz}} & 0 & \frac{\tilde{u}_{iL}}{\rho_{iz}} & \frac{\tilde{u}_{iL}\tilde{v}_{iL}}{f} & -f - \frac{\tilde{u}_{iL}^2}{f} & \tilde{v}_{iL} \\ 0 & -\frac{f}{\rho_{iz}} & \frac{\tilde{v}_{iL}}{\rho_{iz}} & f + \frac{\tilde{v}_{iL}^2}{f} & -\frac{\tilde{u}_{iL}\tilde{v}_{iL}}{f} & -\tilde{u}_{iL} \end{bmatrix} \begin{pmatrix} \dot{\boldsymbol{\rho}} \\ \boldsymbol{\omega} \end{pmatrix} \\
\dot{\mathbf{w}}_{iR} = \begin{pmatrix} \dot{\tilde{u}}_{iR} \\ \dot{\tilde{v}}_{iR} \end{pmatrix} &= \begin{bmatrix} -\frac{f}{\rho_{iz}} & 0 & \frac{\tilde{u}_{iR}}{\rho_{iz}} & \frac{\tilde{u}_{iR}\tilde{v}_{iR}}{f} & -f - \frac{\tilde{u}_{iR}^2}{f} & \tilde{v}_{iR} \\ 0 & -\frac{f}{\rho_{iz}} & \frac{\tilde{v}_{iR}}{\rho_{iz}} & f + \frac{\tilde{v}_{iR}^2}{f} & -\frac{\tilde{u}_{iR}\tilde{v}_{iR}}{f} & -\tilde{u}_{iR} \end{bmatrix} \begin{pmatrix} \dot{\boldsymbol{\rho}} \\ \boldsymbol{\omega} \end{pmatrix}
\end{aligned} \tag{5.36}$$

L'ultimo parametro da stimare è quello riguardante la disparity, facilmente calcolabile se sono note le coordinate u delle features sui piani immagine:

$$\tilde{d}_i = \tilde{u}_{iL} - \tilde{u}_{iR} \tag{5.37}$$

Pertanto, la funzione \mathbf{h} è data da (5.34), (5.36) e (5.37). Si osserva la forte non linearità della funzione \mathbf{h} . In particolare, (5.34) è legata dalla matrice di rotazione (quindi da \mathbf{q}) e dalla traslazione relativa. Le proiezioni sul piano immagine \tilde{u} e \tilde{v} , insieme a ρ_z , fanno parte della matrice che premoltiplica il vettore delle velocità $(\dot{\boldsymbol{\rho}}, \boldsymbol{\omega})^T$. Il calcolo dello Jacobiano è molto complesso, pertanto deve essere fatto in maniera numerica, per esempio utilizzando il metodo delle differenze finite.

Capitolo 6

Risultati dei test sperimentali

Nel corso del seguente capitolo saranno descritti i risultati ottenuti per la stima del moto relativo tra chaser e target, tenendo in considerazione diverse velocità di rotazione del target analizzate con diverse tecniche, in maniera tale da confrontare la bontà degli algoritmi e vedere le principali differenze emerse.

Come visto nel paragrafo 4.1, il target è stato posizionato su di uno stadio rotativo motorizzato. Sono stati quindi eseguiti in totale sei test sperimentali, due di moto stazionario e quattro di moto rotazionale, le cui caratteristiche sono visibili rispettivamente nella Tab. 6.1 e nella Tab. 6.2.

I moti simulati sono stati studiati con diverse tecniche di analisi delle immagini, in particolare le frequenza di acquisizione delle immagini e la disposizione delle features sul target. Il descrittore scelto per ogni simulazione è stato SURF, che come descritto nel paragrafo 3.2.2 offre diversi vantaggi.

Il passaggio iniziale, dopo aver trovato le features in ogni frame per entrambe le camere ed avere eseguito il matching, è stato quello di eliminare gli outliers, come descritto nel paragrafo 3.4. Oltre ad aver applicato il filtraggio RANSAC per eliminare le features che non appartengono al piano epipolare, sono state rifiutate anche altre corrispondenze non appartenenti al target. Infatti, anche dopo l'applicazione di una ROI per ricercare features del target, è possibile che l'algoritmo di ricerca trovi le features che appartengono allo sfondo della scena che sono fonte di rumore per l'algoritmo di stima del moto. Per ovviare a questo problema, sono state

rifiutate corrispondenze che si trovano ad una distanza maggiore di quella del target. Da un punto di vista applicativo questa è una considerazione importante. Infatti, in uno scenario di moto relativo reale nello spazio, può capitare che anche la superficie terrestre entri nel campo di vista delle camere. Le features appartenenti alla Terra, una volta triangolate, saranno a distanza molto maggiore rispetto a quella di un satellite target, e quindi ponendo questa threshold sulla distanza è possibile eliminarle.

TEST		01	02
POSIZIONE [m]	t_x	-0.0136	-0.0136
	t_y	-1.1118	-1.1118
	t_z	-0.1479	-0.1479
ASSETTO [deg]	ϕ	3.1537	-6.9230
	θ	-5.0466	3.6485
	ψ	-138.5685	18.5965
TEMPO DI ACQUISIZIONE [s]		29.7	29.5

Tabella 6.1: Caratteristiche dei due test stazionari. La posa è espressa nel sistema di riferimento del chaser e calcolata usando il Motion Capture.

TEST		03	04	05	06
POSIZIONE [m]	t_x	-0.0136	-0.0136	-0.0136	-0.0136
	t_y	-1.1118	-1.1118	-1.1118	-1.1118
	t_z	-0.1479	-0.1479	-0.1479	-0.1479
ASSETTO INIZIALE [deg]	ϕ	5.6313	5.6313	5.6313	2.9736
	θ	-0.0576	-0.0576	-0.0576	-4.9133
	ψ	167	167	167	-143
ASSETTO FINALE [deg]	ϕ	5.6313	5.6313	2.9736	0.2143
	θ	-0.0576	-0.0576	-4.9133	-5.6522
	ψ	167	167	-143	-113
VELOCITÀ ANGOLARE [deg/s]	ω_x	0	0	0	0
	ω_y	0	0	0	0
	ω_z	12.5	-12.5	1.5625	1
TEMPO DI ACQUISIZIONE [s]		28.8	28.8	32.1	27.5

Tabella 6.2: Caratteristiche dei quattro test rotazionali. La posa è espressa nel sistema di riferimento del chaser e calcolata usando il Motion Capture.

Sono state considerate cinque diverse tipologie di analisi delle immagini, che sono state poi confrontate, il quale unico vincolo comune è che le features siano

presenti al frame i e al frame $i + 1$, necessario per la stima della velocità delle features come visto nel paragrafo 5.3. Esse sono:

- *Scenario 1*: utilizzo di tutte le features trovate con l'algoritmo SURF;
- *Scenario 2*: utilizzo di tutte le features trovate con l'algoritmo SURF ma con una diversa frequenza di acquisizione;
- *Scenario 3*: utilizzo di features ben distribuite sulla superficie del target;
- *Scenario 4*: utilizzo di features concentrate in una specifica regione sulla superficie del target;
- *Scenario 5*: utilizzo di tutte le features trovate con l'algoritmo SURF ma con condizioni iniziali su posizione ed assetto relativo più distanti da quelle reali;

Si sottolinea come nei primi quattro scenari le condizioni iniziali in input all'EKF siano state scelte molto vicine a quelle reali. Lo scenario 5 è necessario per poter valutare che l'algoritmo non sia eccessivamente sensibile alle condizioni iniziali, sia in termini di posizioni e sia in termini di assetto. L'errore nell'inizializzazione può essere non trascurabile, soprattutto per la rotazione. Infatti, piccoli errori nella ricerca dei vertici del pannello solare del target possono produrre errori non indifferenti.

Precedentemente sono state citate tutte le simulazioni eseguite. Verranno però presentati i risultati ottenuti per tutti e cinque gli scenari ma relativamente a tre test sperimentali: uno stazionario (test 01), uno rotazionale con velocità angolare massima del target (test 03) ed un altro rotazionale con velocità angolare più bassa del target (test 06).

6.1 Scenario 1

Nel seguente scenario, tutte le features identificate dall'algoritmo sono state prese in considerazione. In funzione delle condizioni di luminosità, l'effettivo numero di features varia. Se il satellite è ripreso come nella Fig. 6.1 produce un numero molto alto di features. Viceversa, se fosse ripreso dalla parte posteriore il numero sarebbe

più basso in quanto la riflessione rende più difficoltosa la ricerca delle features. Per tutte le simulazioni eseguite, le matrici di covarianza dell'errore di misura \mathbf{R} e di processo \mathbf{Q} sono le stesse, così come la matrice di covarianza dell'errore iniziale di stima $\mathbf{P}_{0|0}$; quest'ultima varia in funzione dello scenario per l'analisi delle immagini scelto.

La matrice \mathbf{R} dipende dall'errore di riproiezione delle features triangolate, il cui valore medio è di circa 5 pixel. Siccome tutto lo studio è stato fatto in coordinate metriche, l'errore in pixel è stato convertito in errore in metri, la cui equazione per la conversione delle coordinate è possibile in quanto i parametri intrinseci sono noti dalla calibrazione, come visto nel paragrafo 2.3.2. L'errore riferito alle componenti di velocità delle features è più alto, in quanto viene propagato nel tempo.

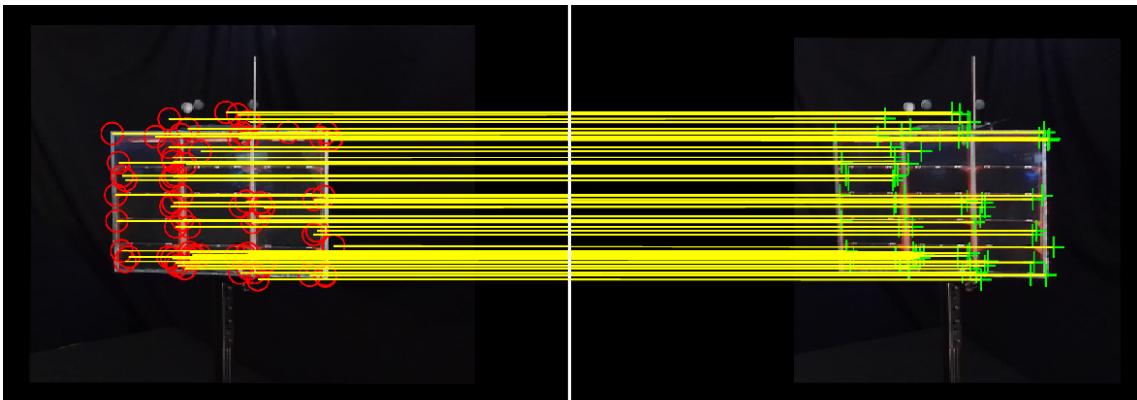


Figura 6.1: Scenario 1: selezione di tutte le features trovate con l'algoritmo SURF.

La matrice \mathbf{Q} viene stimata in una fase di tuning dell'EKF, in cui vengono presi valori coerenti con la stima dello stato. La matrice $\mathbf{P}_{0|0}$ viene scelta in funzione degli errori iniziali, che in questo scenario sono piccoli in quanto il vettore di stato iniziale scelto è piuttosto vicino al vettore di stato reale.

I valori delle matrici e dello stato iniziale sono riassunti rispettivamente nelle Tab. 6.3 e Tab. 6.4. I grafici degli errori per posizione relativa, velocità relativa, assetto relativo e velocità angolare relativa sono visibili nelle Fig. 6.4, Fig. 6.5 e Fig. 6.6, sotto le quali sono presenti le Tab. 6.6, Tab. 6.7 e Tab. 6.8 con gli errori rms.

R	$\mathbf{R}_{u_iL, v_iL, u_iR, v_iR}$ [m ²]	diag (1, 1, 1, 1) $\times 10^{-6}$
	$\mathbf{R}_{u_iL, v_iL, u_iR, v_iR}$ [m ² /s ²]	diag (300, 300, 300, 300) $\times 10^{-6}$
	\mathbf{R}_{d_i} [m ²]	diag (1) $\times 10^{-6}$
Q	\mathbf{Q}_t [m ²]	diag (1, 1, 1) $\times 10^{-4}$
	\mathbf{Q}_v [m ² /s ²]	diag (1, 1, 1) $\times 10^{-4}$
	\mathbf{Q}_q	diag (1, 1, 1, 1) $\times 10^{-4}$
	\mathbf{Q}_ω [rad ² /s ²]	diag (1, 1, 1) $\times 10^{-4}$
P_{0 0}	$\mathbf{P}_{0 0,t}$ [m ²]	diag (1, 1, 1) $\times 10^{-2}$
	$\mathbf{P}_{0 0,v}$ [m ² /s ²]	diag (1, 1, 1) $\times 10^{-2}$
	$\mathbf{P}_{0 0,q}$	diag (1, 1, 1, 1) $\times 10^{-2}$
	$\mathbf{P}_{0 0,\omega}$ [rad ² /s ²]	diag (1, 1, 1) $\times 10^{-2}$

Tabella 6.3: Matrici iniziali dell'EKF per lo scenario 1 per tutti i test.

TEST		01	03	06
t_0 [m]	$t_{x,0}$	0.2	0.2	0.2
	$t_{y,0}$	0.15	0.15	0.15
	$t_{z,0}$	1.15	1.15	1.15
v_0 [m/s]	$v_{x,0}$	0	0	0
	$v_{y,0}$	0	0	0
	$v_{z,0}$	0	0	0
α_0 [deg]	ϕ_0	7	8	7
	θ_0	-9	1	-8
	ψ_0	-135	163	-141
q_0	$q_{0,0}$	0.3852	0.1468	0.3283
	$q_{1,0}$	0.9175	0.9866	-0.9400
	$q_{2,0}$	-0.0956	-0.0017	0.0453
	$q_{3,0}$	0.0262	0.0703	-0.0806
ω_0 [deg/s]	$\omega_{x,0}$	0	0	0
	$\omega_{y,0}$	0	0	0
	$\omega_{z,0}$	0	12.55	1.1

Tabella 6.4: Stato iniziale dello scenario 1.

6.2 Scenario 2

Nello scenario 1, sono stati analizzati tutti i frame della stereo camera, e lo stato dell'EKF è stato aggiornato con la stessa frequenza di acquisizione pari a 15 Hz. Poiché l'EKF propaga lo stato del sistema linearizzando le equazioni del moto, non è opportuno diminuire la frequenza di propagazione. Se si lasciasse invariata la frequenza di acquisizione e si aumentasse quella di propagazione, la stima sarebbe più precisa ma con un costo computazionale decisamente maggiore. Pertanto, in questo scenario si è deciso di diminuire la frequenza di aggiornamento delle misure

passando da 15 Hz a 1 Hz e lasciando propagare lo stato sempre a 15 Hz.

Nei casi stazionari questa modifica non si ripercuote sulla stima, in quanto il target rimane fermo e le features trovate sono sempre quelle. Il problema principale si ha nei casi rotazionali, in cui avendo meno frame è possibile che molte features vengano perse, e la quantità di features perse dipende strettamente dalla velocità di rotazione imposta al target. Dunque, modificare la frequenza di measurement update si riflette principalmente sulla bontà della stima della rotazione relativa e non tanto sulla traslazione relativa.

