



UNIVERSITÀ POLITECNICA DELLE  
MARCHE

CORSO DI LAUREA IN INGEGNERIA INFORMATICA E  
DELL'AUTOMAZIONE

**Algoritmi per l'Estrazione della Frequenza  
Respiratoria da Immagini di Profondità**

*Algorithms for Extracting Respiratory  
Rate from Depth Images*

Relatore:

**Prof. Gambi Ennio**

Candidato:

**Chiappa Christian**

Correlatore:

**Prof.ssa Senigagliesi Linda**

---

ANNO ACCADEMICO 2023/2024

*A chi mi ha ispirato nei momenti di incertezza.  
E a chi mi accompagnerà nei sogni ancora da realizzare.*

# Abstract

Questo lavoro di tesi è incentrato sull'estrazione della frequenza respiratoria di un individuo, sia esso statico che in movimento, utilizzando una telecamera Intel RealSense D455. Per verificare la validità degli algoritmi sviluppati, abbiamo utilizzato la cintura PolarH10 come sensore di riferimento, sincronizzando i dati acquisiti con quelli della telecamera. Le acquisizioni, condotte in un ambiente chiuso e ben illuminato, hanno coinvolto un totale di cinque soggetti.

Nel caso statico, i soggetti sono stati posizionati su una sedia a varie distanze e in piedi con il busto ruotato. Sono stati sviluppati quattro algoritmi che considerano diverse zone del torace per il calcolo della frequenza respiratoria. Il miglior algoritmo ha riscontrato un MAE (Mean Absolute Error) di 0.0798 RPM (respiri per minuto) per soggetti seduti e 0.66 RPM per quelli ruotati, indicando un'ottima precisione di misurazione.

Nel caso dinamico, sono stati condotti tre esperimenti: movimento frontalmente alla telecamera, rotazione del torso e movimenti casuali. Abbiamo implementato ulteriori algoritmi per cercare di annullare l'effetto del movimento corporeo, sottraendo i valori ottenuti dagli algoritmi statici con quelli di profondità provenienti da zone non influenzate dal movimento respiratorio, come collo, punto centrale delle spalle e addome. Sia per il movimento frontale che per la rotazione del torso l'algoritmo che ha performato meglio ha ottenuto un errore di 0.06 RPM nel primo caso e 2.568 RPM nel secondo mentre per i movimenti casuali l'algoritmo migliore ha ottenuto un MAE di 2.328 RPM evidenziando un notevole peggioramento rispetto ai casi statici.

# Indice

<b>1</b>	<b>Introduzione</b>	<b>7</b>
<b>2</b>	<b>Materiali</b>	<b>9</b>
2.1	Sensori utilizzati . . . . .	9
2.1.1	Intel RealSense D455 . . . . .	9
2.1.2	Polar H10 . . . . .	12
2.2	Acquisizioni . . . . .	14
2.2.1	Setup sperimentale . . . . .	14
<b>3</b>	<b>Metodi</b>	<b>16</b>
3.1	Algoritmo di acquisizione dalla depth camera . . . . .	16
3.2	Algoritmo di acquisizione dal Polar . . . . .	17
3.3	Sincronizzazione dei segnali . . . . .	18
3.4	Elaborazione dei dati attraverso i grafici . . . . .	18
3.4.1	Processamento telecamera RealSense . . . . .	18
3.4.2	Segnale ECG estratto dal PolarH10 . . . . .	22
3.4.3	Segnale respiratorio estratto dal PolarH10 . . . . .	24
3.4.4	Segnale asse Z estratto dal PolarH10 . . . . .	26
3.5	Elaborazione dei dati caso statico . . . . .	27
3.5.1	Segnale respiratorio estratto dalla telecamera . . . . .	28
3.6	Elaborazione dei dati caso dinamico . . . . .	29
3.6.1	Segnale respiratorio estratto dalla differenza con il punto centrale delle spalle . . . . .	29
3.6.2	Segnale respiratorio estratto dalla differenza con il collo . . . . .	31
3.6.3	Segnale respiratorio estratto dalla differenza con l'addome . . . . .	33
<b>4</b>	<b>Risultati e discussione</b>	<b>36</b>
4.1	Condizioni statiche . . . . .	36
4.1.1	Soggetti seduti . . . . .	36
4.1.2	Soggetti ruotati . . . . .	42
4.2	Condizioni dinamiche . . . . .	48
4.2.1	Movimento frontale . . . . .	49
4.2.2	Rotazione del torso . . . . .	53
4.2.3	Movimento casuale . . . . .	57

INDICE	4
--------	---

---

5 Sviluppi futuri	65
-------------------	----

6 Conclusione	66
---------------	----

# Elenco delle figure

2.1	Intel RealSense D455 [3]	9
2.2	Stereo Vision Technology	10
2.3	Polar H10 con rappresentazione degli assi	12
2.4	Posizione statica da seduto	14
2.5	Posizione statica leggermente girata	15
3.1	Frame processati con box e keypoints	19
3.2	Punto centrale tra le spalle	19
3.3	Punto sul collo	20
3.4	Punto all'interno del torso	20
3.5	Maschera della parte superiore del torso	21
3.6	Maschera della parte inferiore torso	21
3.7	Maschera del torso	22
3.8	Complesso QRS	23
3.9	Esempio di grafico ECG con annotati i picchi e BPM	24
3.10	Esempio di grafico del segnale respiratorio nel tempo	25
3.11	Esempio di grafico del segnale respiratorio nel dominio della frequenza	26
3.12	Esempio di grafico dell' accelerazione (asse Z) nel tempo	26
3.13	Esempio di grafico dell' accelerazione (asse Z) detrendata e filtrata	27
3.14	Esempio di grafico dell'accelerazione (asse Z) nel dominio della frequenza	27
3.15	Esempio di grafico nel tempo relativo all' algoritmo SingoloPunto	28
3.16	Esempio di grafico filtrato relativo all' algoritmo SingoloPunto	28
3.17	Esempio di grafico nel tempo relativo all' algoritmo SingoloPunto	29
3.18	Confronto grafici SingoloPunto e PuntoCentroSpalle	30
3.19	Grafico nel tempo relativo alla differenza	30
3.20	Grafico nel tempo relativo alla differenza filtrato	31
3.21	Grafico della differenza nel dominio della frequenza	31
3.22	Confronto grafici SingoloPunto e PuntoCollo	32
3.23	Grafico nel tempo relativo alla differenza	32
3.24	Grafico nel tempo relativo alla differenza filtrato	32
3.25	Grafico della differenza nel dominio della frequenza	33
3.26	Confronto grafici SingoloPunto e SuperficiePuntiDown	33
3.27	Grafico nel tempo relativo alla differenza	34

---

3.28	Grafico nel tempo relativo alla differenza filtrato . . . . .	34
3.29	Grafico nel tempo relativo alla differenza . . . . .	35
4.1	Grafici frequenze relative al soggetto 1 a 2 metri . . . . .	38
4.2	Grafici frequenze relative al soggetto 2 ad 1 metro . . . . .	39
4.3	Grafici frequenze relative al soggetto 3 ad 1 metro . . . . .	40
4.4	Grafici frequenze relative al soggetto 3 a 3 metri . . . . .	41
4.5	Grafici frequenze relative al soggetto 1 . . . . .	43
4.6	Grafici frequenze relative al soggetto 2 . . . . .	44
4.7	Grafici frequenze relative al soggetto 3 . . . . .	45
4.8	Grafici frequenze relative al soggetto 4 . . . . .	46
4.9	Grafici frequenze relative al soggetto 5 . . . . .	47
4.10	Grafici differenza con collo . . . . .	50
4.11	Grafici differenza addome . . . . .	51
4.12	Grafici differenza spalle . . . . .	51
4.13	Grafici differenza con collo . . . . .	54
4.14	Grafici differenza addome . . . . .	55
4.15	Grafici differenza spalle . . . . .	55
4.16	Grafici differenza con collo . . . . .	61
4.17	Grafici differenza addome . . . . .	62
4.18	Grafici differenza spalle . . . . .	62

# Capitolo 1

## Introduzione

Il sistema respiratorio è essenziale per fornire ossigeno ai tessuti e rimuovere l'anidride carbonica, mantenendo così l'omeostasi del corpo. La frequenza respiratoria e il volume corrente si adattano alla domanda metabolica, aumentando durante l'attività fisica o in presenza di malattie. Una frequenza respiratoria elevata può indicare gravi condizioni patologiche ed è utilizzata nei sistemi di valutazione come APACHE, CURB-65 e PSI per identificare pazienti critici [27].

Il monitoraggio accurato della frequenza respiratoria è cruciale per la diagnosi e la gestione delle malattie. Studi dimostrano che un'alta frequenza respiratoria può predire arresto cardiopolmonare e morte [11, 14]. Inoltre il segnale respiratorio permette di individuare schemi respiratori anormali tra cui thoracoabdominal asynchrony/paradox, Kussmaul's breathing, apneustic breathing, respirazione di Cheyne-Stokes, respirazione atassica e di Biot, apnea centrale nel sonno, bradipnea, tachipnea, iperventilazione e ipoventilazione. Riconoscere questi pattern è fondamentale per la diagnosi precoce e la gestione efficace delle malattie respiratorie e sistemiche. Tuttavia, la frequenza respiratoria è spesso sottovalutata o misurata in modo improprio nelle pratiche cliniche [7].

La frequenza respiratoria può essere rilevata utilizzando varie tecniche, che si dividono in due categorie principali: tecniche basate sul contatto e tecniche senza contatto. Tra le principali tecniche basate sul contatto troviamo le tecniche basate sul flusso d'aria respiratorio, che si fondano sulla misura del volume e/o della velocità dell'aria inspirata ed espirata durante la respirazione. I flussometri sono i principali sensori utilizzati per questa tecnica. Altre tecniche includono quelle basate sui suoni respiratori, che prendono come riferimento il suono generato dall'aria che scorre attraverso la gola e le vie aeree del paziente, attraverso ad esempio i sensori microfonic. Un'altra tecnica comune è quella basata sulla temperatura dell'aria: durante la respirazione, l'aria espirata è più calda dell'aria inspirata. Tra i più comuni sensori di temperatura troviamo termistori, termocoppie, sensori piroelettrici e sensori a fibra ottica. Un'altra tecnica è basata sull'umidità dell'aria: l'aria inspirata ed espirata differiscono nel contenuto di vapore acqueo, il che permette di utilizzare sensori di umidità per monitorare la respirazione. Inoltre, le tecniche basate



---

sui componenti dell'aria sfruttano il fatto che, durante la respirazione, il contenuto di ossigeno e anidride carbonica differisce tra l'aria inspirata e quella espirata. I sensori a infrarossi sono comunemente usati per rilevare queste variazioni di gas. Le tecniche basate sull'analisi del movimento della parete toracica utilizzano sensori di deformazione (resistivi, induttivi, capacitivi) e sensori di movimento (giroscopi, magnetometri, accelerometri) per monitorare l'espansione e la contrazione del torace durante la respirazione. Inoltre, le tecniche basate sulla modulazione dell'attività cardiaca considerano l'effetto modulatorio che la respirazione ha sull'attività cardiaca. Sensori ECG (elettrocardiogramma) e PPG (fotoplethismografia) sono utilizzati per rilevare queste modulazioni respiratorie [19]. Tra le tecniche senza contatto troviamo principalmente telecamere ad infrarossi che rilevano i cambiamenti di temperatura associati all'aria espirata e inspirata, radar Doppler e telecamere RGB-D per monitorare i movimenti del torace e dell'addome [8, 1].