Le condizioni iniziali del filtro e la matrice di covarianza di misura \mathbf{R} sono le stesse dello scenario 1. La matrice \mathbf{Q} è tenuta costante, ciò che varia è solo la matrice $\mathbf{P}_{0|0}$, ma non per tutti i test. Infatti, nel caso stazionario (test 01) e nel caso rotazionale lento (test 06), un diverso measurement update non è così determinante come per il test 03, in cui si ha una perdita considerevole del numero di features. A ragione di ciò, la matrice $\mathbf{P}_{0|0}$ è la stessa per il test 01 e per il test 06, mentre variano le componenti rotazionali per il test 03, che assumono i seguenti valori:

- $\mathbf{P}_{0|0,q} = \text{diag}(1, 1, 1, 1) \times 10^{-1}$
- $\mathbf{P}_{0|0,\omega} = \text{diag}(0.9, 0.9, 6) \times 10^{-3} \frac{\text{rad}^2}{\text{s}^2}$

I grafici degli errori per posizione relativa, velocità relativa, assetto relativo e velocità angolare relativa sono visibili nelle Fig. 6.7, Fig. 6.8 e Fig. 6.9, sotto le quali sono presenti le Tab. 6.9, Tab. 6.10 e Tab. 6.11 con gli errori rms.

6.3 Scenario 3

Nello scenario 1, come visto, sono state scelte tutte le features che l'algoritmo di ricerca è stato in grado di rilevare. Ovviamente, maggiore è il numero di features e migliore sarà la misura. D'altro canto, aumentando il numero di features, l'algoritmo di stima del moto può diventare più oneroso dal punto di vista computazionale e quindi si è voluto valutare la bontà della stima del moto relativo scegliendo un numero minore di features ma che ricoprono al meglio la superficie del target.

Scegliendo una features in modo casuale sulla superficie del target, sono

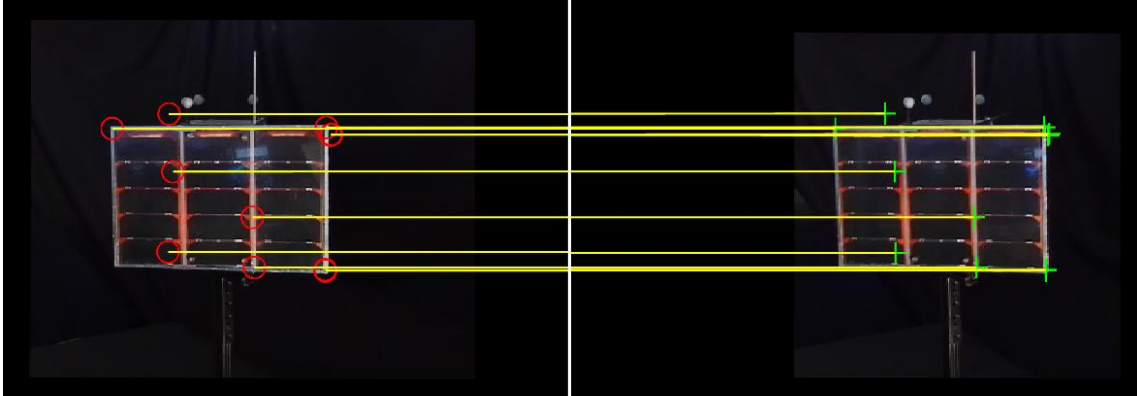


Figura 6.2: Scenario 3: selezione delle features ben distribuite sul target.

state calcolate tutte le distanze dalle altre features e poi sono state scelte le features a distanza maggiore. Il risultato ottenuto, come visibile nella Fig. 6.2, è stato buono.

Dal punto di vista dell'applicazione dell'EKF, l'inizializzazione è la stessa dello scenario 1, in quanto la situazione è molto simile, con l'unica differenza data dal fatto che il numero di features è minore. Tuttavia, essendo le features disposte in maniera abbastanza uniforme, la stima del moto relativo non si discosta di molto da quella ottenuta nello scenario 1. Questo scenario evidenzia che la stessa bontà nella stima del moto relativo può essere ottenuta con un minore numero di features e di conseguenza un minor costo computazionale dell'EKF.

I grafici degli errori per posizione relativa, velocità relativa, assetto relativo e velocità angolare relativa sono visibili nelle Fig. 6.10, Fig. 6.11 e Fig. 6.12, sotto le quali sono presenti le Tab. 6.12, Tab. 6.13 e Tab. 6.14 con gli errori rms.

6.4 Scenario 4

Se nello scenario 3 sono state cercate features distribuite in maniera uniforme sul target, si è provato a vedere cosa succede se le features sono concentrate in una specifica zona del target. La metodologia per trovare queste features è la stessa dello scenario 3, soltanto che in questo caso sono state considerate le distanze minori tra esse, come visibile nella Fig. 6.3.

La principale differenza rispetto agli scenari 1 e 3 riguarda la più difficile stima della rotazione relativa, il che lo rende molto simile allo scenario 2. Infatti,

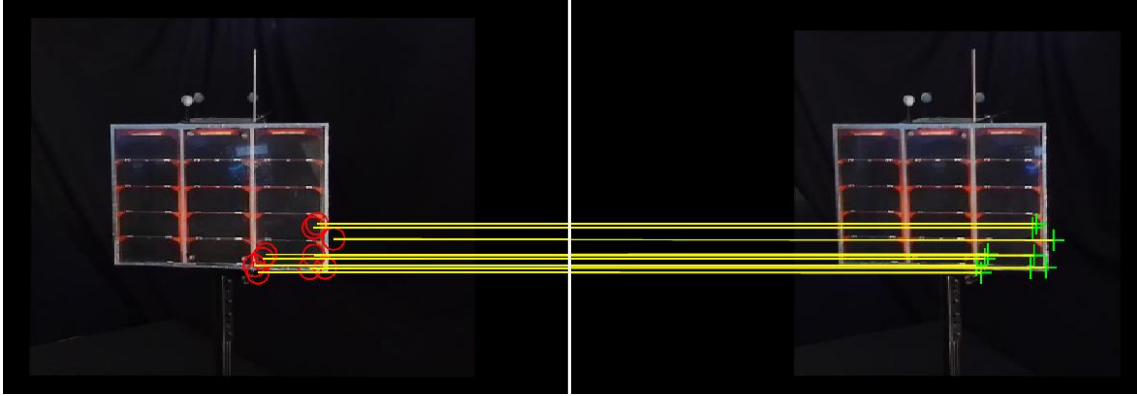


Figura 6.3: Selezione delle features concentrate sul target.

essendo il target di ridotte dimensioni, la traslazione relativa non è eccessivamente intaccata dal fatto che le features sono mal distribuite, ma ciò non vale per la rotazione. A ragione di ciò, le principali differenze riguardano l'inizializzazione dell'EKF, similmente allo scenario 2. Tenendo costanti le condizioni iniziali, la matrice \mathbf{R} e la matrice \mathbf{Q} , ciò che varia sono le componenti rotazionali della matrice $\mathbf{P}_{0|0}$ per tutti i test, che assumono i seguenti valori:

- $\mathbf{P}_{0|0,q} = \text{diag}(3, 3, 3, 3) \times 10^{-1}$
- $\mathbf{P}_{0|0,\omega} = \text{diag}(0.1, 0.1, 1) \times 10^{-2} \frac{\text{rad}^2}{\text{s}^2}$

I grafici degli errori per posizione relativa, velocità relativa, assetto relativo e velocità angolare relativa sono visibili nelle Fig. 6.13, Fig. 6.14 e Fig. 6.15, sotto le quali sono presenti le Tab. 6.15, Tab. 6.16 e Tab. 6.17 con gli errori rms.

6.5 Scenario 5

Nei precedenti 4 scenari, le condizioni iniziali in termini vettore di stato sono state scelte abbastanza vicine al valore reale dopo aver eseguito una buona inizializzazione, descritta nel paragrafo 5.1.1. Tuttavia, può capitare che l'inizializzazione non sia estremamente precisa e quindi gli errori di stima iniziale possono essere più grandi. Questa difficoltà nell'inizializzazione è maggiore soprattutto per quanto riguarda la rotazione relativa.

A ragione di ciò, in questo scenario sono state modificate le condizioni

iniziali in termini di posizione iniziale ed assetto iniziale, mentre le componenti di velocità sono state mantenute costanti. La tecnica per rilevare le features è la stessa dello scenario 1, ovvero tenere in considerazione tutte le features che l'algoritmo di ricerca riesce a rilevare. Le nuove condizioni iniziali sono visibili nella Tab 6.5.

TEST		01	03	06
\mathbf{t}_0 [m]	$t_{x,0}$	0.1	0.1	0.1
	$t_{y,0}$	0.05	0.05	0.05
	$t_{z,0}$	1.25	1.25	1.25
\mathbf{v}_0 [m/s]	$v_{x,0}$	0	0	0
	$v_{y,0}$	0	0	0
	$v_{z,0}$	0	0	0
$\boldsymbol{\alpha}_0$ [deg]	ϕ_0	17	18	17
	θ_0	-19	-9	-18
	ψ_0	-125	173	-131
\mathbf{q}_0	$q_{0,0}$	0.4288	0.0724	0.3840
	$q_{1,0}$	-0.8765	0.9820	-0.8985
	$q_{2,0}$	0.0775	-0.0869	0.0802
	$q_{3,0}$	-0.2047	0.1509	-0.1970
$\boldsymbol{\omega}_0$ [deg/s]	$\omega_{x,0}$	0	0	0
	$\omega_{y,0}$	0	0	0
	$\omega_{z,0}$	0	12.55	1.1

Tabella 6.5: Stato iniziale dello scenario 5.

In questo scenario, la matrice \mathbf{R} e la matrice \mathbf{Q} agli altri scenari, mentre variano le condizioni iniziali e la matrice $\mathbf{P}_{0|0}$. Nonostante si prenda un errore iniziale sulla traslazione maggiore, questo non si ripercuote sulle rispettive componenti della matrice $\mathbf{P}_{0|0}$, in quanto i valori presi in precedenza sono sufficienti a mantenere l'errore di stima all'interno della banda di incertezza. L'unica differenza rispetto allo scenario 1 si ha in corrispondenza della componente di assetto relativo della matrice $\mathbf{P}_{0|0}$, ovvero $\mathbf{P}_{0|0,q} = \text{diag}(2, 2, 2, 2) \times 10^{-1}$.

I grafici degli errori per posizione relativa, velocità relativa, assetto relativo e velocità angolare relativa sono visibili nelle Fig. 6.16, Fig. 6.17 e Fig. 6.18, sotto le quali sono presenti le Tab. 6.18, Tab. 6.19 e Tab. 6.20 con gli errori rms.

6.6 Grafici dei risultati sperimentali

Nei precedenti paragrafi sono stati descritti gli scenari considerati, i quali risultati verranno mostrati all'interno di questo paragrafo, con relative tabelle riguardanti gli errori rms. Nello scenario 5, quello con differenti condizioni iniziali, gli errori rms sono riferiti a quando il filtro di Kalman va a convergenza. In particolare, siccome non tutte le variabili di moto vanno a convergenza nello stesso tempo, il tempo di convergenza considerato sarà quello dove tutte le variabili sono nell'intorno dell'errore nullo. Nei paragrafi 6.7 e 6.8 verranno commentati i risultati ottenuti.

Le condizioni iniziali, elencate nella Tab. 6.4 per i primi 4 scenari e nella Tab. 6.5 per lo scenario 5, sono espresse in coordinate camera, il cui sistema di riferimento è stato descritto in precedenza. Siccome il Motion Capture misura il moto relativo tra chaser e target e non tra stereo camera e target, per calcolare l'errore nella stima del moto tramite l'EKF occorre eseguire una rototraslazione, resa possibile in quanto la matrice \mathbf{T}_C^I è nota. Una volta noto l'andamento delle variabili nello stesso sistema di riferimento, è possibile calcolare l'errore della generica variabile x come $\epsilon_x = x_{MC} - x_{EKF}$, dove x_{MC} è la stima della variabile data dal Motion Capture, cioè lo strumento fiduciario, mentre x_{EKF} è la stima della variabile data dall'EKF. Nei grafici successivi sarà rappresentato l'errore ϵ_x con la relativa banda di incertezza a 3σ , ricavata dalla matrice \mathbf{P} come descritto nel paragrafo 5.1.3.

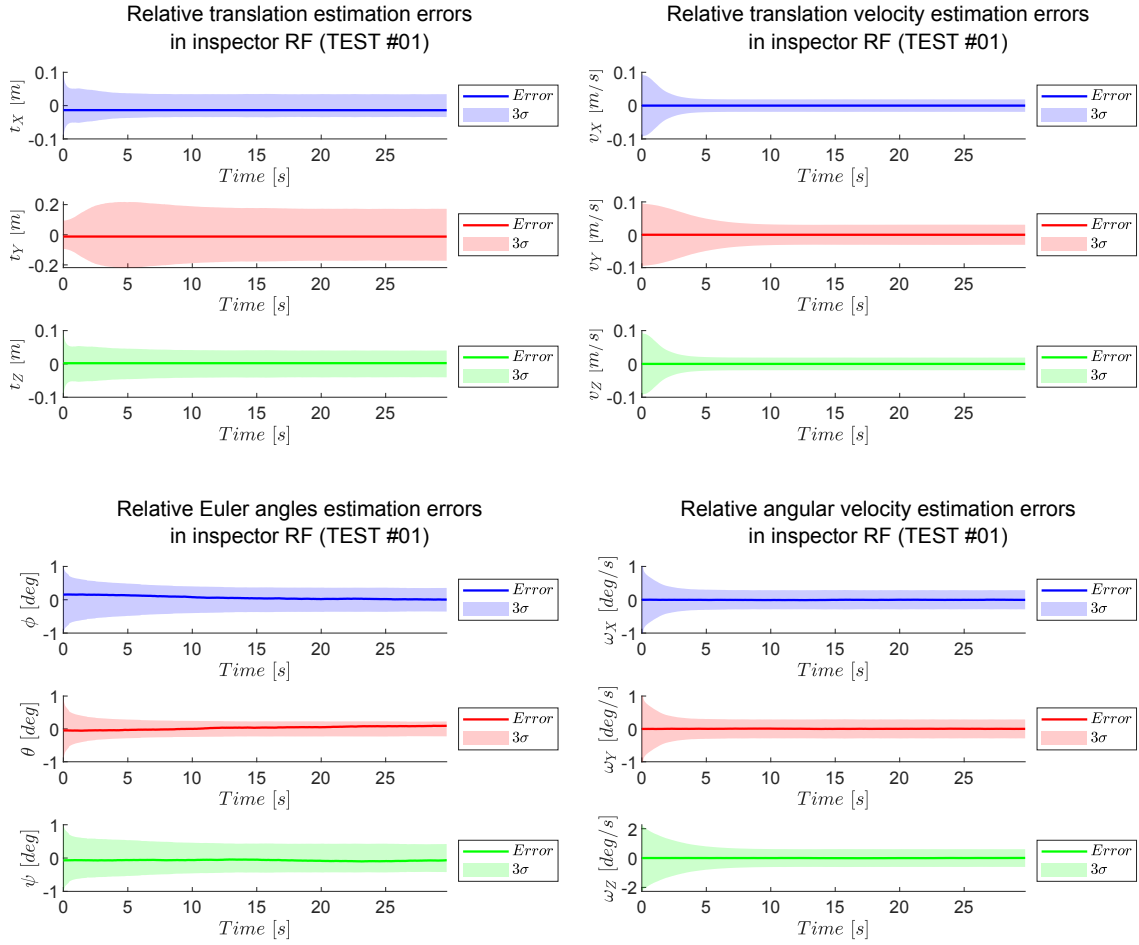


Figura 6.4: Errori nello scenario 1 del test 01.

TRASLAZIONE [m]	ϵ_{t_x}	0.0137	ϵ_t	0.0106
	ϵ_{t_y}	0.0120		
	ϵ_{t_z}	0.0022		
VELOCITÀ TRASLAZIONALE [m/s]	ϵ_{v_x}	1.2811E-05	ϵ_v	1.3810E-05
	ϵ_{v_y}	1.3145E-05		
	ϵ_{v_z}	1.5336E-05		
ROTAZIONE [deg]	ϵ_ϕ	0.0811	ϵ_α	0.0698
	ϵ_θ	0.0547		
	ϵ_ψ	0.0709		
VELOCITÀ ROTAZIONALE [deg/s]	ϵ_{ω_x}	0.0046	ϵ_ω	0.0042
	ϵ_{ω_y}	0.0040		
	ϵ_{ω_z}	0.0039		

Tabella 6.6: Errori rms nello scenario 1 del test 01.

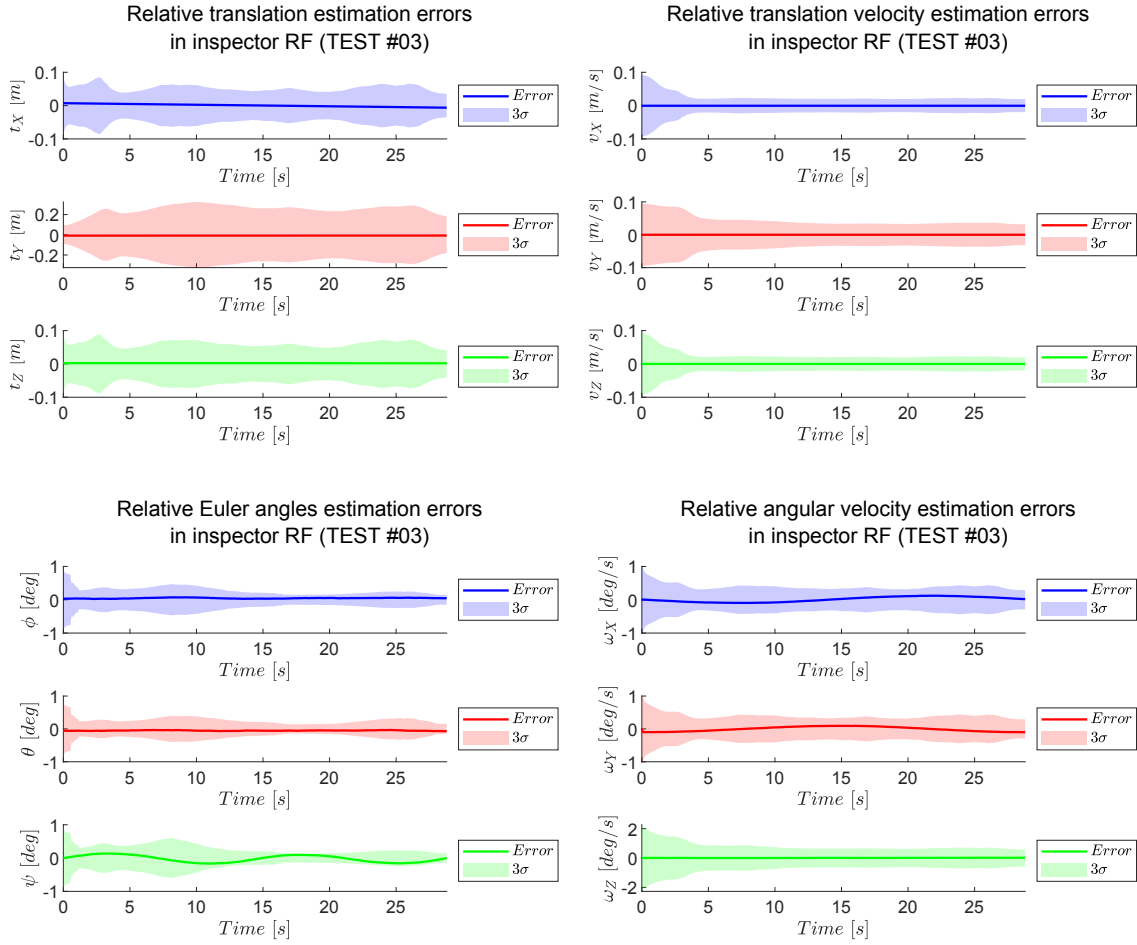


Figura 6.5: Errori nello scenario 1 del test 03.

TRASLAZIONE [m]	ϵ_{t_x}	0.0039	ϵ_t	0.0054
	ϵ_{t_y}	0.0081		
	ϵ_{t_z}	0.0022		
VELOCITÀ TRASLAZIONALE [m/s]	ϵ_{v_x}	4.6590E-04	ϵ_v	2.7054E-04
	ϵ_{v_y}	2.6851E-05		
	ϵ_{v_z}	4.2376E-05		
ROTAZIONE [deg]	ϵ_ϕ	0.0487	ϵ_α	0.0715
	ϵ_θ	0.0517		
	ϵ_ψ	0.1015		
VELOCITÀ ROTAZIONALE [deg/s]	ϵ_{ω_x}	0.0765	ϵ_ω	0.0596
	ϵ_{ω_y}	0.0683		
	ϵ_{ω_z}	0.0120		

Tabella 6.7: Errori rms nello scenario 1 del test 03.

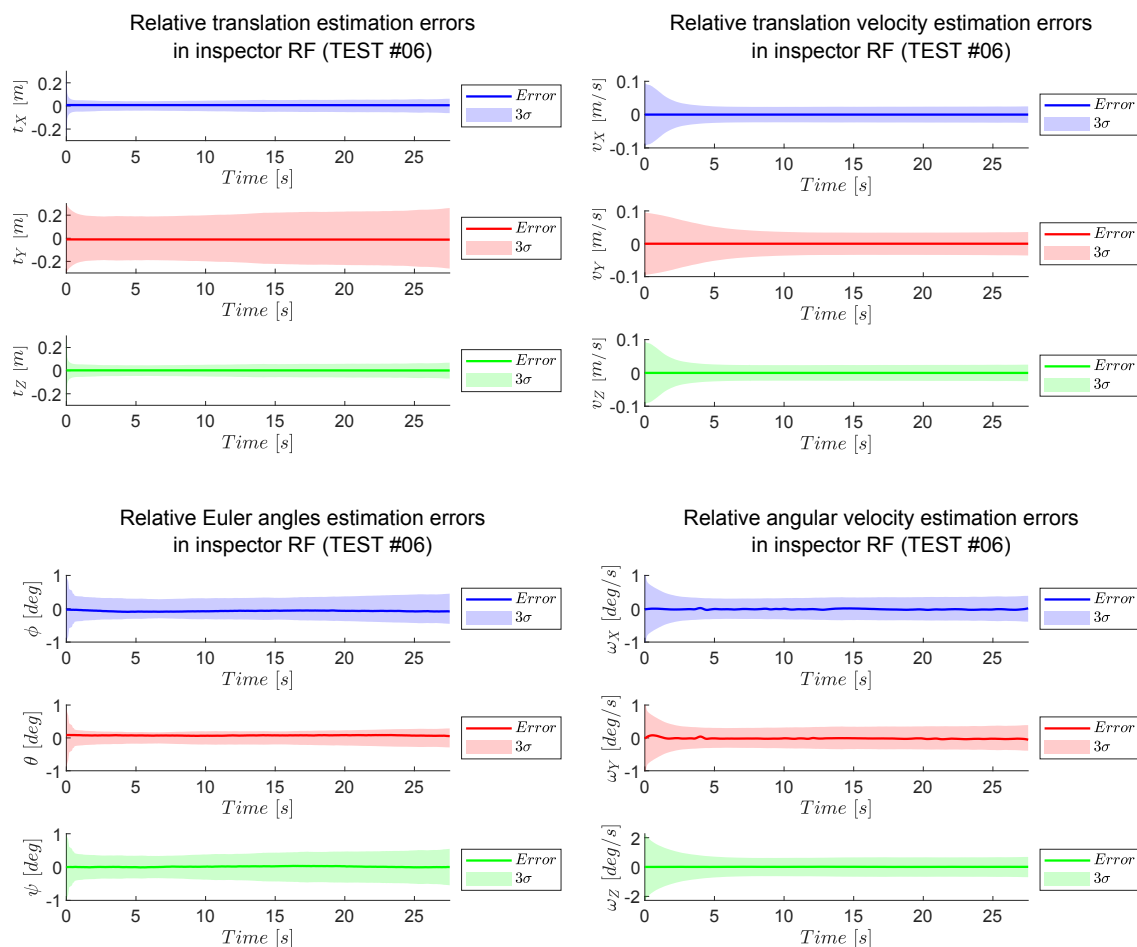


Figura 6.6: Errori nello scenario 1 del test 06.

TRASLAZIONE [m]	ϵ_{t_x}	0.0070	ϵ_t	0.0075
	ϵ_{t_y}	0.0107		
	ϵ_{t_z}	0.0020		
VELOCITÀ TRASLAZIONALE [m/s]	ϵ_{v_x}	4.4369E-05	ϵ_v	7.1381E-05
	ϵ_{v_y}	1.1443E-04		
	ϵ_{v_z}	1.4948E-05		
ROTAZIONE [deg]	ϵ_ϕ	0.0694	ϵ_α	0.0601
	ϵ_θ	0.0759		
	ϵ_ψ	0.0160		
VELOCITÀ ROTAZIONALE [deg/s]	ϵ_{ω_x}	0.0183	ϵ_ω	0.0197
	ϵ_{ω_y}	0.0255		
	ϵ_{ω_z}	0.0133		

Tabella 6.8: Errori rms nello scenario 1 del test 06.

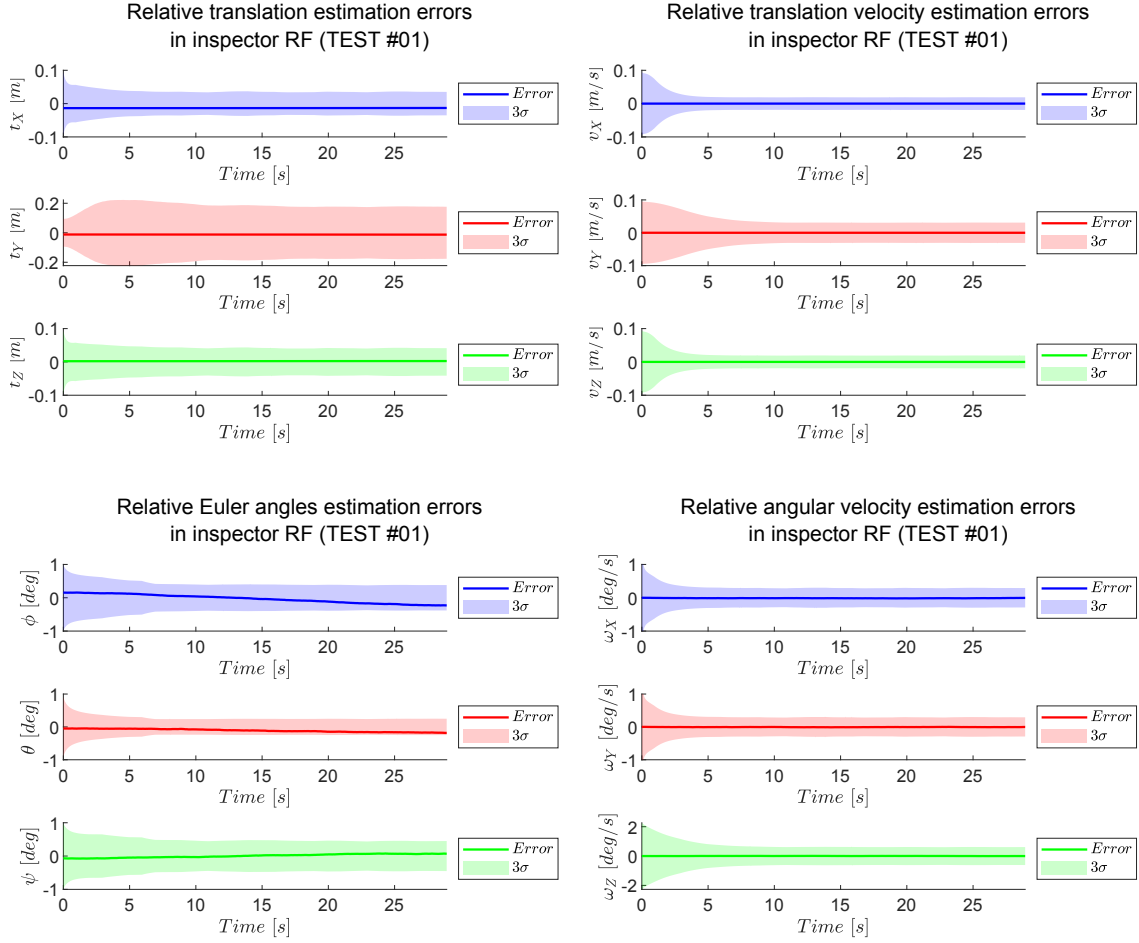


Figura 6.7: Errori nello scenario 2 del test 01.

TRASLAZIONE [m]	ϵ_{t_x}	0.0134	ϵ_t	0.0106
	ϵ_{t_y}	0.0122		
	ϵ_{t_z}	0.0023		
VELOCITÀ TRASLAZIONALE [m/s]	ϵ_{v_x}	2.1119E-05	ϵ_v	2.4275E-05
	ϵ_{v_y}	3.2007E-05		
	ϵ_{v_z}	1.7242E-05		
ROTAZIONE [deg]	ϵ_ϕ	0.0892	ϵ_α	0.0859
	ϵ_θ	0.1079		
	ϵ_ψ	0.0505		
VELOCITÀ ROTAZIONALE [deg/s]	ϵ_{ω_x}	0.0129	ϵ_ω	0.0089
	ϵ_{ω_y}	0.0063		
	ϵ_{ω_z}	0.0054		

Tabella 6.9: Errori rms nello scenario 2 del test 01.