I sensori indossabili offrono una misurazione continua e portatile, ma possono causare disagio o irritazione cutanea. Inoltre, richiedono un posizionamento preciso che può essere difficile in pazienti agitati. Le tecniche senza contatto, meno invasive, riducono il rischio di infezioni, possono però essere costose e meno precise.

L'uso delle telecamere RGB per il monitoraggio della respirazione rappresenta un campo di ricerca promettente con potenziali applicazioni cliniche e diagnostiche significative. Nonostante le sfide, i recenti progressi tecnologici e metodologici stanno migliorando l'accuratezza e l'affidabilità di questi sistemi, rendendoli strumenti utili per il monitoraggio non invasivo della salute respiratoria. Esistono diversi articoli che trattano queste tematiche, evidenziando diverse tecniche e approcci. Ad esempio, nell'articolo "A Real-Time Respiration Monitoring and Classification System Using a Depth Camera and Radars" [15], viene presentato un sistema di monitoraggio che utilizza telecamere di profondità (Kinect) e radar insieme ad algoritmi di classificazione come SVM, decision trees e random forest, ottenendo un'accuratezza complessiva del 90%. Un altro studio, "Robust Non-Contact Monitoring of Respiratory Rate using a Depth Camera" [2], misura la frequenza respiratoria mediante una fotocamera Intel RealSense D415 e un algoritmo in tempo reale, raggiungendo una precisione RMSD inferiore a un respiro al minuto (0,69 RPM). Infine, l'articolo "Unobtrusive and Automatic Classification of Multiple People's Abnormal Respiratory Patterns in Real Time Using Deep Neural Network and Depth Camera" [26], utilizza il Kinect v2 e un algoritmo BI-AT-GRU in combinazione con una rete neurale RNN (Recurrent Neural Network), arrivando a un'accuratezza di circa il 95%.

# Capitolo 2

## Materiali

### 2.1 Sensori utilizzati

#### 2.1.1 Intel RealSense D455

L'Intel RealSense D455 è una telecamera di profondità avanzata che sfrutta la tecnologia stereoscopica attraverso due sensori di profondità e un sensore RGB ad alta risoluzione, permettendo di produrre immagini RGB-D. Questo tipo di immagini integrano informazioni di colore con dati di profondità, risultando fondamentali in una vasta gamma di applicazioni, inclusi settori come la robotica, la realtà aumentata, la scansione 3D e l'automazione industriale [23].

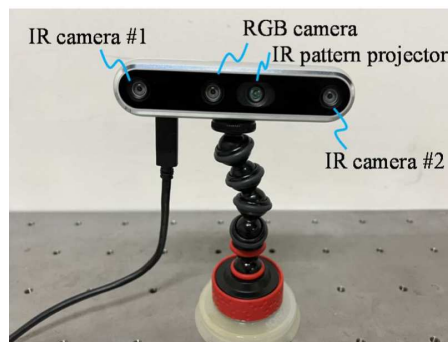


Figura 2.1: Intel RealSense D455 [3]

La tecnologia stereoscopica della D455 si basa su due sensori di profondità (sinistro e destro) posizionati a una distanza fissa di 95 mm tra loro, con la possibilità di utilizzare opzionalmente un proiettore ad infrarossi. Questo proiettore emette un pattern IR statico sulla scena, migliorando l'accuratezza della profondità soprattutto in condizioni di luce scarsa o con elevato contrasto di luminosità. I sensori catturano simultaneamente immagini della stessa scena da due prospettive differenti, permettendo al processore interno della telecamera di confrontare e correlare i punti corrispondenti tra le due immagini. Il processore calcola la differenza o "disparità" tra i punti delle immagini sinistra e destra, utilizzando questa informazione

per derivare valori di profondità attraverso la geometria stereoscopica. I valori di profondità risultanti sono utilizzati per generare un "frame di profondità".

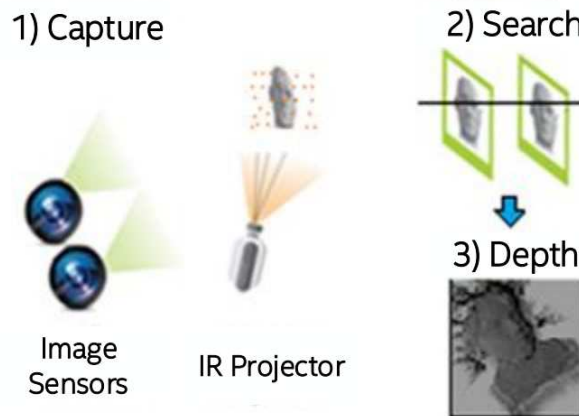


Figura 2.2: Stereo Vision Technology

Come tutte le fotocamere di profondità, affronta sfide comuni, tra cui la presenza della cosiddetta 'Invalid Depth Band'. Questo termine indica una zona specifica nell'immagine di profondità nella quale il sensore della telecamera non è in grado di misurare accuratamente la distanza. Anche se non abbiamo riscontrato in maniera troppo evidente tale fenomeno durante le acquisizioni grazie anche ad un accurata calibrazione della telecamera stessa.

Una caratteristica distintiva della D455 è l'inclusione di un'Unità di Misura Inerziale (IMU), composta da accelerometro e giroscopio, che fornisce dati di movimento e orientamento. Sebbene in questo progetto la telecamera sia stata utilizzata su treppiede senza sfruttare pienamente questa funzionalità, l'IMU è particolarmente utile per applicazioni che richiedono il tracciamento preciso del movimento, come la robotica mobile, i droni e la realtà aumentata. Questo migliora la sincronizzazione dei dati di profondità e colore con i dati di movimento in ambienti dinamici.

La D455 è completamente compatibile con il kit di sviluppo RealSense SDK 2.0, che offre strumenti e API per l'integrazione di applicazioni personalizzate su diverse piattaforme e linguaggi di programmazione, tra cui Python, utilizzato per questo progetto.

Le principali specifiche tecniche sono riportate nella tabella sottostante 2.1.

Portata ideale	Da 0,6 a 6 m
Portata massima	10 m
Unita di misura inerziale	Bosch BMI055
Tecnologia di profondità	Stereoscopico
Massima risoluzione della profondità	1280x720
Frequenza fotogrammi di profondità	Fino a 90 fps
Precisione della profondità	<2% a 4 m
Distanza minima di profondità alla risoluzione massima	52 cm
FOV profondità	87° x 58°
Tecnologia sensore RGB	Global Shutter
Massima risoluzione RGB	1280x800
Frequenza fotogrammi RGB	Fino a 30 fps
FOV sensore RGB(O x V)	90 x 65°
Dimensioni	124 × 26 × 29 mm
Connettori	USB-C 3.1
Supporto al montaggio	Treppiede

Tabella 2.1: Specifiche tecniche Intel RealSense D455

### 2.1.2 Polar H10

Il Polar H10 è un dispositivo indossabile progettato per monitorare in maniera accurata la frequenza cardiaca in una varietà di situazioni di allenamento ed è considerato il cardiofrequenzimetro più accurato da molte fonti [20].



Figura 2.3: Polar H10 con rappresentazione degli assi

Il Polar H10 è in grado di rilevare l'elettrocardiogramma (ECG) basandosi sul principio che ogni battito cardiaco è generato da un impulso elettrico che si propaga attraverso il cuore; il sensore rileva questi impulsi elettrici attraverso elettrodi realizzati in materiali conduttivi. Il design del cinturino, noto come Polar Pro Strap, include elettrodi aggiuntivi rispetto ai modelli precedenti per migliorare la precisione del rilevamento, e consente inoltre di mantenere il dispositivo saldamente in posizione e aderente alla pelle per una accurata misurazione.

Diversi articoli [24, 16, 13] convalidano la fascia toracica Polar H10 per il monitoraggio della frequenza cardiaca dimostrando che i dati ECG raccolti dal sensore sono utili nella pratica clinica per la valutazione del ritmo cardiaco basale, della fibrillazione atriale e delle contrazioni premature. È stata riscontrata una percentuale minima di difficoltà nell'interpretazione delle registrazioni, dovuta a segnali indesiderati come i movimenti muscolari, disturbi elettromagnetici e movimenti del corpo.

Un altro componente chiave del Polar H10, impiegato anche nel progetto proposto, è l'accelerometro, la cui efficacia nel monitoraggio della frequenza cardiaca in condizioni statiche e dinamiche durante l'attività sportiva è stata ampiamente documentata nell'articolo [17]. L'accelerometro a tre assi rileva il movimento nelle direzioni X, Y e Z, ed è progettato per captare anche i piccoli movimenti con elevata precisione, permettendo di distinguere tra vari tipi di attività fisica e di misurare l'intensità del movimento. In particolare, per questo progetto, è stato utile il dato proveniente dall'asse Z dell'accelerometro per monitorare il respiro in condizioni stazionarie.

Per quanto riguarda la connettività, le tecnologie implementate nel Polar H10 includono il Bluetooth Low Energy (BLE), che consente una trasmissione dei dati fino a due dispositivi contemporaneamente ed è efficiente dal punto di vista energetico, garantendo un'ampia compatibilità con numerosi dispositivi. Il BLE assicura anche una connessione stabile e continua, fondamentale durante un allenamento. Un'altra tecnologia per la trasmissione dei dati non utilizzata nel progetto è ANT+, che po-

trebbe permettere di trasmettere dati a più dispositivi contemporaneamente. Un altro punto fondamentale del dispositivo è sicuramente la durata della batteria garantita per 400 ore con una singola sostituzione, permettendoci di lavorare in maniera continua senza la necessità di frequenti cambi di batteria.

Le principali specifiche tecniche sono riportate nella tabella sottostante 2.2.

Dimensioni	34 x 65 x 10 mm
Peso totale	60 g
Peso senza cinturino	21 g
Raggio di trasmissione	9000 cm
Temperatura minima	-10 °C
Temperatura massima	50 °C
Impermeabilità	WR30
Velocità processore	64 MHz
Versione Bluetooth	BLE
Versione ANT+	2.1
Tipologia batteria	CR 2025
Durata batteria	400 h
Registrazione ECG	monocanale
Larghezza di Banda	0.7 - 40 Hz

Tabella 2.2: Specifiche tecniche Polar H10

## 2.2 Acquisizioni

### 2.2.1 Setup sperimentale

Nel nostro studio, abbiamo coinvolto 5 soggetti di genere maschile di età inferiore ai 30 anni. Le misurazioni sono state eseguite sia in condizioni statiche che dinamiche, utilizzando diversi scenari:

- Soggetto seduto diretto verso la telecamera: I soggetti sono stati posizionati staticamente a distanze di 1,2 e 3 metri, guardando direttamente verso la telecamera. La distanza è stata misurata con precisione utilizzando un misuratore laser.



Figura 2.4: Posizione statica da seduto

- Soggetto in piedi, girato: I soggetti sono stati posizionati staticamente ma girati lateralmente rispetto alla telecamera per valutare la capacità di rilevazione della frequenza respiratoria da diverse angolazioni.

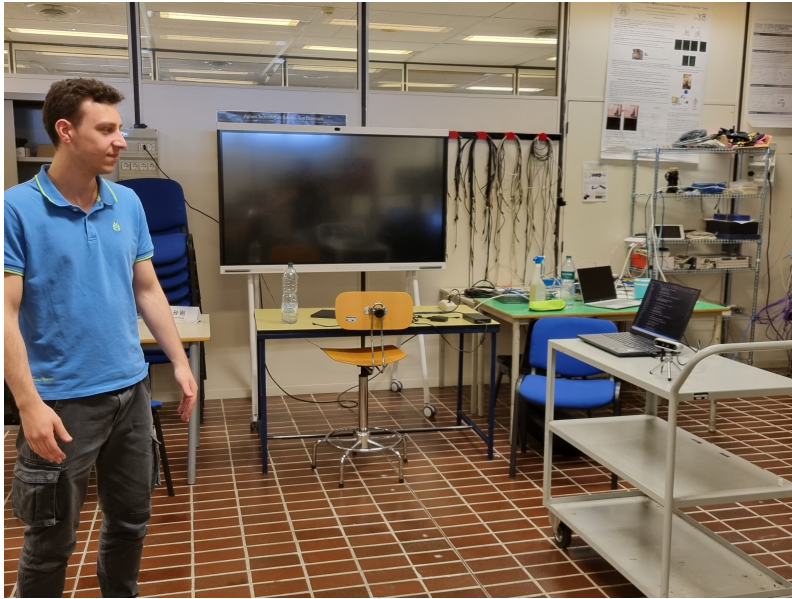


Figura 2.5: Posizione statica leggermente girata

- Soggetto in movimento: Sono stati considerati diversi tipi di movimenti dinamici tra cui quello frontale in cui il soggetto si è mosso avanti e indietro di fronte alla telecamera in modo preciso, poi la semplice rotazione del busto e infine anche movimenti casuali non predicibili, rimanendo nel raggio di visibilità della telecamera in modo da simulare movimenti dinamici che potrebbero verificarsi in scenari reali.

I dati della profondità venivano acquisiti dalla scena attraverso la telecamera Intel RealSense D455 montata su un treppiede e rivolta verso il soggetto monitorato. La telecamera insieme al laptop a cui era collegata sono stati posizionati su un supporto fisso all'interno di un ambiente chiuso e in condizioni di illuminazione stabili e uniformi. La telecamera registrava con una frequenza di 30fps e risoluzione di 640x480. Durante l'acquisizione è stato possibile monitorare i vari frame acquisiti mostrandoli a schermo sul laptop permettendoci così di fermare l'acquisizione nei casi problematici in cui la telecamera acquisiva male la profondità o rilevava soggetti sbagliati nella scena. Ogni soggetto ha indossato una cintura Polar H10 come sensore di riferimento, che acquisiva dati in sincronia con la telecamera. Prima di indossare la cintura, gli elettrodi venivano opportunamente umidificati per garantire acquisizioni più precise.



# Capitolo 3

## Metodi

### 3.1 Algoritmo di acquisizione dalla depth camera

L'algoritmo utilizzato per acquisire i dati dalla depth camera è contenuto all'interno della funzione asincrona `acquire_realsense_data` e si basa su questi punti fondanti:

- **Configurazione della pipeline:** Inizialmente viene creata una pipeline di dati utilizzando la libreria `pyrealsense2` [22] il wrapper Python per Intel RealSense SDK 2.0 che fornisce l'associazione da C++ a Python necessaria per accedere all'SDK. La pipeline è il punto di accesso principale per la gestione del flusso di dati dalla telecamera e viene configurata per acquisire sia il flusso di profondità che il flusso di colore con una risoluzione di 640x480 pixel e una frequenza di 30 fps. I flussi di dati vengono poi allineati per garantire la corrispondenza tra i dati di profondità e di colore.
- **Acquisizione dei Dati:** In maniera ciclica i frame di colore e profondità vengono acquisiti dalla camera, allineati e verificati per assicurarsi che siano validi fino al raggiungimento di un tempo di esecuzione predefinito `EXECUTION_TIME`. Per ogni coppia di frame di colore e profondità otteniamo anche il relativo timestamp. I dati di frame e timestamp vengono convertiti in array NumPy e ritornati alla funzione chiamante ovvero il `main`. I frame raccolti vengono inoltre visualizzati in tempo reale utilizzando OpenCV, permettendo un monitoraggio visivo dell'acquisizione.
- **Interruzione della Pipeline:** Al termine del periodo di acquisizione, la pipeline viene fermata per interrompere il flusso di dati e liberare le risorse.
- **Salvataggio dei Dati:** I dati acquisiti vengono poi salvati su disco utilizzando il formato `.npy` in modo da essere processati successivamente.

## 3.2 Algoritmo di acquisizione dal Polar

L'algoritmo di acquisizione dei dati dal dispositivo Polar H10 è progettato per raccogliere informazioni elettrocardiografiche (ECG) e di accelerazione (ACC) in tempo reale. Di seguito sono descritti i punti principali:

- **Connessione:** L'algoritmo inizia con la configurazione e la connessione al dispositivo Polar H10 mediante il suo indirizzo MAC, che cambia nel caso di spegnimento del dispositivo. Utilizzando Bleak [21], una libreria per la gestione di dispositivi BLE in Python, viene creato un client per gestire la connessione e le operazioni di lettura/scrittura delle caratteristiche GATT (Generic Attribute Profile) come il numero di modello, nome del produttore, livello della batteria oltre ai servizi per l'ECG e accelerometro. Ogni servizio e caratteristica GATT è identificata da un UUID (Universally Unique Identifier) definito inizialmente. La connessione al dispositivo viene poi verificata tramite la funzione *connect\_to\_polar* nella quale se la connessione è andata a buon fine vengono lette le informazioni essenziali tra cui il numero di modello, il nome del produttore e il livello della batteria. Questo ci assicura che il Polar sia correttamente connesso e pronto per la raccolta dati.
- **Raccolta dati:** All'interno della funzione asincrona *acquire\_polar\_data* vengono inviati comandi di scrittura per iniziare la raccolta dei dati ECG e di accelerazione dal dispositivo Polar H10. Le notifiche vengono abilitate per entrambi i tipi, permettendo al dispositivo di inviare continuamente i dati al client attraverso il protocollo Bluetooth per un periodo di tempo definito (EXECUTION\_TIME). Durante la ricezione dei dati ECG, la funzione di callback *ecg\_data\_conv* è attivata ad ogni pacchetto ricevuto. Questa funzione verifica prima di tutto che il pacchetto di dati faccia riferimento ai dati ECG controllando che il primo byte di dati sia 0x00 successivamente estrae il timestamp dal pacchetto e converte i dati ECG da array di byte a numeri interi con segno. I dati vengono quindi memorizzati insieme ai rispettivi timestamp. Analogamente ai dati ECG, la funzione di callback *acc\_data\_conv* viene chiamata ogni volta che vengono ricevuti nuovi dati dall'accelerometro. Verifica che il primo byte dei dati sia 0x02, estrae il timestamp e converte i campioni accelerometrici (x, y, z) da array di byte a numeri interi con segno. I dati di accelerazione vengono anch'essi memorizzati insieme ai rispettivi timestamp.
- **Salvataggio dei Dati:** Al termine della raccolta dei dati, tutte le informazioni raccolte vengono salvate su disco in formato NumPy in modo tale da poterli elaborare in futuro.
- **Disconnessione:** Infine, il client si disconnette dal dispositivo Polar H10, garantendo una corretta liberazione delle risorse di rete e una chiusura ordinata delle operazioni di comunicazione.

## 3.3 Sincronizzazione dei segnali

Nell'ottica di sincronizzare l'acquisizione dei dati dal Polar e dalla telecamera RealSense abbiamo utilizzato la libreria *asyncio* [12], una libreria standard di Python che supporta la scrittura di codice concorrente asincrono utilizzando il paradigma di programmazione asincrona. Questa libreria è progettata per gestire operazioni asincrone in modo efficiente, consentendo al programma di eseguire altre operazioni mentre aspetta che operazioni lunghe di I/O (Input/Output) o di altro tipo siano completate. Abbiamo impiegato in particolare il metodo *gather*, che consente di eseguire più coroutines asincrone contemporaneamente e di attendere che tutte completino la loro esecuzione prima di procedere. Nel nostro caso, le coroutines utilizzate erano *acquire\_realsense\_data* e *acquire\_polar\_data*, che venivano eseguite simultaneamente per garantire la sincronizzazione delle acquisizioni dai due dispositivi. All'interno di ciascuna coroutine, abbiamo utilizzato il metodo *sleep* della libreria *asyncio* per gestire il tempo di esecuzione e assicurare che entrambe le coroutines terminassero allo stesso momento.

## 3.4 Elaborazione dei dati attraverso i grafici

### 3.4.1 Processamento telecamera RealSense

Abbiamo deciso di eseguire un post-processing dei dati poichè eseguire l'acquisizione in contemporanea con il processamento era molto oneroso dal punto di vista computazionale rischiando di perdere alcuni frame. Inizialmente si caricano i dati relativi al colore e alla profondità che erano stati acquisiti nella sezione 3.1, viene successivamente caricato un modello pre-addestrato yolov8n-pose per il rilevamento delle pose.

YOLO (You Only Look Once) è un framework per il rilevamento degli oggetti che si distingue per il suo equilibrio tra velocità e precisione. YOLOv8 utilizzato nel nostro progetto, rilasciato da Ultralytics nel gennaio 2023, include miglioramenti significativi rispetto alle versioni precedenti. YOLOv8 utilizza un'architettura avanzata con il backbone CSPDarknet53 modificato e un modulo C2f per una migliore contestualizzazione e precisione nel rilevamento. Adotta un modello senza ancoraggi, predicendo direttamente le bounding boxes senza dover utilizzare punti di ancoraggio predefiniti. Questo contribuisce a una maggiore flessibilità nel rilevamento di oggetti di varie forme e dimensioni. Inoltre ha una testa disaccoppiata che gestisce separatamente oggettività, classificazione e regressione. Utilizza funzioni di attivazione sigmoide per il punteggio di oggettività e softmax per la classificazione, con funzioni di perdita CIoU e DFL per migliorare la precisione, specialmente per oggetti più piccoli [25].