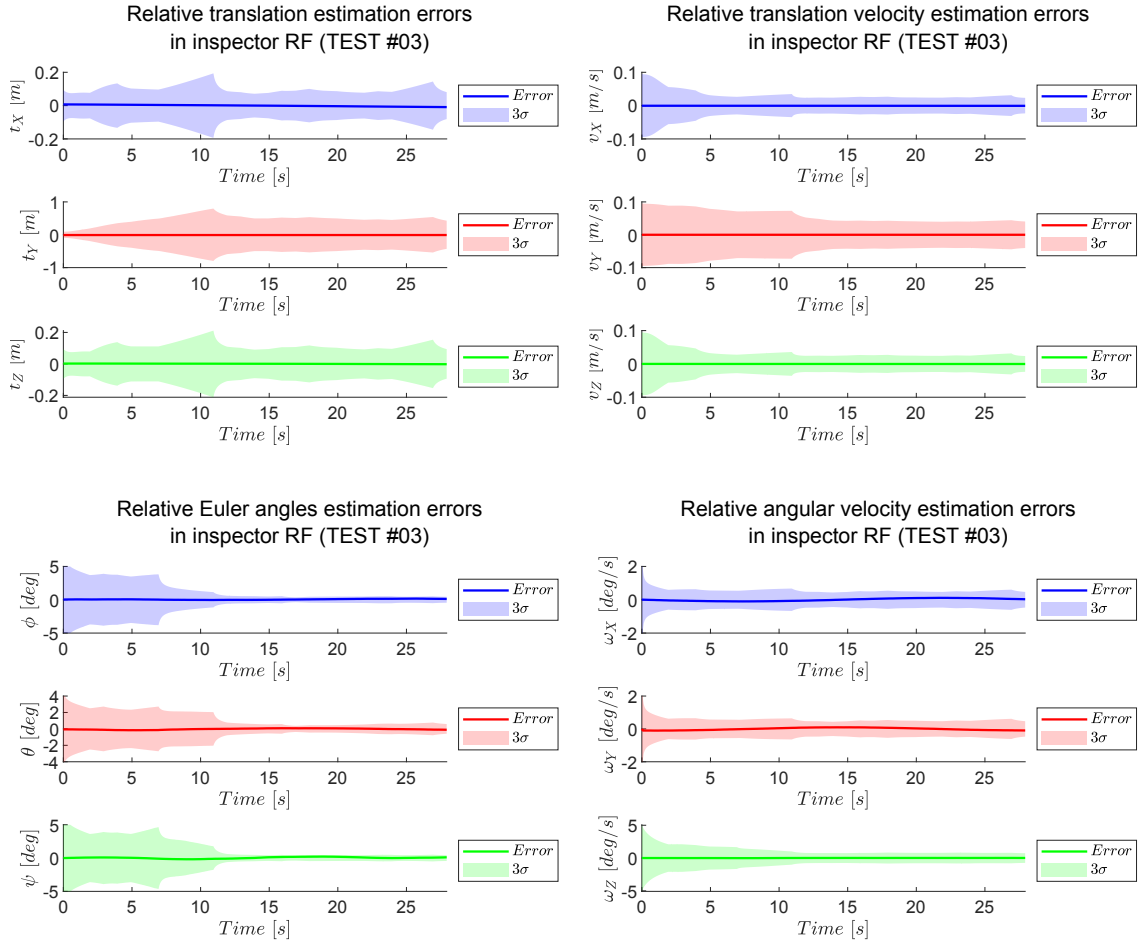


Figura 6.8: Errori nello scenario 2 del test 03.

TRASLAZIONE [m]	ϵ_{t_x}	0.0047	ϵ_t	0.0057
	ϵ_{t_y}	0.0085		
	ϵ_{t_z}	0.0014		
VELOCITÀ TRASLAZIONALE [m/s]	ϵ_{v_x}	5.4876E-04	ϵ_v	3.3042E-04
	ϵ_{v_y}	3.9193E-05		
	ϵ_{v_z}	1.5768E-04		
ROTAZIONE [deg]	ϵ_ϕ	0.1039	ϵ_α	0.1224
	ϵ_θ	0.0992		
	ϵ_ψ	0.1559		
VELOCITÀ ROTAZIONALE [deg/s]	ϵ_{ω_x}	0.0748	ϵ_ω	0.0721
	ϵ_{ω_y}	0.0688		
	ϵ_{ω_z}	0.0727		

Tabella 6.10: Errori rms nello scenario 2 del test 03.

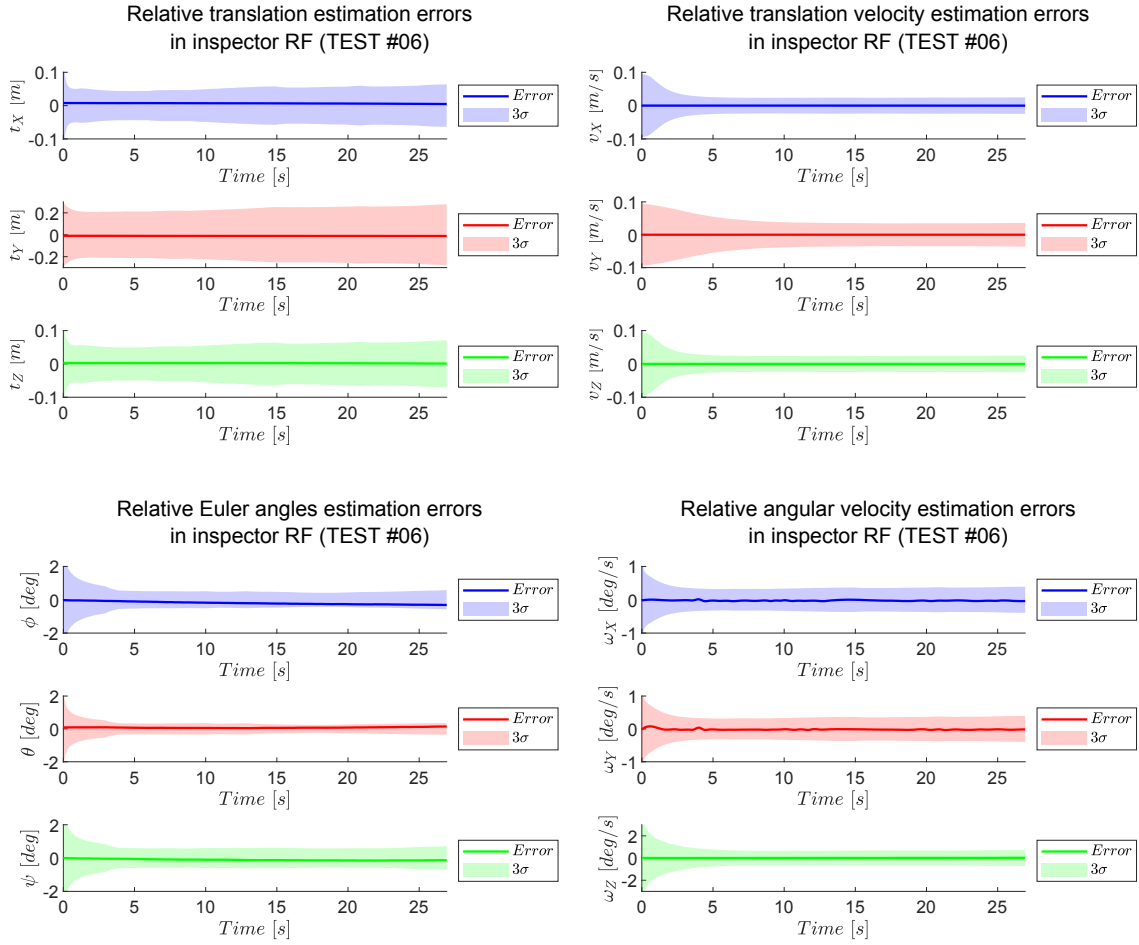


Figura 6.9: Errori nello scenario 2 del test 06.

TRASLAZIONE [m]	ϵ_{t_x}	0.0067	ϵ_t	0.0073
	ϵ_{t_y}	0.0105		
	ϵ_{t_z}	0.0022		
VELOCITÀ TRASLAZIONALE [m/s]	ϵ_{v_x}	4.6994E-05	ϵ_v	8.3938E-05
	ϵ_{v_y}	1.3217E-04		
	ϵ_{v_z}	3.8197E-05		
ROTAZIONE [deg]	ϵ_ϕ	0.0527	ϵ_α	0.0860
	ϵ_θ	0.0835		
	ϵ_ψ	0.1116		
VELOCITÀ ROTAZIONALE [deg/s]	ϵ_{ω_x}	0.0273	ϵ_ω	0.0218
	ϵ_{ω_y}	0.0240		
	ϵ_{ω_z}	0.0111		

Tabella 6.11: Errori rms nello scenario 2 del test 06.

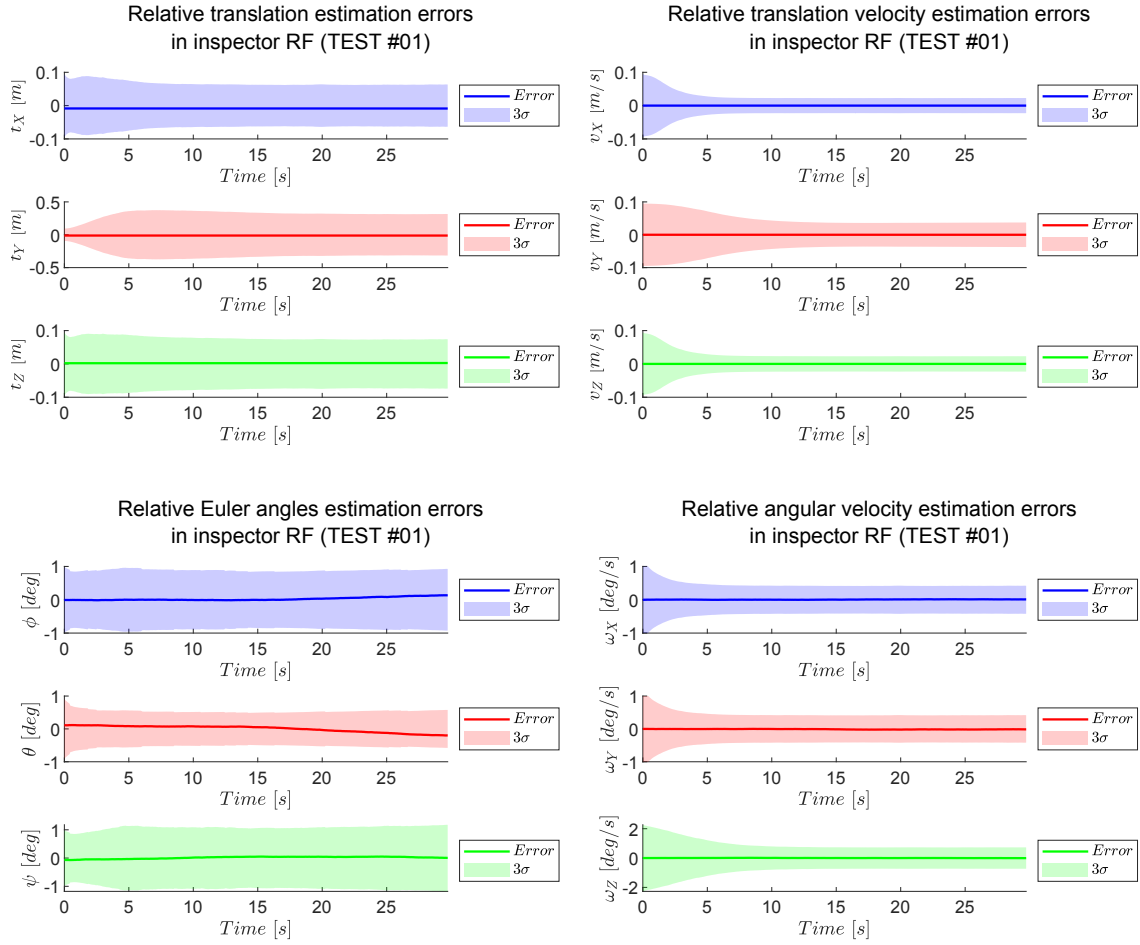


Figura 6.10: Errori nello scenario 3 del test 01.

TRASLAZIONE [m]	ϵ_{t_x}	0.0086	ϵ_t	0.0086
	ϵ_{t_y}	0.0119		
	ϵ_{t_z}	0.0023		
VELOCITÀ TRASLAZIONALE [m/s]	ϵ_{v_x}	6.6293E-06	ϵ_v	1.6430E-05
	ϵ_{v_y}	1.4318E-05		
	ϵ_{v_z}	2.3683E-05		
ROTAZIONE [deg]	ϵ_ϕ	0.0534	ϵ_α	0.0681
	ϵ_θ	0.0975		
	ϵ_ψ	0.0394		
VELOCITÀ ROTAZIONALE [deg/s]	ϵ_{ω_x}	0.0079	ϵ_ω	0.0091
	ϵ_{ω_y}	0.0123		
	ϵ_{ω_z}	0.0059		

Tabella 6.12: Errori rms nello scenario 3 del test 01.

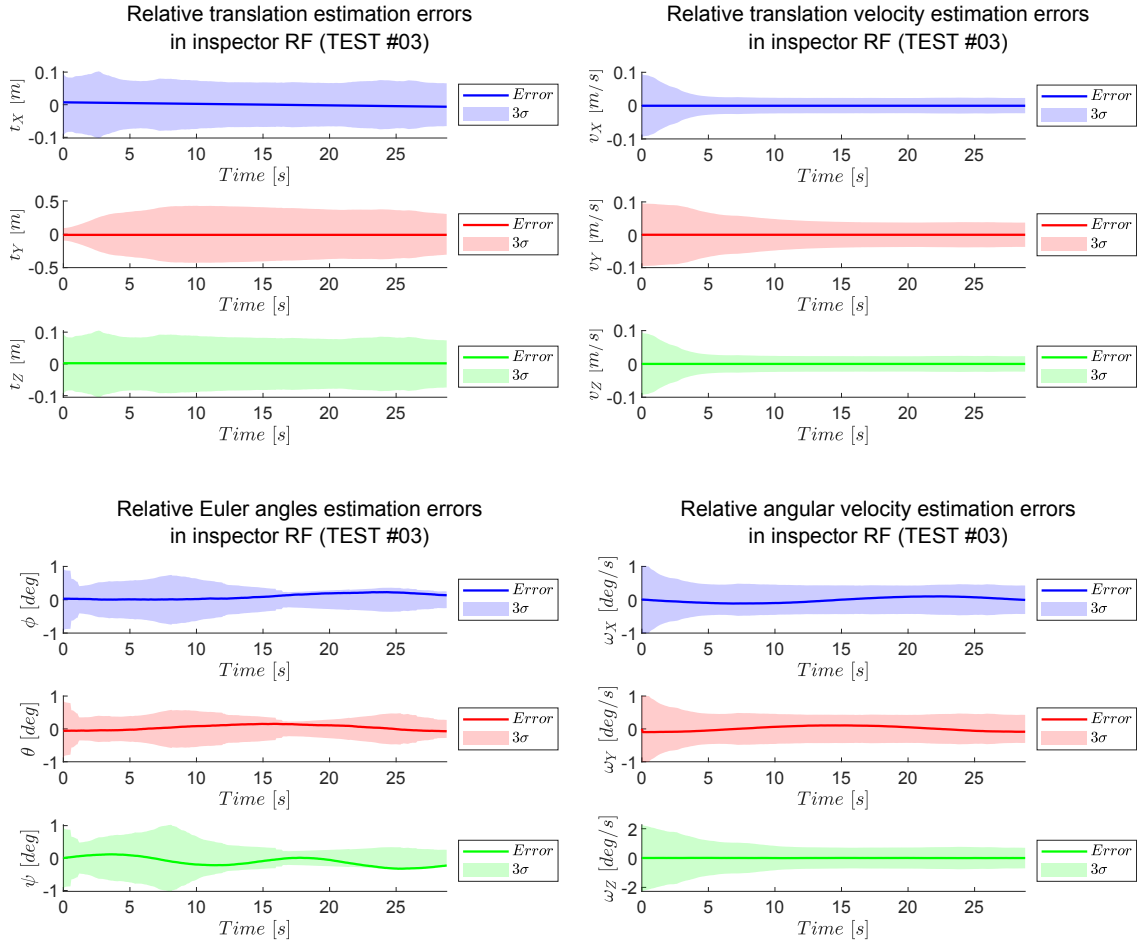


Figura 6.11: Errori nello scenario 3 del test 03.