Nel post-processing, per ogni coppia di frame (colore e profondità), vengono individuate le varie bounding boxes presenti nella scena, tramite un filtraggio viene presa in considerazione soltanto la prima che viene identificata e con una confidenza mag-

giore del 70%. All'interno di questa box, vengono rilevati i keypoints, tra cui il naso, le spalle e i punti del bacino, di particolare interesse per il nostro caso.

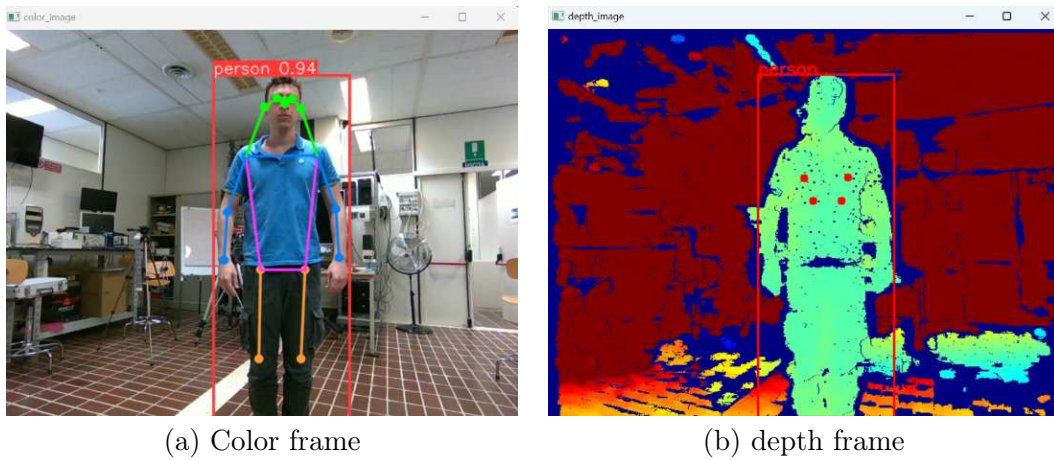


Figura 3.1: Frame processati con box e keypoints

Una volta individuati i keypoints vengono applicati i diversi algoritmi per il calcolo della profondità all'interno delle maschere di interesse tra cui:

- *PuntoCentroSpalle*: Calcola la profondità al punto centrale tra due spalle.

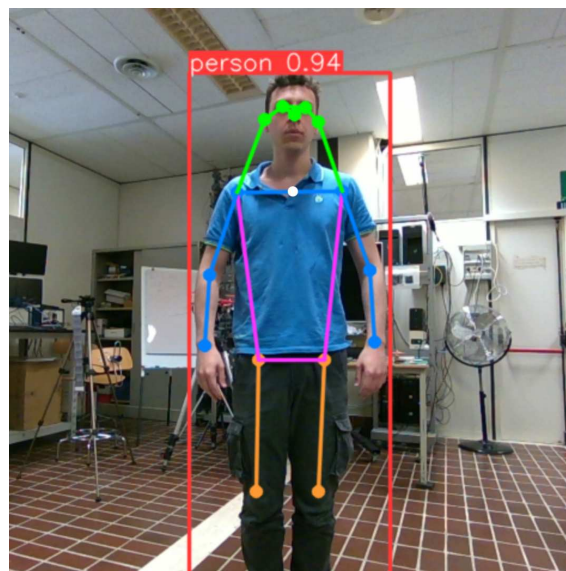


Figura 3.2: Punto centrale tra le spalle

- *PuntoCollo*: Calcola la profondità nel punto medio tra il centro delle spalle e il naso, che approssima la posizione del collo.

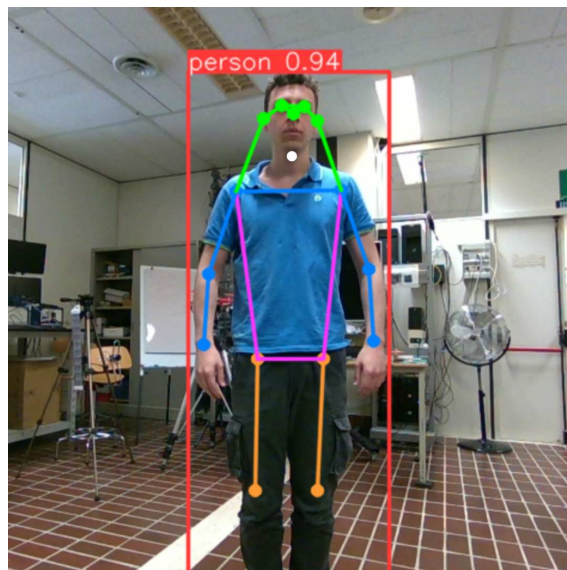


Figura 3.3: Punto sul collo

- *SingoloPunto*: Calcola la profondità in un punto specifico all'interno del torso, determinato come il centroide del triangolo formato dalle spalle e dal centro del torso.

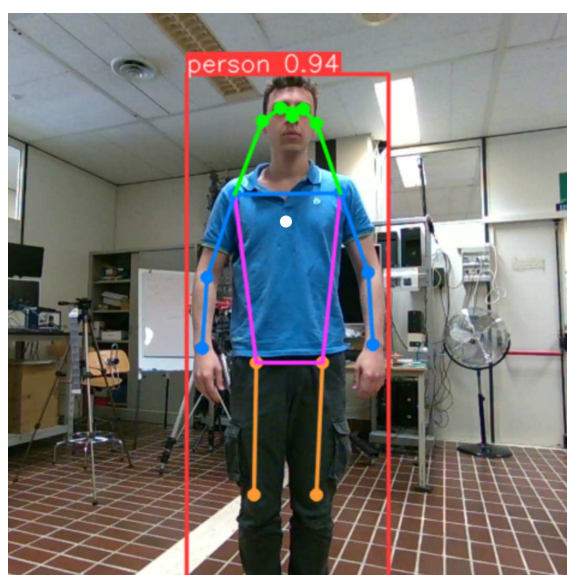


Figura 3.4: Punto all'interno del torso

- *SuperficiePuntiUp*: Calcola la profondità media della regione superiore del torso, delimitata dalle spalle e dai punti medi tra spalle e bacino.

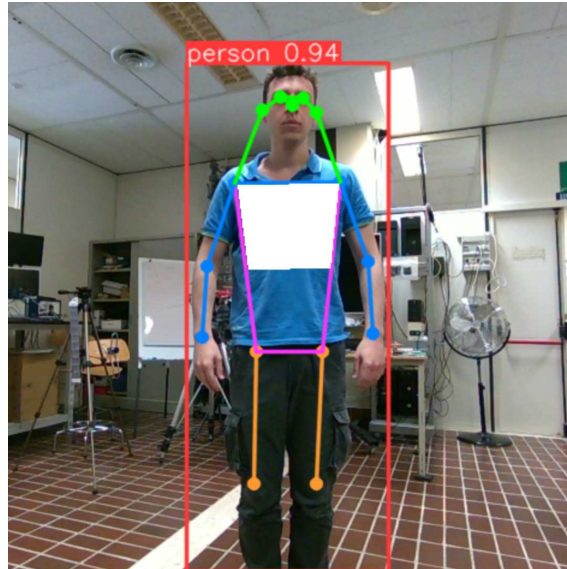


Figura 3.5: Maschera della parte superiore del torso

- *SuperficiePuntiDown*: Calcola la profondità media della regione inferiore del torso, delimitata dai punti medi tra spalle e bacino e dai punti del bacino.

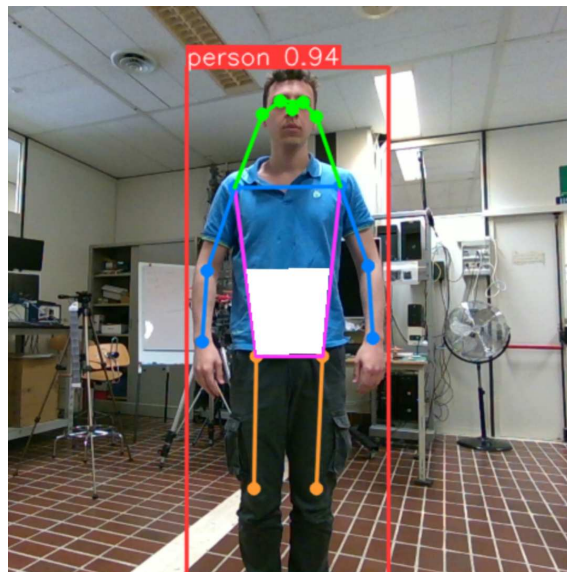


Figura 3.6: Maschera della parte inferiore torso

- *SuperficiePunti*: Calcola la profondità media dell'intera regione del torso, delimitata dalle spalle e dai punti del bacino.

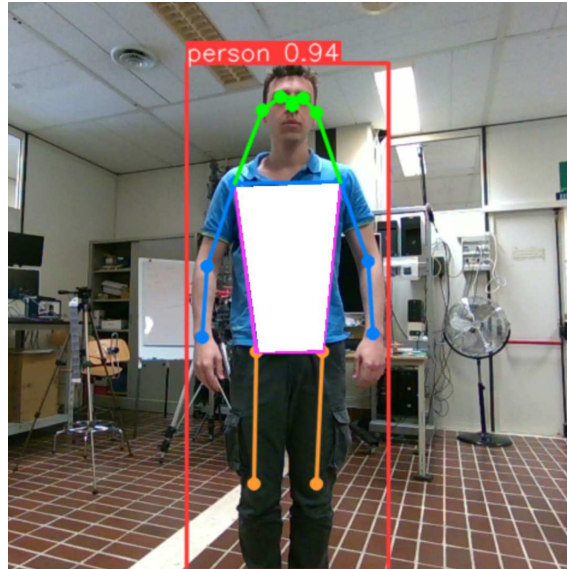


Figura 3.7: Maschera del torso

I valori della profondità ottenuti da ogni algoritmo vengono poi salvati su liste separate per poi essere convertite in array numpy e salvati in file .npy che utilizzeremo poi per fare il plotting dei dati tramite grafici.

### 3.4.2 Segnale ECG estratto dal PolarH10

Per elaborare il segnale ECG, inizialmente vengono caricati i dati ECG ottenuti durante le acquisizioni descritte nella sezione 3.2. Successivamente, scartiamo i primi 5 secondi di dati, poiché nelle prove eseguite abbiamo rilevato che erano inaccurati, probabilmente a causa della connessione iniziale con il dispositivo. Utilizzando le funzionalità offerte dalla libreria BioSPPy [5], abbiamo analizzato il segnale ECG rimanente, identificando i complessi QRS [10]. Il complesso QRS rappresenta la depolarizzazione elettrica dei ventricoli del cuore ed è composto da una serie di onde che indicano diversi eventi elettrici durante il ciclo cardiaco.

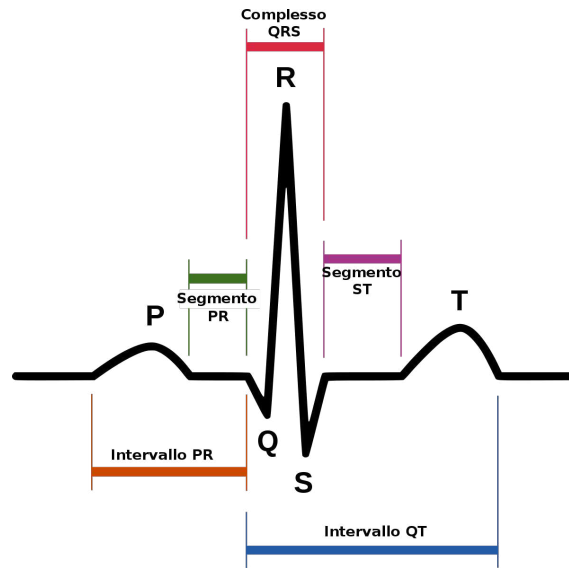


Figura 3.8: Complesso QRS

Componenti principali:

- **Onda Q**: È la prima deflessione negativa subito prima del picco R. Non è sempre visibile o presente.
- **Picco R**: È il punto più alto e prominente del complesso QRS. Rappresenta la depolarizzazione rapida dei ventricoli del cuore, che è l'evento elettrico principale durante la contrazione ventricolare durante la quale i ventricoli si contraggono per pompare il sangue.
- **Onda S**: È la deflessione negativa che segue immediatamente il picco R. Indica la continuazione della depolarizzazione, ma in senso opposto.

Dopo aver individuato i complessi QRS si è proceduto con il calcolo dei picchi R. Dividendo il numero di complessi QRS per il tempo trascorso (meno i primi 5 secondi scartati) e moltiplicando il risultato per 60 si ottiene la frequenza cardiaca approssimativa (BPM). Infine viene fatto un plot del grafico ECG nel tempo con annotati i vari picchi R e il numero di BPM.



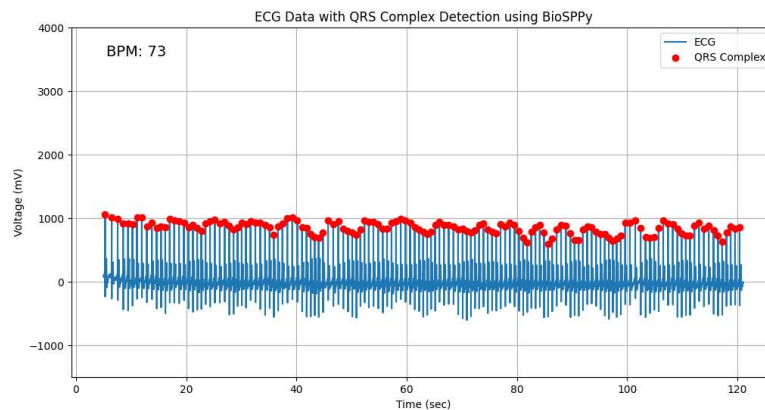


Figura 3.9: Esempio di grafico ECG con annotati i picchi e BPM

### 3.4.3 Segnale respiratorio estratto dal PolarH10

Come nella sezione 3.4.2, per poter ricavare il segnale respiratorio utilizziamo i dati acquisiti precedentemente dal Polar nella sezione 3.2 scartando i primi 5 secondi.

Durante la respirazione, la frequenza cardiaca varia in risposta al ciclo respiratorio; tale fenomeno è noto come "Modulazione Respiratoria della Frequenza Cardiaca" (Respiratory Sinus Arrhythmia - RSA) [4]. Durante l'inspirazione, la frequenza cardiaca tende ad aumentare, mentre durante l'espirazione tende a diminuire. Questo pattern può somigliare a una forma d'onda sinusoidale quando si analizzano i picchi R in sequenza come mostrato nel grafico 3.9.

Sfruttando le funzionalità offerte dalla libreria Neurokit2 [18], un pacchetto Python open source per l'elaborazione dei segnali neurofisiologici, siamo stati in grado di identificare inizialmente i picchi R dei vari complessi QRS e, da questi, calcolare la frequenza cardiaca. Utilizzando la funzione `ecg_rsp` di Neurokit2, abbiamo estratto il segnale respiratorio dall'andamento della frequenza cardiaca, sfruttando il fenomeno della RSA. La funzione di default applica il metodo `vangent2019`, anche se è possibile usarne altri come `soni2019`, `sarkar2015` e `charlton2016`, di cui di seguito verrà fornita una spiegazione dettagliata.

Il metodo `Charlton2016` [6] è una tecnica specifica proposta da Charlton et al. nel 2016 utilizzata per estrarre il segnale respiratorio dal segnale della frequenza cardiaca (ECG). Il primo passo consiste nel filtrare il segnale ECG soggetto a disturbi elettrici e rumore per isolare le componenti di interesse associate alla modulazione respiratoria della frequenza cardiaca. Questo filtraggio può includere tecniche come il filtraggio passa-banda nel range di frequenze 0.066-1Hz per rimuovere le frequenze indesiderate e isolare le variazioni respiratorie. Una volta che il segnale ECG è stato preparato tramite filtraggio, il metodo ricorre all'uso di una trasformata di Fourier adattiva per analizzare il segnale. Questa trasformata permette di adattarsi alle variazioni temporali nel segnale, identificando le componenti nel dominio delle frequenze che corrispondono alla modulazione respiratoria della frequenza cardiaca.

Questa fase è cruciale per individuare con precisione le variazioni ritmiche associate alla respirazione. Le componenti identificate tramite la trasformata di Fourier adattiva possono presentare rumore o variazioni rapide che potrebbero non riflettere accuratamente il ritmo respiratorio desiderato. Perciò, le componenti estratte vengono sottoposte a un processo di smoothing per ridurre il rumore e migliorare la coerenza del segnale respiratorio ottenuto. Questo passaggio aiuta a produrre un segnale EDR più stabile e affidabile. Una volta completati i passaggi precedenti, il segnale respiratorio EDR viene generato combinando le componenti ritmiche identificate. Questo segnale rappresenta la variazione ciclica della frequenza cardiaca correlata alla respirazione del soggetto.

Ritornando al codice originario una volta ottenuto il segnale respiratorio lo andiamo a graficare nel tempo.

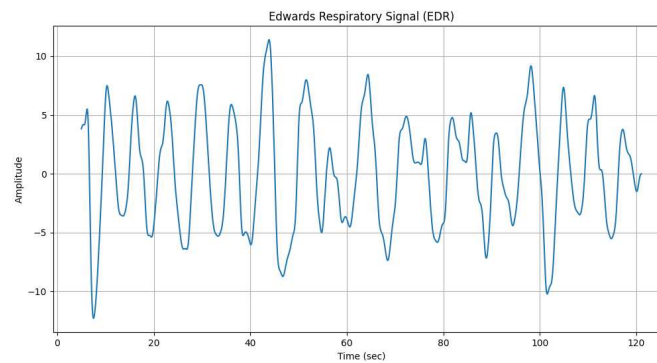


Figura 3.10: Esempio di grafico del segnale respiratorio nel tempo

Dal grafico 3.10 è possibile calcolare il numero di respiri nell'intervallo di tempo considerato contando il numero di picchi del segnale, ognuno rappresenta un ciclo respiratorio (inspirazione o espirazione). In questo esempio sono circa 17. Per ottenere però una stima più esatta dei respiri e della frequenza respiratoria si procede con il calcolo della trasformata di Fourier per convertire il segnale EDR nel dominio della frequenza e si applica una maschera per selezionare le frequenze tra 0.1 e 1 Hz che corrispondono al range tipico della frequenza respiratoria. Successivamente viene identificato l'indice del picco massimo nello spettro di potenza delle frequenze selezionate e la frequenza corrispondente a questo picco rappresenta la frequenza respiratoria predominante. Infine viene fatto il plot dello spettro di potenza del segnale EDR nel range di frequenze selezionato con annotata la frequenza di picco.

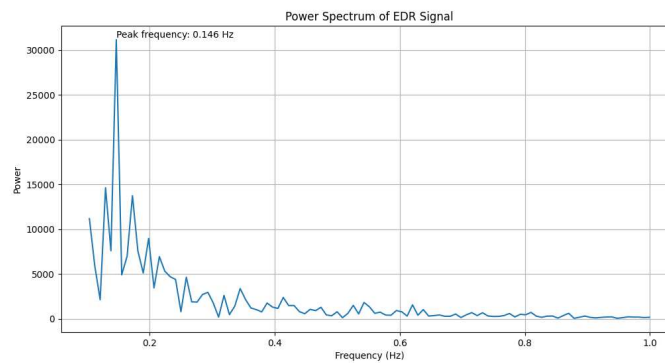


Figura 3.11: Esempio di grafico del segnale respiratorio nel dominio della frequenza

Dal grafico 3.11 possiamo calcolare il numero di respiri prendendo in considerazione la frequenza di picco e moltiplicandola per il tempo di acquisizione scartando i primi 5 secondi nell'esempio riportato sono 115 secondi quindi avremo un numero di respiri pari a 16.79 che coincide con il risultato ottenuto considerando il grafico nel tempo.

#### 3.4.4 Segnale asse Z estratto dal PolarH10

Per elaborare il segnale dell'asse Z carichiamo inizialmente i dati relativi all'accelerazione acquisiti dal Polar precedentemente nella sezione 3.2. Tra i dati delle accelerazioni lungo i vari assi vengono estrapolati solo quelli relativi all'asse Z e convertiti in metri al secondo quadro, come sempre vengono scartati i primi 5 secondi. Viene poi visualizzato il grafico nel dominio del tempo.