TRASLAZIONE [m]	ϵ_{t_x}	0.0039	ϵ_t	0.0055
	ϵ_{t_y}	0.0084		
	ϵ_{t_z}	0.0021		
VELOCITÀ TRASLAZIONALE [m/s]	ϵ_{v_x}	4.6664E-04	ϵ_v	2.7177E-04
	ϵ_{v_y}	4.2571E-05		
	ϵ_{v_z}	4.4974E-05		
ROTAZIONE [deg]	ϵ_ϕ	0.0888	ϵ_α	0.0971
	ϵ_θ	0.0869		
	ϵ_ψ	0.1134		
VELOCITÀ ROTAZIONALE [deg/s]	ϵ_{ω_x}	0.0747	ϵ_ω	0.0599
	ϵ_{ω_y}	0.0719		
	ϵ_{ω_z}	0.0045		

Tabella 6.13: Errori rms nello scenario 3 del test 03.

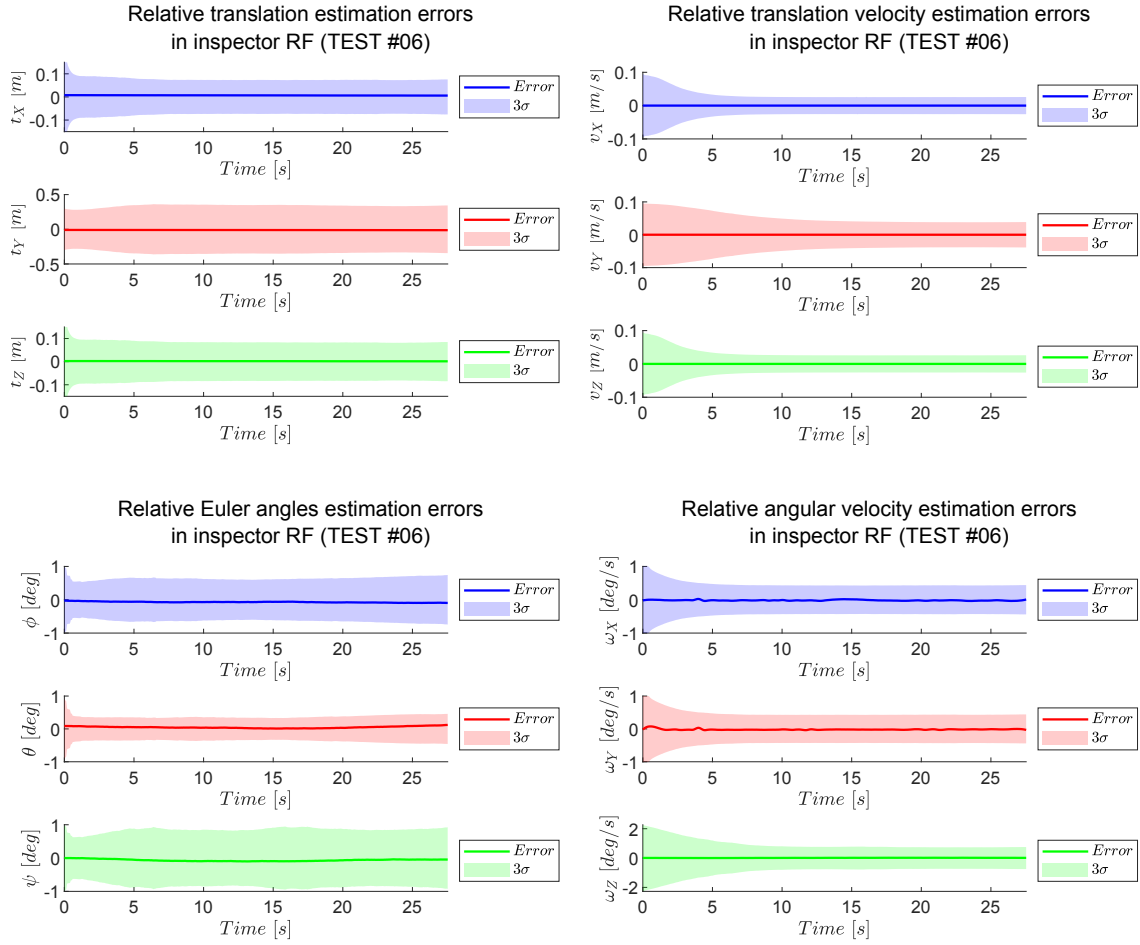


Figura 6.12: Errori nello scenario 3 del test 06.

TRASLAZIONE [m]	ϵ_{t_x}	0.0068	ϵ_t	0.0076
	ϵ_{t_y}	0.0110		
	ϵ_{t_z}	0.0020		
VELOCITÀ TRASLAZIONALE [m/s]	ϵ_{v_x}	2.3021E-05	ϵ_v	5.8904E-05
	ϵ_{v_y}	9.9182E-05		
	ϵ_{v_z}	6.4936E-06		
ROTAZIONE [deg]	ϵ_ϕ	0.0719	ϵ_α	0.0663
	ϵ_θ	0.0562		
	ϵ_ψ	0.0697		
VELOCITÀ ROTAZIONALE [deg/s]	ϵ_{ω_x}	0.0208	ϵ_ω	0.0200
	ϵ_{ω_y}	0.0243		
	ϵ_{ω_z}	0.0133		

Tabella 6.14: Errori rms nello scenario 3 del test 06.

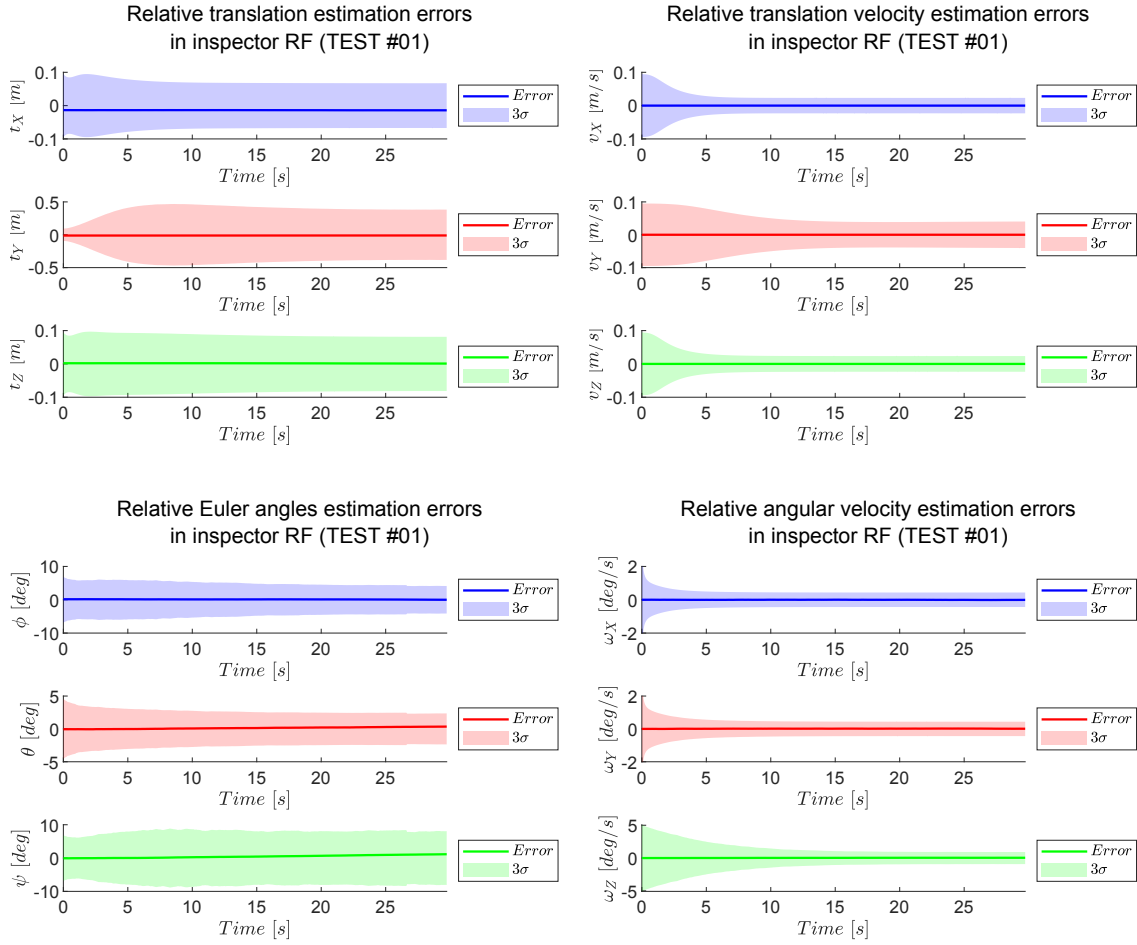


Figura 6.13: Errori nello scenario 4 del test 01.

TRASLAZIONE [m]	ϵ_{t_x}	0.0138	ϵ_t	0.0103
	ϵ_{t_y}	0.0113		
	ϵ_{t_z}	0.0018		
VELOCITÀ TRASLAZIONALE [m/s]	ϵ_{v_x}	2.3001E-05	ϵ_v	3.3955E-05
	ϵ_{v_y}	3.5403E-05		
	ϵ_{v_z}	4.0944E-05		
ROTAZIONE [deg]	ϵ_ϕ	0.1087	ϵ_α	0.3701
	ϵ_θ	0.1847		
	ϵ_ψ	0.6042		
VELOCITÀ ROTAZIONALE [deg/s]	ϵ_{ω_x}	0.0058	ϵ_ω	0.0203
	ϵ_{ω_y}	0.0144		
	ϵ_{ω_z}	0.0315		

Tabella 6.15: Errori rms nello scenario 4 del test 01.

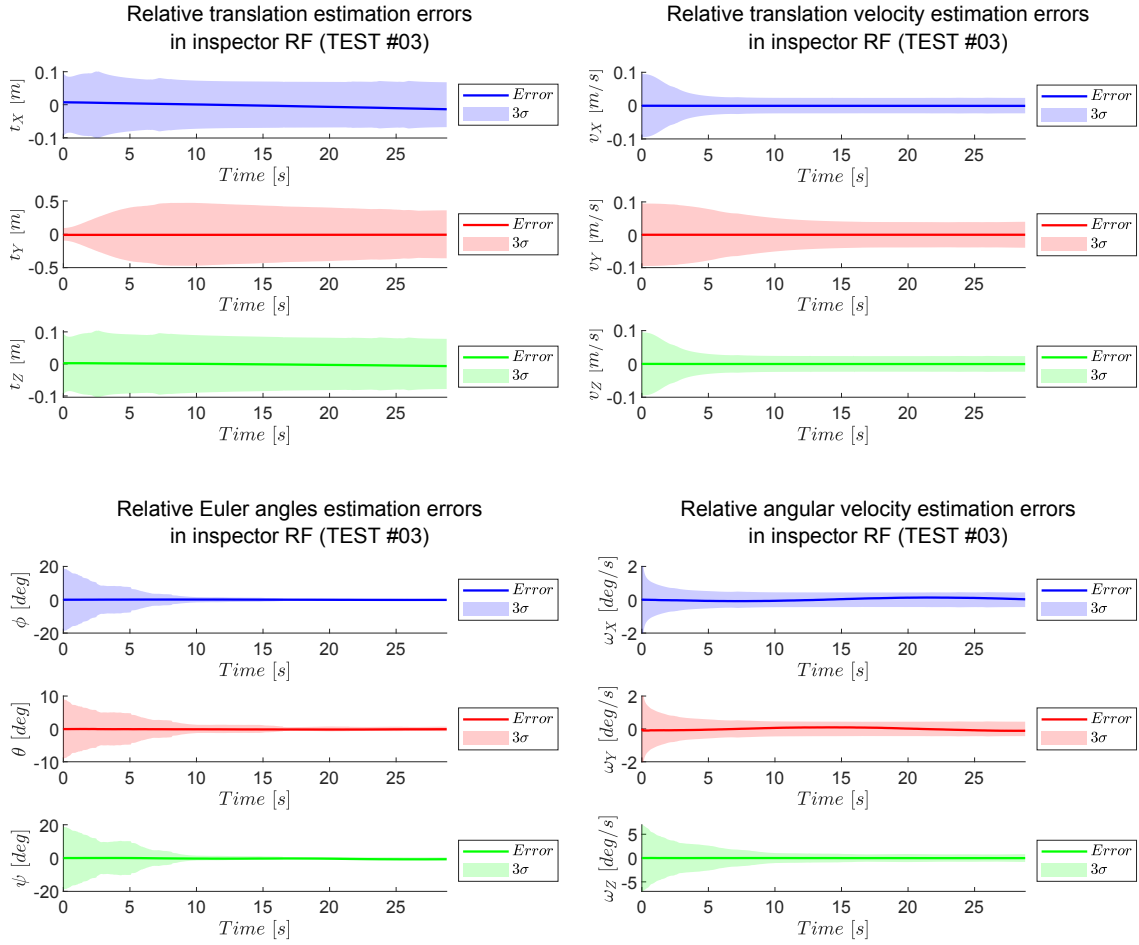


Figura 6.14: Errori nello scenario 4 del test 03.

TRASLAZIONE [m]	ϵ_{t_x}	0.0066	ϵ_t	0.0061
	ϵ_{t_y}	0.0077		
	ϵ_{t_z}	0.0030		
VELOCITÀ TRASLAZIONALE [m/s]	ϵ_{v_x}	6.9112E-04	ϵ_v	4.3903E-04
	ϵ_{v_y}	3.0829E-05		
	ϵ_{v_z}	3.1568E-04		
ROTAZIONE [deg]	ϵ_ϕ	0.3935	ϵ_α	0.6225
	ϵ_θ	0.2369		
	ϵ_ψ	0.9754		
VELOCITÀ ROTAZIONALE [deg/s]	ϵ_{ω_x}	0.0792	ϵ_ω	0.0628
	ϵ_{ω_y}	0.0715		
	ϵ_{ω_z}	0.0215		

Tabella 6.16: Errori rms nello scenario 4 del test 03.

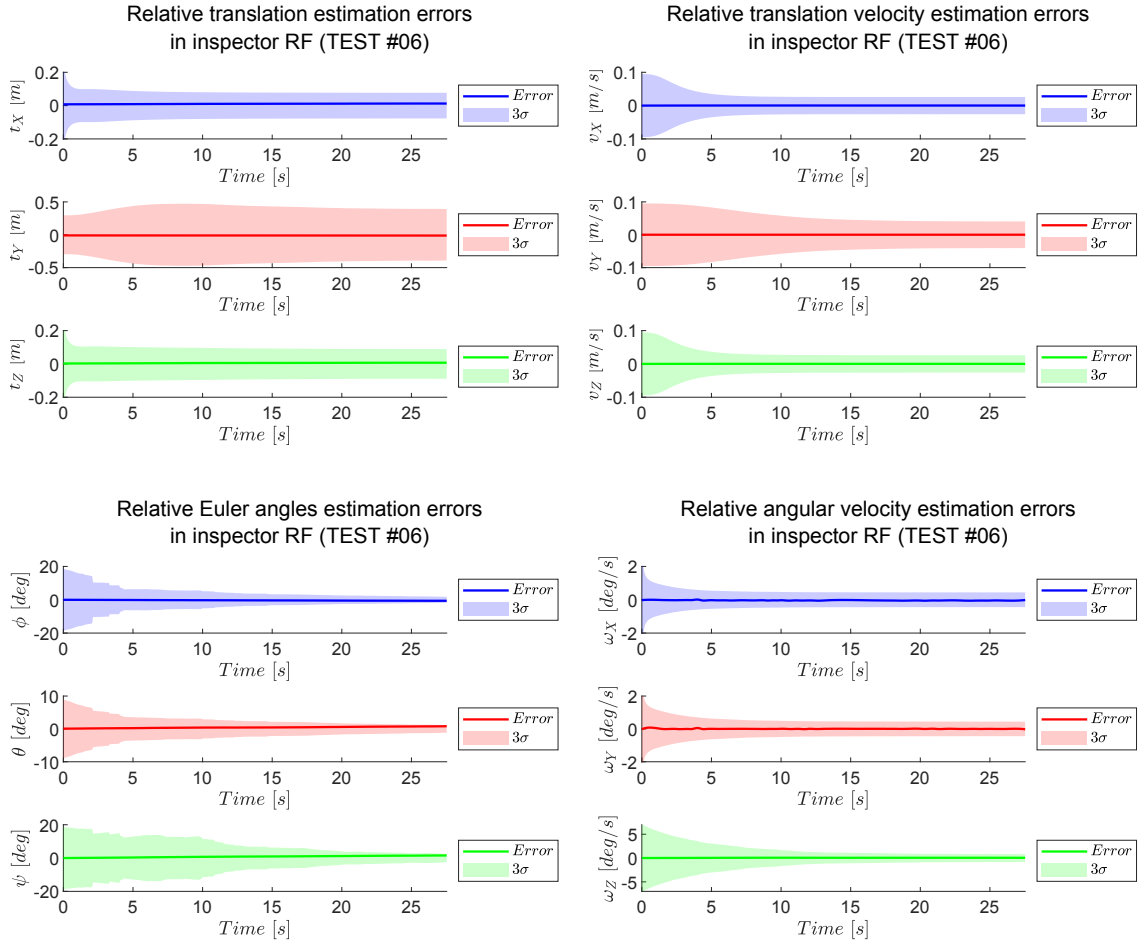


Figura 6.15: Errori nello scenario 4 del test 06.

TRASLAZIONE [m]	ϵ_{t_x}	0.0107	ϵ_t	0.0094
	ϵ_{t_y}	0.0110		
	ϵ_{t_z}	0.0056		
VELOCITÀ TRASLAZIONALE [m/s]	ϵ_{v_x}	2.6006E-04	ϵ_v	2.0775E-04
	ϵ_{v_y}	1.0550E-04		
	ϵ_{v_z}	2.2521E-04		
ROTAZIONE [deg]	ϵ_ϕ	0.0815	ϵ_α	0.5063
	ϵ_θ	0.4934		
	ϵ_ψ	0.7203		
VELOCITÀ ROTAZIONALE [deg/s]	ϵ_{ω_x}	0.0389	ϵ_ω	0.0393
	ϵ_{ω_y}	0.0150		
	ϵ_{ω_z}	0.0538		

Tabella 6.17: Errori rms nello scenario 4 del test 06.

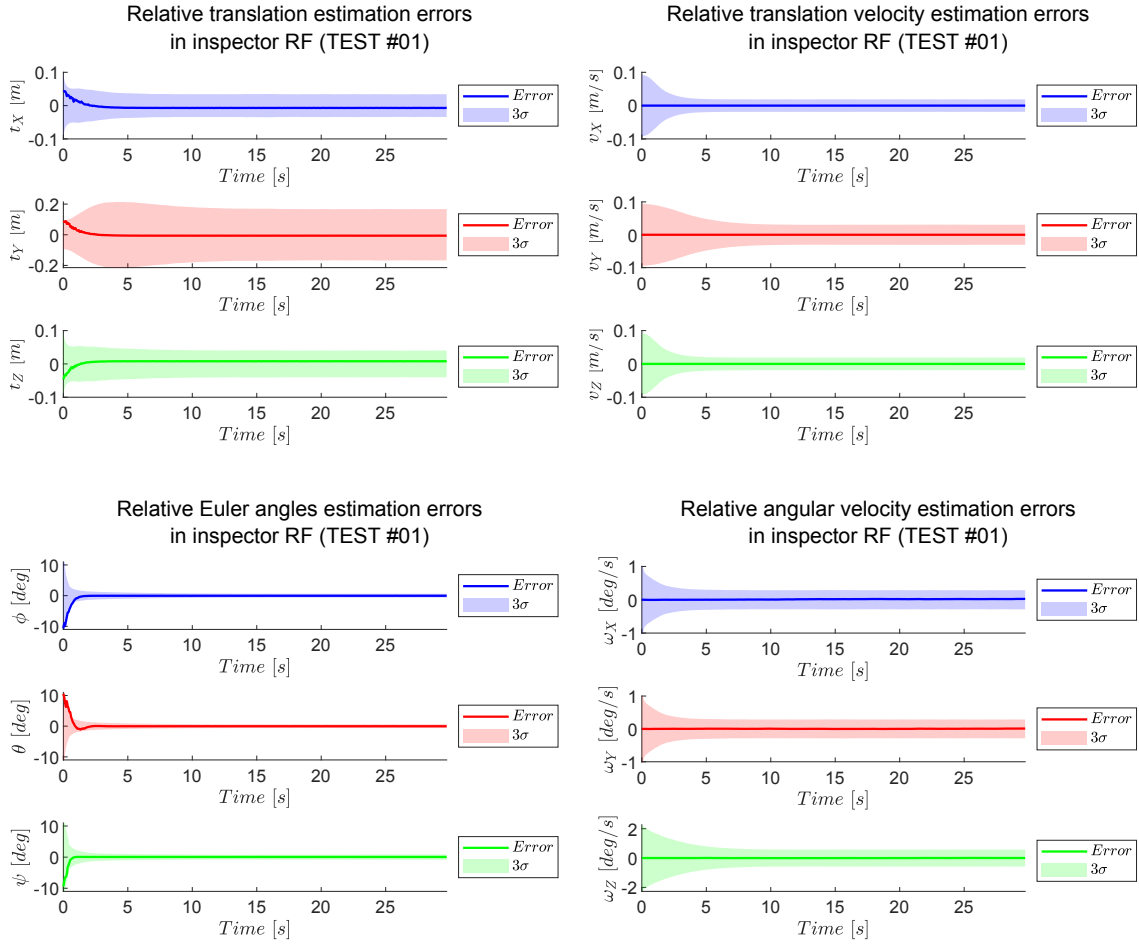


Figura 6.16: Errori nello scenario 5 del test 01.

TRASLAZIONE [m]	ϵ_{t_x} ϵ_{t_y} ϵ_{t_z}	0.0120 0.0130 0.0030		ϵ_t	0.0104
VELOCITÀ TRASLAZIONALE [m/s]	ϵ_{v_x} ϵ_{v_y} ϵ_{v_z}	1.1349E-05 1.2189E-05 1.3576E-05		ϵ_v	1.2405E-05
ROTAZIONE [m]	ϵ_ϕ ϵ_θ ϵ_ψ	0.0503 0.1147 0.0602		ϵ_α	0.0801
VELOCITÀ ROTAZIONALE [m/s]	ϵ_{ω_x} ϵ_{ω_y} ϵ_{ω_z}	0.0087 0.0042 0.0030		ϵ_ω	0.0058
TEMPO CONVERGENZA [s]	~ 4				

Tabella 6.18: Errori rms nello scenario 5 del test 01.

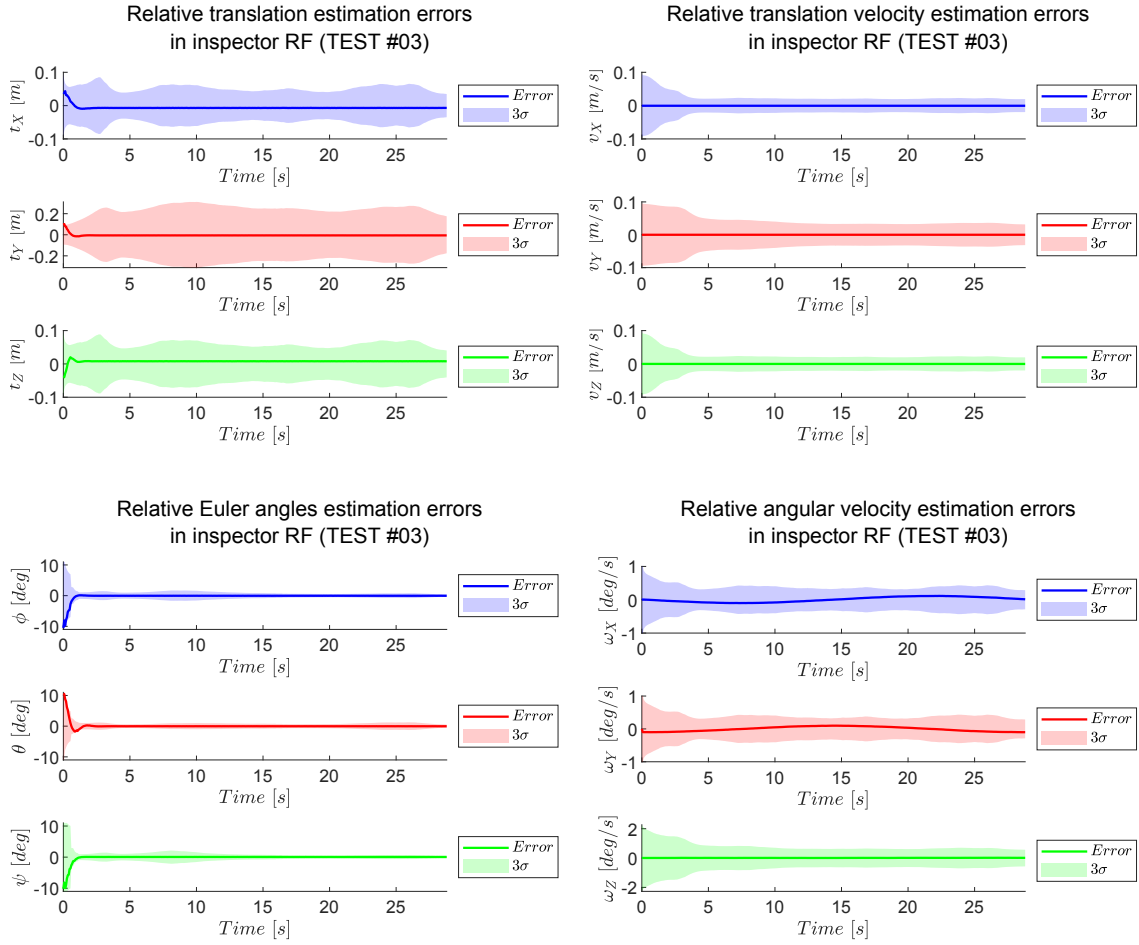


Figura 6.17: Errori nello scenario 5 del test 03.

TRASLAZIONE [m]	ϵ_{t_x}	0.0051		ϵ_t	0.0071
VELOCITÀ TRASLAZIONALE [m/s]	ϵ_{t_y}	0.0090			
	ϵ_{t_z}	0.0067			
	ϵ_{v_x}	4.6932E-04	ϵ_v	2.7406E-04	
ϵ_{v_y}	1.3176E-05				
ϵ_{v_z}	6.9917E-05				
ROTAZIONE [m]	ϵ_ϕ	0.0712		ϵ_α	0.1118
	ϵ_θ	0.1145			
	ϵ_ψ	0.1390			
VELOCITÀ ROTAZIONALE [m/s]	ϵ_{ω_x}	0.0744		ϵ_ω	0.0583
	ϵ_{ω_y}	0.0672			
	ϵ_{ω_z}	0.0117			
TEMPO CONVERGENZA [s]	~ 2.5				

Tabella 6.19: Errori rms nello scenario 5 del test 03.

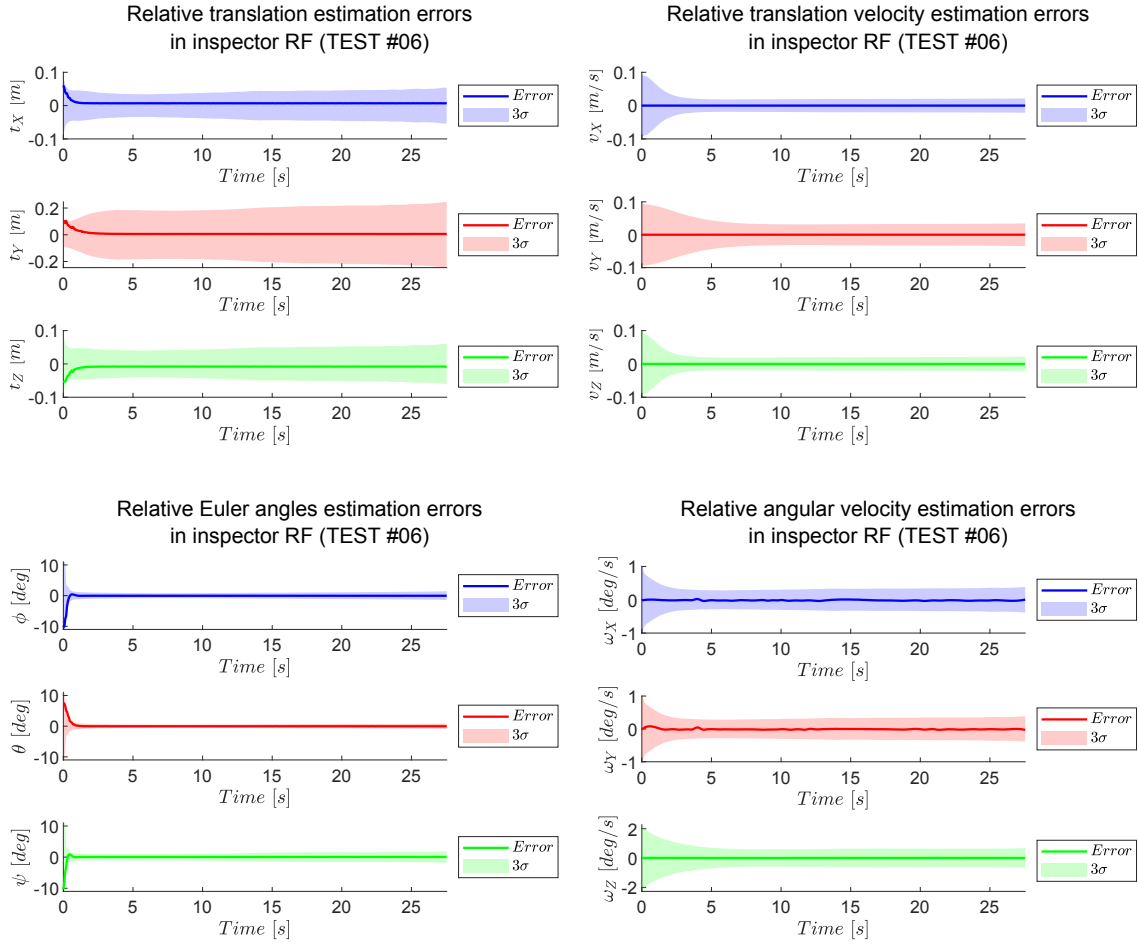


Figura 6.18: Errori nello scenario 5 del test 06.