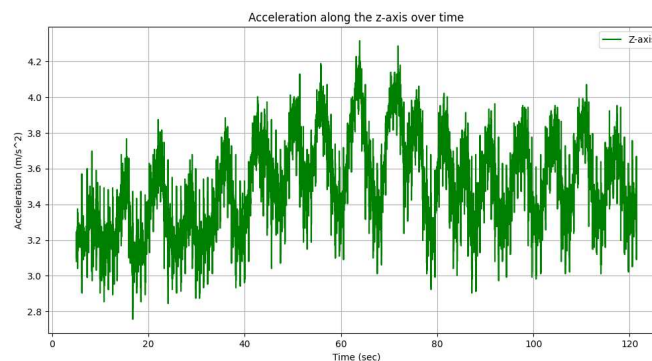


Figura 3.12: Esempio di grafico dell' accelerazione (asse Z) nel tempo

Il segnale di accelerazione viene detrendato utilizzando la funzione *signal.detrend* della libreria SciPy [9] che si occupa di rimuovere la tendenza lineare dal segnale in modo tale che oscilli attorno allo zero. Questo può essere utile per il filtraggio o la trasformata di Fourier, dove la presenza di una tendenza lineare potrebbe distorcere i risultati. Successivamente è stato progettato un filtro passa-banda di Butterworth per isolare le frequenze nel range di interesse (da 0.1 Hz a 0.8 Hz). Il filtro è stato

realizzato tramite la funzione *signal.butter* della libreria SciPy e applicato al segnale con la funzione *signal.filtfilt*. Il segnale così ottenuto (vedi Figura 3.13) presenta una notevole riduzione del rumore permettendoci quindi di valutare i picchi per il calcolo dei respiri.

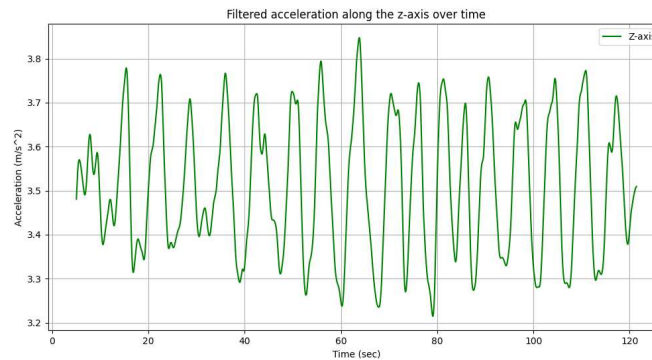


Figura 3.13: Esempio di grafico dell' accelerazione (asse Z) detrendata e filtrata

Viene eseguita poi la Trasformata di Fourier (FFT) per convertire il segnale dal dominio del tempo a quello della frequenza. Si calcola lo spettro di potenza delle frequenze selezionate tramite FFT per identificare le componenti di frequenza dominanti nel segnale. Viene individuato il picco di frequenza nel range di interesse specificato (da 0.1 Hz a 1 Hz) frequenze tipiche della respirazione, utilizzando l'indice del valore massimo nello spettro di potenza. Il picco di frequenza corrisponderà quindi alla frequenza respiratoria.

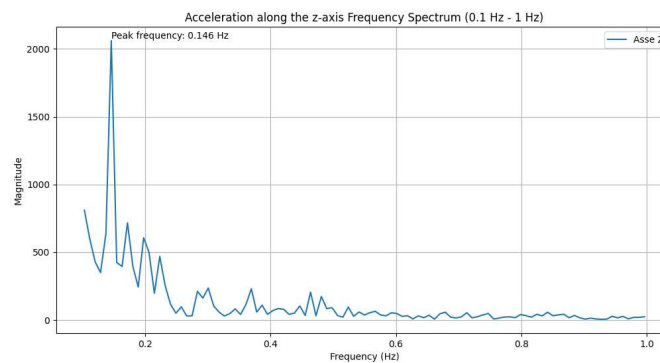


Figura 3.14: Esempio di grafico dell'accelerazione (asse Z) nel dominio della frequenza

## 3.5 Elaborazione dei dati caso statico

Una volta processati i dati provenienti dalla telecamera RealSense attraverso i vari algoritmi spiegati nella sottosezione 3.4.1 possiamo elaborare e graficare i risultati

di profondità ottenuti ricavando così la frequenza respiratoria. I grafici riportati in questa sezione fanno riferimento alle stesse condizioni citate nella sezione precedente ovvero soggetto 2 seduto ad un metro.

### 3.5.1 Segnale respiratorio estratto dalla telecamera

Vengono caricati tutti i dati di profondità provenienti da uno dei quattro algoritmi SingoloPunto, SuperficiePuntiUp, SuperficiePuntiDown, SuperficiePunti insieme ai timestamps. Anche in questo caso vengono scartati i primi 5 secondi di dati per poter fare un confronto omogeneo con i risultati ottenuti dal Polar. Successivamente facciamo un controllo sui dati in modo da verificarne la correttezza e la consistenza, nel caso di dati sbagliati dovuti ad errori di acquisizione li andiamo a sostituire con il dato precedente nella sequenza. Il grafico 3.15 mostra i dati di profondità nel tempo.

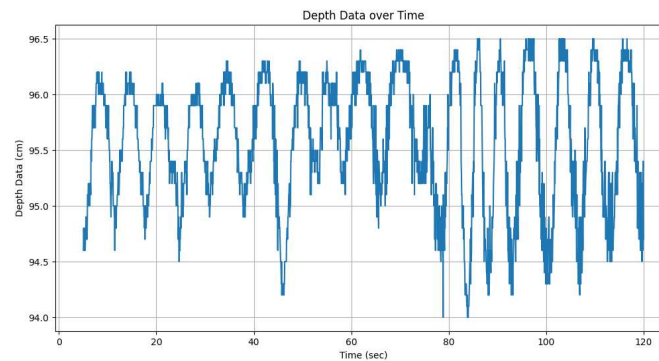


Figura 3.15: Esempio di grafico nel tempo relativo all'algoritmo SingoloPunto

Le oscillazioni sinusoidali presenti nel grafico dei dati di profondità descrivono il movimento del torace durante la respirazione. Per rimuovere il rumore si esegue il filtraggio mediante un filtro passa-banda di Butterworth tra le frequenze 0.1 e 0.8Hz.

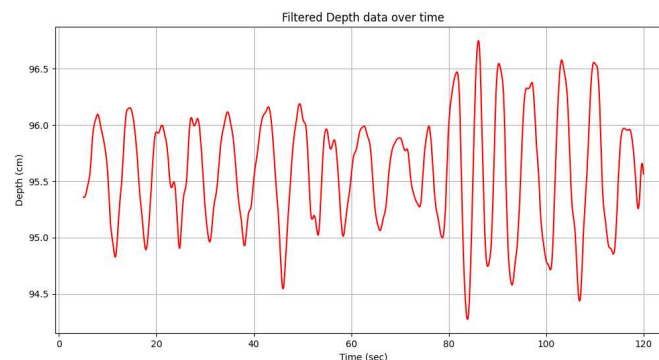


Figura 3.16: Esempio di grafico filtrato relativo all'algoritmo SingoloPunto

Infine si calcola la trasformata di Fourier (FFT) dei dati di profondità non filtrati per ottenere lo spettro di potenza delle frequenze presenti nei dati. Viene calcolata la magnitudine dell'FFT e vengono selezionate solo le frequenze comprese tra 0.05 Hz e 1 Hz per il plot. Viene individuato il picco nello spettro di potenza e la frequenza corrispondente che rappresenta quindi la frequenza respiratoria. Il grafico 3.17 riporta le magnitudini dell'FFT rispetto alle frequenze selezionate con annotato il picco di frequenza.

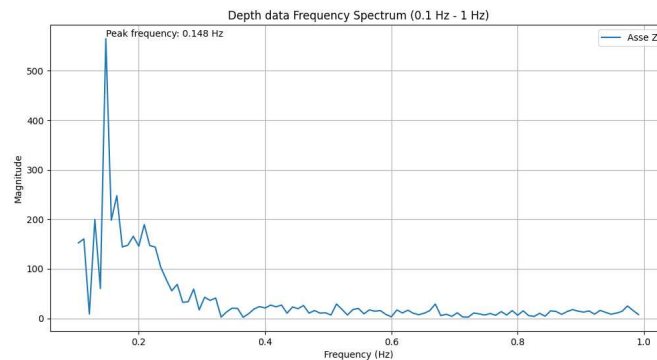


Figura 3.17: Esempio di grafico nel tempo relativo all'algoritmo SingoloPunto

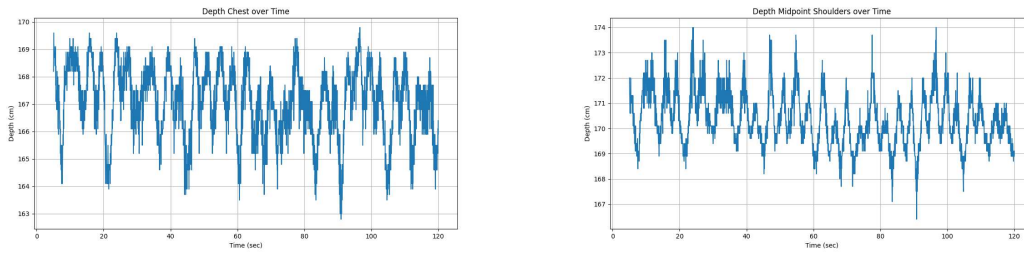
## 3.6 Elaborazione dei dati caso dinamico

Nel caso dinamico, l'analisi della respirazione diventa più complessa a causa della sovrapposizione del movimento respiratorio microscopico del torace con il movimento macroscopico del soggetto nello spazio. Per affrontare questa complessità, sono stati sviluppati algoritmi in grado di estrapolare le variazioni di profondità dovute esclusivamente al movimento respiratorio. Il principio alla base degli algoritmi che verranno spiegati di seguito è stato cercare delle aree del corpo che non siano soggette al movimento causato dalla respirazione ma solo a quello del movimento del soggetto. Esempi di queste aree includono il punto centrale tra le spalle, il collo e l'addome. Sottraendo i dati di profondità provenienti dai quattro algoritmi nel caso statico da quelli provenienti da queste zone è possibile isolare le variazioni di profondità legate esclusivamente alla respirazione. I grafici riportati in questa sezione fanno riferimento al soggetto 2 con rotazione del torso, per esemplificare la descrizione prendiamo come riferimento per gli algoritmi in condizioni statiche l'algoritmo SingoloPunto sebbene sia possibile utilizzare uno qualunque degli altri.

### 3.6.1 Segnale respiratorio estratto dalla differenza con il punto centrale delle spalle

Inizialmente vengono caricati i timestamps e i dati di profondità dell'algoritmo SingoloPunto insieme a quelli provenienti dall'algoritmo PuntoCentroSpalle descritto nella sottosezione 3.4.1. Per entrambe le tipologie di dati viene implementato un

processo di sostituzione dei dati con il valore precedente nel caso essi siano nulli, questo viene fatto per gestire eventuali errori dovuti all'acquisizione della telecamera. Come sempre vengono scartati i primi 5 secondi per avere un confronto omogeneo con gli altri grafici. Viene poi plottato il grafico nel tempo relativo all'algoritmo SingoloPunto e quello relativo all'algoritmo PuntoCentroSpalle in modo tale da poterli confrontare visivamente.