TRASLAZIONE [m]	ϵ_{t_x}	0.0071		ϵ_t	0.0085
VELOCITÀ TRASLAZIONALE [m/s]	ϵ_{t_y}	0.0121			
	ϵ_{t_z}	0.0045			
ROTAZIONE [m]	ϵ_{v_x}	4.5237E-05		ϵ_v	7.7441E-05
	ϵ_{v_y}	1.2377E-04			
	ϵ_{v_z}	2.5023E-05			
VELOCITÀ ROTAZIONALE [m/s]	ϵ_{ϕ}	0.0524		ϵ_{α}	0.0691
	ϵ_{θ}	0.0873			
	ϵ_{ψ}	0.0630			
TEMPO CONVERGENZA [s]	ϵ_{ω_x}	0.0302		ϵ_{ω}	0.0272
	ϵ_{ω_y}	0.0348			
	ϵ_{ω_z}	0.0101			
			~ 2		

Tabella 6.20: Errori rms nello scenario 5 del test 06.

6.7 Risultati del moto traslazionale relativo

Per la tipologia di moto di pura rotazione imposta al target, la traslazione relativa non è molto affetta dallo scenario preso in considerazione. A ragione di ciò, le Fig. 6.19 e Fig. 6.20 sono emblematiche, in quanto è possibile osservare come gli errori rms per la stima della posizione relativa di tutti i test siano molto simili per tutti gli scenari. Si osserva inoltre che in generale l'errore lungo della variabile t_z è minore per tutti gli scenari ed i test analizzati.

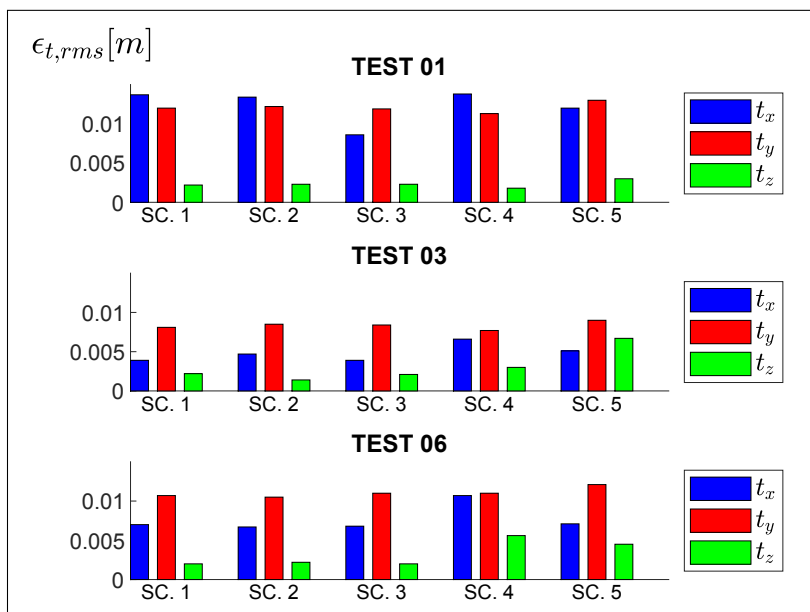


Figura 6.19: Errori rms della traslazione per tutti gli scenari dei test considerati.

Più interessante è analizzare il comportamento delle bande di incertezza del moto traslazionale relativo, sia in termini di posizione e sia in termini di velocità, visibili figure dalla Fig. 6.4 alla Fig. 6.18. In tutti i grafici, si può notare come la banda di incertezza più larga sia in corrispondenza dell'asse y_I . Il motivo per cui avviene ciò è da attribuire al fatto che, come si evince dalla Fig. 4.4, l'asse y_I è orientato lungo l'asse z_C , in verso opposto. Per come è stato definito il sistema di riferimento della stereo camera, l'asse z_C è lungo l'asse ottico, che è quello soggetto al maggior rumore. Infatti, piccoli errori nella identificazione delle coordinate in pixel delle features si possono tradurre in errori maggiori nella triangolazione, soprattutto nella direzione dell'asse ottico.

Altro aspetto interessante riguarda il fatto che gli errori nella determina-

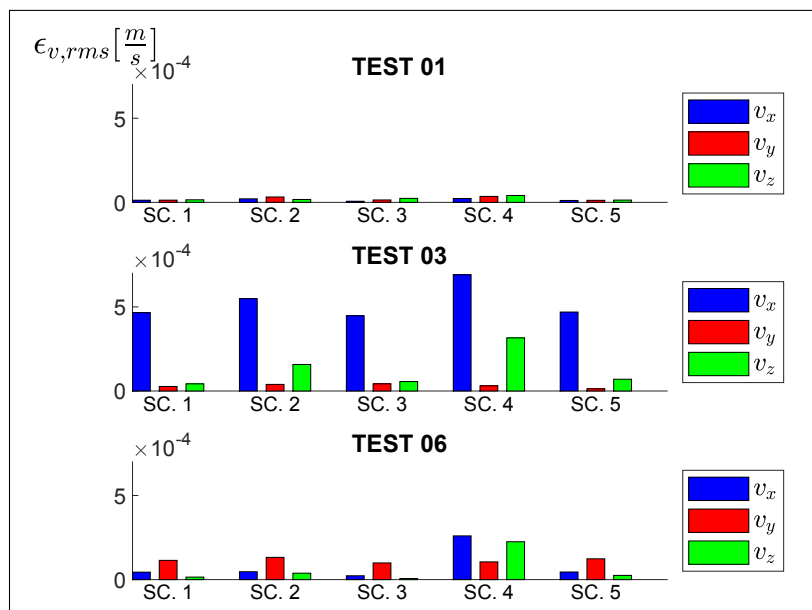


Figura 6.20: Errori rms della velocità traslazionale per tutti gli scenari dei test considerati.

zione della posizione relativa sono minori nel test 03. Questo può derivare dal fatto che dei tre, il test 03 è l'unico che compie una rotazione completa: il satellite viene visto in ogni sua parte, e quindi la determinazione della posizione relativa risulta più completa, con un processo simile a quello visto per l'inizializzazione della traslazione relativa per l'EKF, descritto nel paragrafo 5.1.1.

Per la velocità, invece, questo aspetto non vale. Infatti, se poter osservare il target in tutta la sua superficie può essere un vantaggio per la determinazione della posizione, la velocità è più suscettibile alla rotazione del target: più essa è alta, come nel test 03, più è fonte di errore. Si può comunque notare che l'errore assume valori bassi, con un range compreso tra 0.1 e 0.6 mm/s.

6.8 Risultati del moto rotazionale relativo

Differentemente dalla traslazione relativa, la rotazione relativa è molto più sensibile alla variazione degli scenari. Questa problematica è dovuta al fatto che il moto impresso al target è di pura rotazione. Siccome l'obiettivo è di mantenere l'errore di stima sempre all'interno della banda di incertezza, la differenza principale tra tutti gli scenari si riflette in una banda di incertezza più o meno larga a seconda

dei casi.

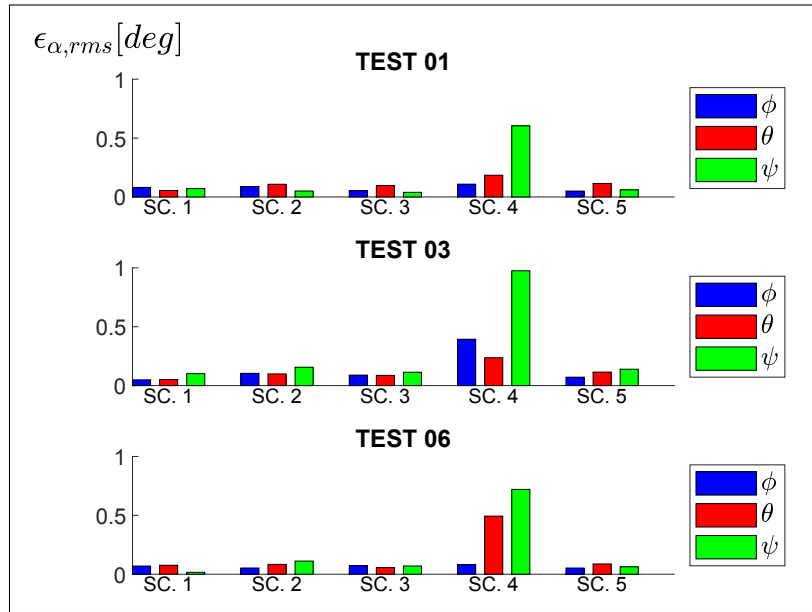


Figura 6.21: Errori rms della rotazione per tutti gli scenari dei test considerati.

Lo scenario con un errore rms maggiore, come visibile nella Fig. 6.21, è indubbiamente lo scenario 4, ovvero quello in cui le features sono concentrate in una limitata superficie del target, senza coprirlo in maniera uniforme. Anche lo scenario 2, ovvero quello con un diverso measurement update, presenta errori leggermente più grandi, ma non paragonabili a quelli dello scenario 4. Lo scenario 5, ovvero quello in cui le condizioni iniziali di assetto relativo sono più distanti dall'assetto reale, alla convergenza si comporta come gli altri scenari, in quanto le features utilizzate sono le stesse dello scenario 1, ovvero prendendo in considerazione tutte le features possibili. Si osserva che per lo scenario 5, essendo l'errore iniziale piuttosto alto, la matrice $\mathbf{P}_{0|0}$ per la rotazione assume valori maggiori, come affermato nel paragrafo 6.5. Tuttavia, essendo le features utilizzate complete, il filtro tende presto a convergere e quindi anche la banda di incertezza diventa più stretta.

Un altro aspetto importante riguarda il fatto che il test 03 presenta errori più grandi per la rotazione rispetto al test 01 ed al test 06. Questa è una conseguenza naturale del fatto che la velocità rotazionale è più alta, e quindi tra frame e frame molte features possono essere perse. In ogni scenario, ma soprattutto nello scenario 4, è presente un errore maggiore per il test 03. L'angolo maggiormente affetto da errore è l'angolo ψ , ovvero la direzione angolare di rotazione del target. Per questo

motivo, è normale che l'angolo ψ sia quello soggetto ad una minor precisione di stima.

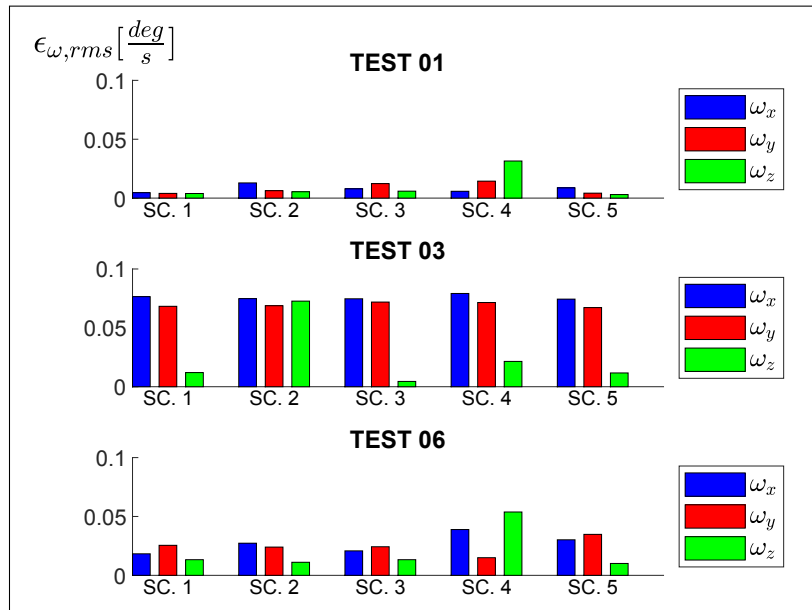


Figura 6.22: Errori rms della velocità rotazionale per tutti gli scenari dei test considerati.

Interessante notare che, come visibile nella Fig. 6.22, la velocità rotazionale sia meno sensibile alla variazione degli scenari rispetto alla rotazione, sebbene si noti un lieve aumento al solito nello scenario 4. Per il test 01 ciò è dovuto al fatto che il target è fermo, e conseguentemente l'errore si riduce. Similarmente, nel test 06, avendo imposto al target una rotazione lenta, l'errore è piuttosto contenuto seppur leggermente più alto rispetto al test 01. Questi errori sono molto più elevati nel test 03. Essendo la simulazione con una velocità rotazionale più elevata, è normale che anche gli errori siano più elevati.

Sempre per quanto riguarda la velocità rotazionale, si nota come nella maggior parte dei casi l'errore di stima più elevato non è in corrispondenza della variabile ω_z , cioè quella di rotazione nominale del target, bensì in corrispondenza di ω_x e ω_y . Ciò è dovuto al fatto che l'asse z del target non è perfettamente parallelo all'asse di rotazione stadio rotativo, ma leggermente inclinato. In tal senso, una componente di velocità rotazionale è anche presente sugli altri angoli seppur molto piccola. L'errore si genera dal momento che nelle condizioni iniziali tali velocità sono state poste nulle. Variando tali condizioni iniziali per avvicinarle a quelle reali, quest'errore diminuisce. In ogni caso, l'errore si mantiene in un range accettabile

inferiore a 0.02 deg/s per il test 01, 0.08 deg/s per il test 03 e 0.04 deg/s per il test 06.

Capitolo 7

Conclusioni

Nel corso di questa tesi è stato descritto il moto relativo tra satelliti e la tecnica per la stima del moto tramite un sistema di stereo visione. È stata spiegata la teoria che sta a monte con le equazioni necessarie da utilizzare per questo tipo di studio. Ora verranno fatte delle considerazioni finali su vantaggi e svantaggi nell'utilizzo di un sistema di stereo visione per la stima del moto relativo, sulla base dei risultati ottenuti dall'attività sperimentale.

La stereo camera è un sensore di dimensioni relativamente ridotte, leggero e che necessita di un'alimentazione elettrica limitata (Tab. 4.2). Queste peculiari caratteristiche la rendono idonea all'utilizzo nello spazio, in quanto poter risparmiare su elementi quali massa e potenza elettrica è un aspetto che ogni veicolo spaziale deve tenere conto. D'altro canto, bisogna considerare che l'elaborazione delle immagini è molto onerosa dal punto di vista computazionale, quindi il computer di bordo deve essere in grado di far fronte anche a questo aspetto. Tutte le simulazioni eseguite per questo lavoro sono state fatte in post-processing. Per un'applicazione real-time deve essere utilizzato un computer con buone prestazioni di calcolo e gli algoritmi devono essere ottimizzati.

Altra problematica da considerare riguarda la qualità delle immagini, soprattutto in termini di riflessioni. Nelle simulazioni, in funzione della posa del target, si sono verificate condizioni tali per cui le alte riflessioni sul pannello solare del target e la superficie metallica del target hanno reso difficoltosa la ricerca di features. Un

satellite in orbita bassa, a meno di particolari condizioni orbitali come per esempio il sun-sincronismo, è affetto da condizioni di luminosità molto variabili. Se le condizioni di luminosità sono tali da generare alte riflessioni, la ricerca delle features risulta essere più complicata, in quanto esse potrebbero risultare non ben disposte sul target o peggio l'algoritmo non riuscirebbe a trovarle. Inoltre, la presenza della Terra o delle stelle nel campo di vista delle camere può essere un problema non di poco conto. Quest'ultimo può essere però mitigato tramite l'imposizione di una threshold sulla distanza delle features trovate.

I principali risultati di questo lavoro di tesi possono essere riassunti come segue.

- *Scelta delle features:*

Se le features vengono scelte in maniera ottimale, l'EKF è in grado di fornire una stima del moto relativo con buona accuratezza anche a fronte di un numero più ridotto di features. Si è visto che avere features concentrate in una zona specifica del target porti ad avere una stima peggiore rispetto ad avere features distribuite in maniera uniforme. In questo contesto, lo scenario 4 è quello da evitare, mentre gli altri scenari portano ad una stima migliore, come riportato nei paragrafi 6.7 e 6.8.

- *Ritardo nell'aggiornamento dello stato:*

Avere un measurement update con frequenza alta può rappresentare un problema per quanto concerne le applicazioni in real-time. Il tempo per analizzare le immagini per ogni frame non è trascurabile, e se la frequenza è alta può portare a ritardi nell'aggiornamento dello stato in maniera non compatibile con le manovre che si stanno eseguendo. In questo contesto, lo scenario 2 è quello che si presta di più ad applicazioni in real-time, in quanto lo stato viene propagato con una frequenza di 15 Hz e tale da avere un processo di linearizzazione delle equazioni del moto che non porta ad eccessivi errori, mentre le misure sono aggiornate con una frequenza di 1 Hz tale da non appesantire troppo l'algoritmo di stima.

Avere tante features porta ad una stima migliore però può avere un effetto ne-

gativo sulla velocità dell'algoritmo. Si è visto che, in generale, lo stesso livello di accuratezza può essere ottenuto con meno features ma ben distribuite. In questo contesto, lo scenario 3 è l'ideale.

- *Inizializzazione dello stato:*

Il processo di inizializzazione dell'EKF è cruciale per garantire la convergenza del filtro. Tuttavia, non si può speculare sull'estrema precisione, ma bisogna tenere conto che possa verificarsi un errore che deve essere comunque contenuto. Pertanto, in un contesto reale lo scenario 5 è quello che lo rappresenta in maniera più corretta.

Visti tutti questi aspetti, lo scenario reale applicativo da prendere in considerazione potrebbe essere una combinazione tra gli scenari 2, 3 e 5. In questa maniera si minimizza il problema del ritardo dovuto ad un'alta frequenza di acquisizione delle immagini e da un numero eccessivo di features, le features sono distribuite uniformemente sul target in cui lo stato iniziale deve essere stimato tramite degli algoritmi per mantenere l'errore nell'ordine dei 10 cm per la posizione e dei 10 deg per l'assetto.

Questo lavoro di tesi può essere utilizzato per migliorare degli aspetti già presenti allo stato dell'arte e a sua volta sviluppato ulteriormente per analizzare altri elementi importanti. Ad esempio, nel paragrafo 3.5, quando è stata discussa la selezione delle features, è stato spiegato come nel caso stazionario sia possibile eseguire un tracciamento della stessa features durante tutto il tempo di acquisizione delle immagini, ma non nel caso rotazionale. Un aspetto interessante potrebbe essere creare un algoritmo che riesca a riconoscere una determinata features tramite un processo di ispezione, come per esempio il vertice di un pannello solare, un'antenna o un ugello, in maniera tale che il sistema di visione riesca a riconoscere tale componente una volta che entra nel campo di vista della stereo camera. In tal senso, un approccio che potrebbe essere interessante è integrare lo studio eseguito con una rete neurale convoluzionale (CNN) per la stima del moto relativo attraverso il tracciamento di alcune features particolari. Degli studi sulle CNN sono stati fatti in [33] e [34]. La CNN è fondamentale nel campo del *deep learning* [35] e pertanto

è molto promettente per lo scopo dello studio delle features e quindi del moto del target.

Un altro aspetto riguarda la tipologia di filtro di Kalman utilizzata. L'EKF risulta necessario in quanto le equazioni del moto sono non lineari. D'altro canto, la linearizzazione del modello di processo è fonte di errore proporzionalmente alla frequenza adottata per la discretizzazione dello stato. Se la discretizzazione venisse fatta con un intervallo troppo piccolo, quindi frequenza di propagazione troppo alta, il costo computazionale aumenterebbe. Viceversa, se avesse una frequenza troppo bassa, la stima peggiorerebbe. Per questo motivo, un trade-off sul valore di frequenza di propagazione deve essere fatto. Tuttavia, esistono altre tipologie di filtro di Kalman la cui precisione è migliore senza intaccare eccessivamente il costo computazionale. Uno di questi è l'UKF (*Unscented Kalman Filter*), in cui una buona analisi per uno scenario cooperativo è stata fatta in [36], il cui sviluppo potrebbe essere applicarlo allo scenario non cooperativo adottato in questo lavoro di tesi.

Concludendo, visti tutti questi aspetti, stimare il moto relativo tra un satellite chaser ed un target non cooperativo tramite un sistema di stereo visione è una soluzione molto promettente che tuttavia richiede particolare attenzione a certe problematiche analizzate in questa tesi e legate sia all'analisi delle immagini per l'estrazione di features sia all'algoritmo che utilizza queste ultime informazioni assieme ad un modello della dinamica relativa tra i due satelliti. Gli sviluppi ed i miglioramenti futuri porteranno ad utilizzare questa tecnologia sempre di più nelle missioni spaziali.

Bibliografia

- [1] <https://boeing.mediaroom.com/2014-8-26-Boeing-Continues-Progress-on-Improved-Space-Station-Docking-System>.
- [2] Pierdomenico Fracchiolla. «Analysis and Validation of a Vision-Based Pose Initialization Algorithm for Non-Cooperative Spacecrafts». In: 2019.
- [3] Brent E. Tweddle. «Computer vision based navigation for spacecraft proximity operations». In: 2010.
- [4] Quirin Mühlbauer et al. «ROBOTICS SPACE SYSTEMS AND SUBSYSTEMS FOR ADVANCED FUTURE PROGRAMMES». In: 2012.
- [5] Han Cai et al. «A space-based stereovision system for relative state estimation of space debris». In: set. 2016. DOI: 10.2514/6.2016-5515.
- [6] F. Rosso et al. «Stereo vision system for capture and removal of space debris». In: gen. 2013, pp. 201–207.
- [7] I. Newton. *Philosophiæ Naturalis Principia Mathematica*. 1687.
- [8] W. Clohessy e R. Wiltshire. «Terminal guidance system for satellite rendezvous». In: *J. Astronaut. Sci.* 27(9) (1960), pp. 653–678.
- [9] R. Wertz. *Spacecraft Attitude Determination and Control*. Kluwer Academic Publishers, 1988.
- [10] M. Lesser. «3 - Charge coupled device (CCD) image sensors». In: *High Performance Silicon Imaging*. A cura di Daniel Durini. Woodhead Publishing, 2014, pp. 78–97. ISBN: 978-0-85709-598-5. DOI: <https://doi.org/10.1533/9780857097521.1.78>. URL: <https://www.sciencedirect.com/science/article/pii/B9780857095985500038>.
- [11] A. Venkatanarayanan e E. Spain. «13.03 - Review of Recent Developments in Sensing Materials». In: *Comprehensive Materials Processing*. A cura di Saleem Hashmi et al. Oxford: Elsevier, 2014, pp. 47–101. ISBN: 978-0-08-096533-8. DOI: <https://doi.org/10.1016/B978-0-08-096532-1.01303-0>. URL: <https://www.sciencedirect.com/science/article/pii/B9780080965321013030>.
- [12] Peter Sturm. «Pinhole Camera Model». In: *Computer Vision: A Reference Guide*. A cura di Katsushi Ikeuchi. Boston, MA: Springer US, 2014, pp. 610–613. ISBN: 978-0-387-31439-6. DOI: 10.1007/978-0-387-31439-6_472. URL: https://doi.org/10.1007/978-0-387-31439-6_472.
- [13] https://docs.fab-image.com/current/studio/filters/CameraCalibration/CalibrateCamera_Telecentric.html.

- [14] Z. Zhang. «A flexible new technique for camera calibration». In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22.11 (2000), pp. 1330–1334. DOI: 10.1109/34.888718.
- [15] C. Harris e M. Stephens. «A Combined Corner and Edge Detector». In: *The 4th Alvey Vision Conference, Manchester* (1988), pp. 147–151. DOI: 10.1023/B:VISI.0000029664.99615.94.
- [16] Herbert Bay et al. «Speeded-Up Robust Features (SURF)». In: *Computer Vision and Image Understanding* 110.3 (2008), pp. 346–359. DOI: <https://doi.org/10.1016/j.cviu.2007.09.014>. URL: <https://www.sciencedirect.com/science/article/pii/S1077314207001555>.
- [17] Marius Muja e David Lowe. «Fast Matching of Binary Features». In: mag. 2012, pp. 404–410. ISBN: 978-1-4673-1271-4. DOI: 10.1109/CRV.2012.60.
- [18] Richard Hartley e Andrew Zisserman. «Multiple view geometry in computer vision». In: *Robotica* 23 (2005), pp. 271–271. DOI: 10.1017/S0263574705211621.
- [19] Andrea Valmorbida et al. «Design of a ground-based facility to reproduce satellite relative motions». In: *2017 IEEE International Workshop on Metrology for AeroSpace (MetroAeroSpace)*. 2017, pp. 468–473. DOI: 10.1109/MetroAeroSpace.2017.7999619.
- [20] Andrea Valmorbida, Mattia Mazzucato e Marco Pertile. «Calibration procedures of a vision-based system for relative motion estimation between satellites flying in proximity». In: *Measurement* 151 (2020), p. 107161. ISSN: 0263-2241. DOI: <https://doi.org/10.1016/j.measurement.2019.107161>. URL: <https://www.sciencedirect.com/science/article/pii/S0263224119310279>.
- [21] <https://www.stereolabs.com/assets/datasheets/zed2-camera-datasheet.pdf>.
- [22] <https://www.standa.lt/files/usb/8SMC1-USBhF>.
- [23] <https://www.merrysprout.com/posts/camera/pnp/>.
- [24] Xiao-Shan Gao et al. «Complete Solution Classification for the Perspective-Three-Point Problem». In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 25 (set. 2003), pp. 930–943. DOI: 10.1109/TPAMI.2003.1217599.
- [25] Mohamed Rizon et al. «Object Detection using Circular Hough Transform». In: *American Journal of Applied Sciences* 2 (dic. 2005). DOI: 10.3844/ajassp.2005.1606.1609.
- [26] R.E. Kalman. «A New Approach to Linear Filtering and Prediction Problems». In: *Journal of Basic Engineering* 82 (1960), pp. 35–45. DOI: 10.2514/8.8704.
- [27] R. C. Gonzalez, R. E. Woods e S. L. Eddins. *Digital Image Processing Using MATLAB*. Pearson Prentice Hall, 2004.

- [28] Feng Yu et al. «Stereo-Vision-Based Relative Pose Estimation for the Rendezvous and Docking of Noncooperative Satellites». In: *Mathematical Problems in Engineering* 2014 (nov. 2014), pp. 1–12. DOI: 10.1155/2014/461283.
- [29] Han Yu et al. «Relative dynamics estimation of non-cooperative spacecraft with unknown orbit elements and inertial tensor». In: *Chinese Journal of Aeronautics* 29.2 (2016), pp. 479–491. ISSN: 1000-9361. DOI: <https://doi.org/10.1016/j.cja.2016.01.013>. URL: <https://www.sciencedirect.com/science/article/pii/S1000936116000455>.
- [30] Feng Yu et al. «Stereo-Vision-Based Relative Pose Estimation for the Rendezvous and Docking of Noncooperative Satellites». In: *Mathematical Problems in Engineering* 2014 (nov. 2014), pp. 1–12. DOI: 10.1155/2014/461283.
- [31] Dongming Ge et al. «Motion and inertial parameter estimation of non-cooperative target on orbit using stereo vision». In: *Advances in Space Research* 66.6 (2020), pp. 1475–1484. ISSN: 0273-1177. DOI: <https://doi.org/10.1016/j.asr.2020.05.029>. URL: <https://www.sciencedirect.com/science/article/pii/S0273117720303628>.
- [32] Mattia Mazzucato et al. «Stereoscopic Vision-Based Relative Navigation for Spacecraft Proximity Operations». In: *2018 5th IEEE International Workshop on Metrology for AeroSpace (MetroAeroSpace)*. 2018, pp. 369–374. DOI: 10.1109/MetroAeroSpace.2018.8453524.
- [33] Shaoqing Ren et al. «Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks». In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39.6 (2017), pp. 1137–1149. DOI: 10.1109/TPAMI.2016.2577031.
- [34] Yu Xiang et al. «PoseCNN: A Convolutional Neural Network for 6D Object Pose Estimation in Cluttered Scenes». In: (nov. 2017).
- [35] Pedro F. Proença e Yang Gao. «Deep Learning for Spacecraft Pose Estimation from Photorealistic Rendering». In: *2020 IEEE International Conference on Robotics and Automation (ICRA)*. 2020, pp. 6007–6013. DOI: 10.1109/ICRA40945.2020.9197244.
- [36] Giovanni Rizzolo. «Unscented Kalman Filters for Vision-Based Relative Dynamics Estimation between Cooperating Spacecraft». In: 2019.