(a) Grafico nel tempo relativo all'algoritmo SingoloPunto

(b) Grafico nel tempo relativo all'algoritmo PuntoCentroSpalle

Figura 3.18: Confronto grafici SingoloPunto e PuntoCentroSpalle

Successivamente viene calcolata la differenza di profondità tra i dati relativi all'algoritmo SingoloPunto e quelli del punto centrale delle spalle. Il risultato viene poi plottato nel tempo.

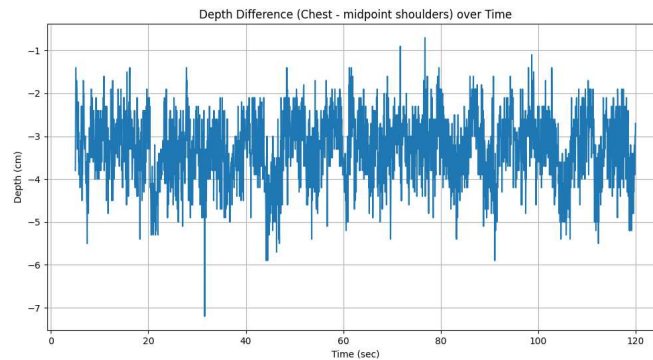


Figura 3.19: Grafico nel tempo relativo alla differenza

Viene eseguito un filtraggio passa-banda sul segnale di differenza di profondità. Questo serve a isolare le frequenze di interesse, comprese tra 0.13 Hz e 0.8 Hz, eliminando il rumore fuori banda.

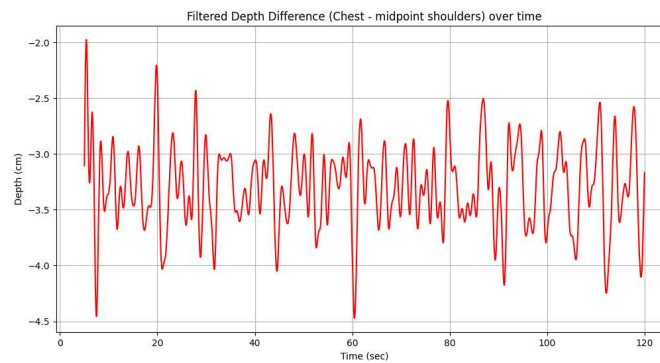


Figura 3.20: Grafico nel tempo relativo alla differenza filtrato

Si esegue infine un'analisi spettrale del segnale di differenza di profondità non filtrata. Utilizzando la trasformata di Fourier (FFT), il segnale viene convertito dal dominio del tempo al dominio delle frequenze, permettendo di esaminare le componenti sinusoidali che lo compongono. Vengono selezionate tramite una maschera solo le frequenze comprese tra 0.1 Hz e 1Hz, intervallo rilevante per la respirazione. Tra queste, si identifica la frequenza con la massima magnitudine, che rappresenta la frequenza respiratoria. Questa frequenza di picco viene visualizzata nel grafico 3.21 insieme allo spettro di frequenza.

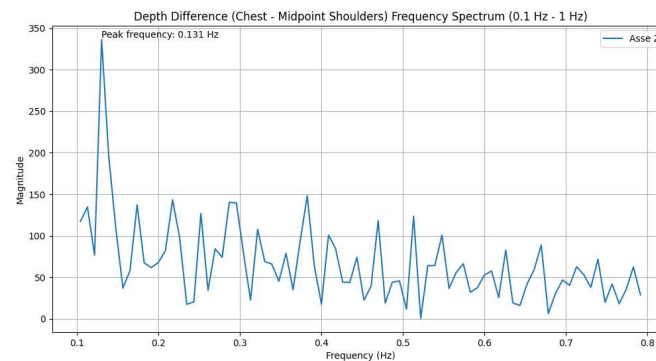
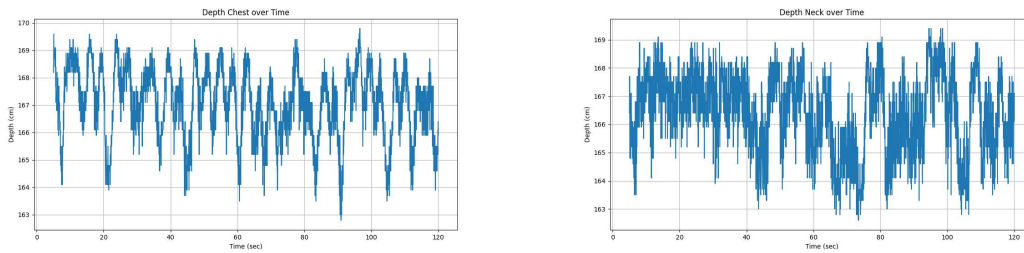


Figura 3.21: Grafico della differenza nel dominio della frequenza

### 3.6.2 Segnale respiratorio estratto dalla differenza con il collo

Come nella sottosezione 3.6.1 vengono caricati i timestamps e i dati di profondità dell'algoritmo SingoloPunto insieme però in questo caso a quelli provenienti dall'algoritmo PuntoCollo descritto nella sottosezione 3.4.1. Come già spiegato vengono eseguiti dei controlli sulla nullità dei dati e in tal caso si applica la sostituzione del valore con il valore precedente e vengono scartati i primi 5 secondi. Viene poi plottato il grafico nel tempo relativo all'algoritmo SingoloPunto e quello relativo all'algoritmo PuntoCollo in modo tale da poterli confrontare visivamente.





(a) Grafico nel tempo relativo all'algoritmo SingoloPunto      (b) Grafico nel tempo relativo all'algoritmo PuntoCollo

Figura 3.22: Confronto grafici SingoloPunto e PuntoCollo

Viene poi fatta la differenza tra i dati di profondità dei due algoritmi e ne viene plottato il risultato nel tempo.

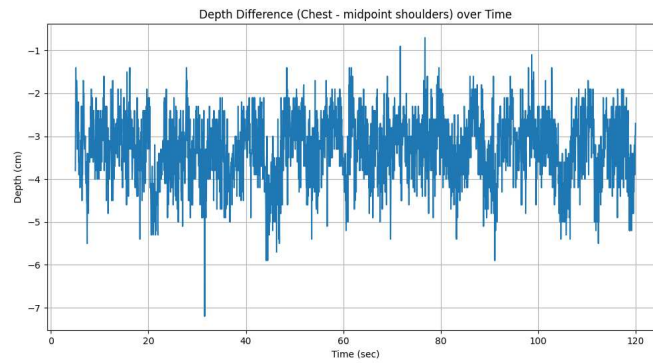


Figura 3.23: Grafico nel tempo relativo alla differenza

Sucessivamente viene realizzato un filtraggio passa-banda sul segnale di differenza. Questo serve a isolare le frequenze di interesse, comprese tra 0.13 Hz e 0.8 Hz, eliminando il rumore fuori banda e viene plottato il risultato ottenuto.

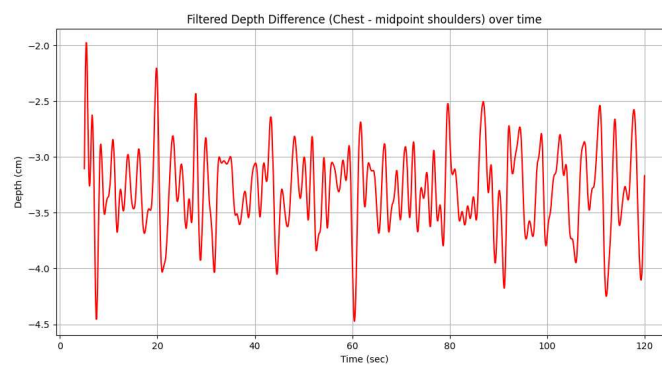


Figura 3.24: Grafico nel tempo relativo alla differenza filtrato

Viene eseguita la trasformata di Fourier sui dati di profondità per trasformarli dal dominio del tempo a quello della frequenza. Questo permette di analizzare le componenti di frequenza del segnale. Viene applicata una maschera per selezionare solo le frequenze comprese tra 0.13 Hz e 0.8 Hz, che corrispondono alla gamma di frequenze respiratorie. Si identifica il picco principale nello spettro di frequenza, che rappresenta la frequenza respiratoria dominante. Viene generato un grafico dello spettro di frequenza, mostrando le componenti di frequenza dei dati di profondità insieme alla frequenza di picco.

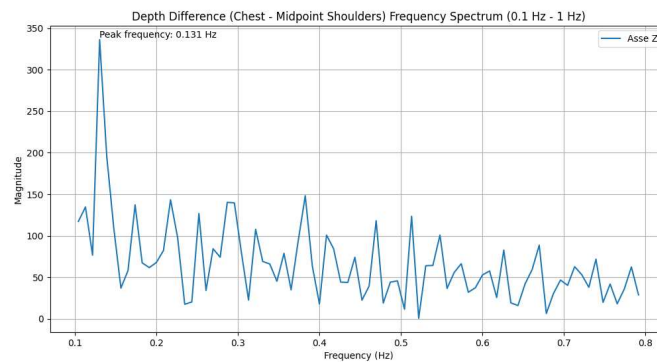
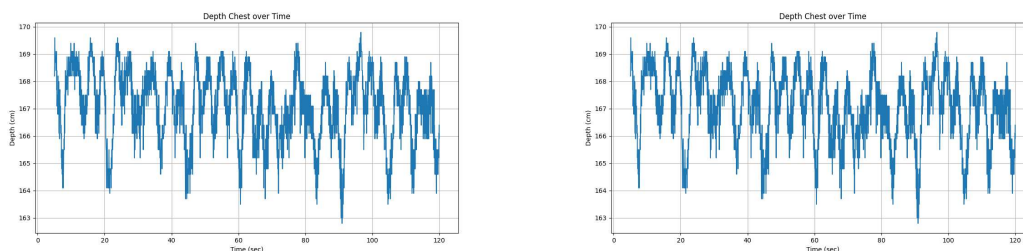


Figura 3.25: Grafico della differenza nel dominio della frequenza

### 3.6.3 Segnale respiratorio estratto dalla differenza con l'addome

Come nei casi precedenti carichiamo i timestamps ,i dati di profondità relativi all'algoritmo SingoloPunto insieme a quelli dell'algoritmo SuperficiePuntiDown. Come al solito vengono svolti i controlli sulla correttezza dei dati e scartati i primi 5 secondi. Viene poi plottato il grafico nel tempo relativo all'algoritmo SingoloPunto e quello relativo all'algoritmo SuperficiePuntiDown in modo tale da poterli confrontare visivamente.



(a) Grafico nel tempo relativo all'algoritmo SingoloPunto (b) Grafico nel tempo relativo all'algoritmo SuperficiePuntiDown

Figura 3.26: Confronto grafici SingoloPunto e SuperficiePuntiDown

Viene poi fatta la differenza tra i dati di profondità dei due algoritmi e ne viene plottato il risultato nel tempo. In questo ultimo caso come algoritmi per fare la differenza non possiamo prendere quello SuperficiePuntiDown che ci fornisce la media della profondità di tutti i punti della fascia addominale perché poi sottraendolo a se stesso otterremmo un segnale nullo.

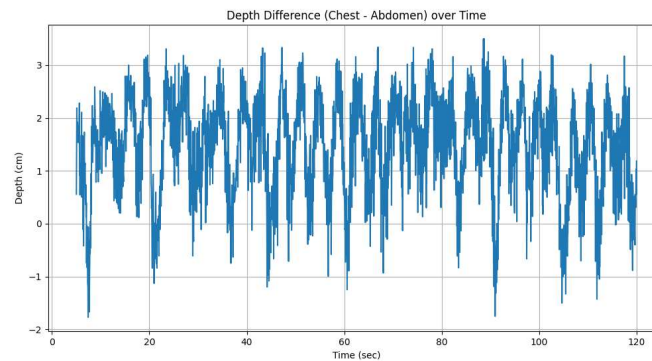


Figura 3.27: Grafico nel tempo relativo alla differenza

Viene eseguito un filtraggio passa-banda con la stessa maschera di frequenze e plottato il segnale filtrato.

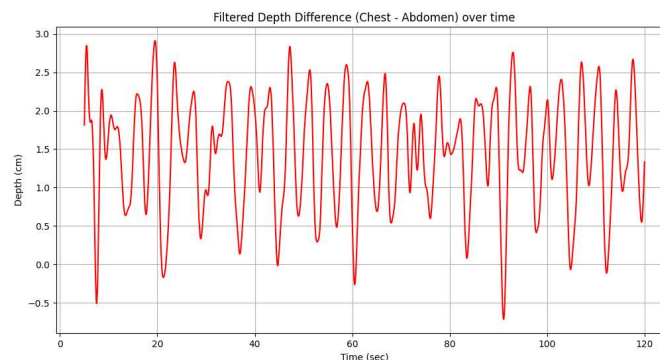


Figura 3.28: Grafico nel tempo relativo alla differenza filtrato

Infine viene fatta la trasformata di Fourier, determinato il picco di frequenza e plottato il grafico del segnale nel dominio della frequenza con annotata la frequenza relativo al picco.

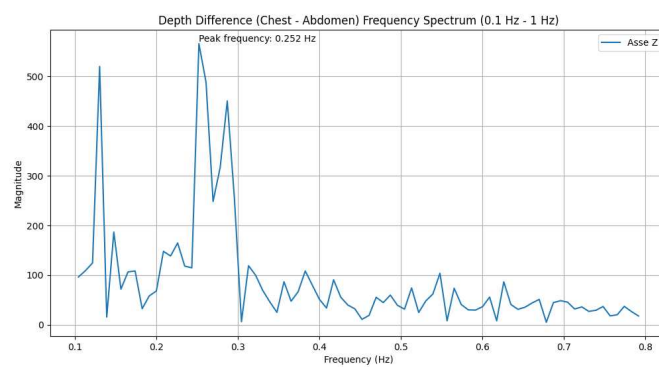


Figura 3.29: Grafico nel tempo relativo alla differenza

# Capitolo 4

## Risultati e discussione

Nella seguente sezione sono riportati i dati relativi a tutte le acquisizioni fatte sia in condizioni statiche che dinamiche corredati da alcuni grafici esemplificativi. Verranno poi discussi i risultati ottenuti con l'obiettivo di ricavare le performance per ogni algoritmo e determinare quindi quello che ha riscontrato una maggiore precisione nei vari esperimenti.

### 4.1 Condizioni statiche

Per quanto riguarda le condizioni statiche abbiamo confrontato i quattro algoritmi principali sviluppati per questo tipo di scenario e trattati nella sezione 3.5.

#### 4.1.1 Soggetti seduti

Abbiamo acquisito i dati di 3 soggetti seduti su una sedia e immobili, per ogni soggetto abbiamo fatto delle acquisizioni a differenti distanze dalla fotocamera rispettivamente 1,2 e 3 metri i risultati ottenuti per il soggetto 1 sono riportati nella tabella 4.1 mentre quelli del soggetto 2 nella tabella 4.2 e infine quelli del soggetto 3 nella tabella 4.3.

	1m	2m	3m
algoritmo 1(Hz)	0.209	0.209	0.209
algoritmo 2(Hz)	0.209	0.418	0.209
algoritmo 3(Hz)	0.209	0.209	0.209
algoritmo 4(Hz)	0.209	0.200	0.209
frequenza EDR(Hz)	0.208	0.207	0.215
frequenza asse Z(Hz)	0.207	0.207	0.215
bpm	96	89	85

Tabella 4.1: Risultati acquisizioni soggetto 1 seduto

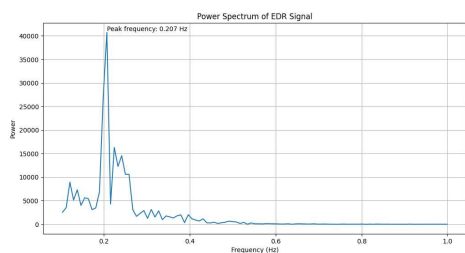
	1m	2m	3m
algoritmo 1(Hz)	0.148	0.130	0.217
algoritmo 2(Hz)	0.148	0.130	0.217
algoritmo 3(Hz)	0.148	0.130	0.217
algoritmo 4(Hz)	0.148	0.130	0.217
frequenza EDR(Hz)	0.146	0.138	0.216
frequenza asse Z(Hz)	0.146	0.138	0.216
bpm	73	73	91

Tabella 4.2: Risultati acquisizioni soggetto 2 seduto

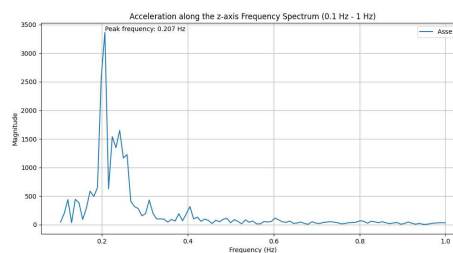
	1m	2m	3m
algoritmo 1(Hz)	0.261	0.191	0.183
algoritmo 2(Hz)	0.244	0.191	0.183
algoritmo 3(Hz)	0.261	0.191	0.183
algoritmo 4(Hz)	0.261	0.191	0.183
frequenza EDR(Hz)	0.260	0.190	0.182
frequenza asse Z(Hz)	0.259	0.190	0.182
bpm	66	56	60

Tabella 4.3: Risultati acquisizioni soggetto 3 seduto

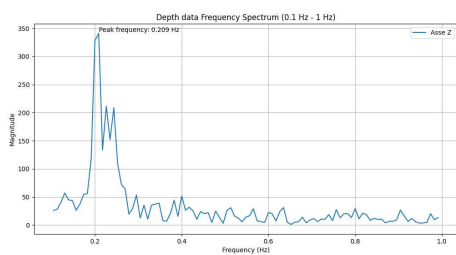
Qui di seguito sono riportati alcuni grafici che evidenziano come l'algoritmo 2, ovvero SuperficiePuntiDown, sia quello con il maggior livello di rumore. Questo fenomeno è principalmente attribuibile al fatto che i soggetti, durante tutte le acquisizioni, indossavano una maglietta. Tuttavia una maglietta che non aderisce perfettamente alla zona addominale può risultare problematica per la rilevazione della profondità, in quanto può muoversi a causa di correnti d'aria o altri movimenti.



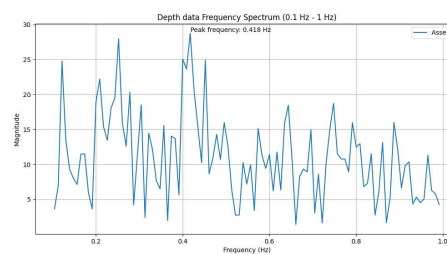
(a) EDR



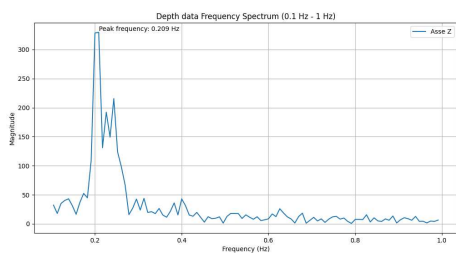
(b) Asse Z



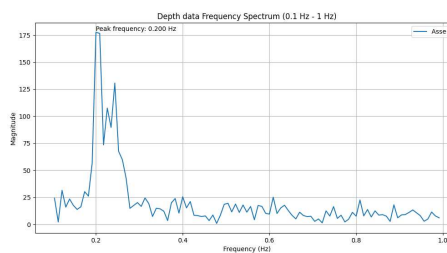
(c) Algoritmo 1



(d) Algoritmo 2

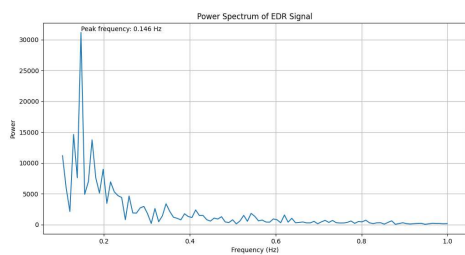


(e) Algoritmo 3

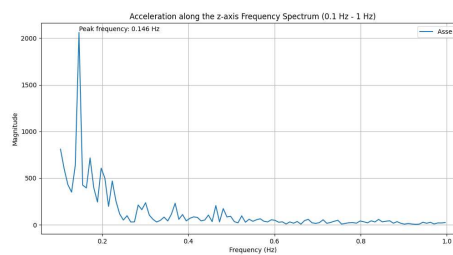


(f) Algoritmo 4

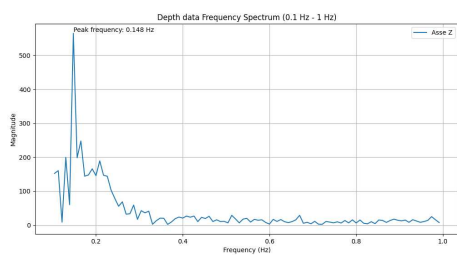
Figura 4.1: Grafici frequenze relative al soggetto 1 a 2 metri



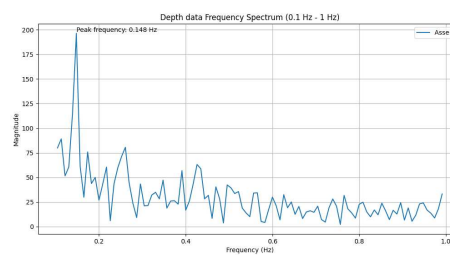
(a) EDR



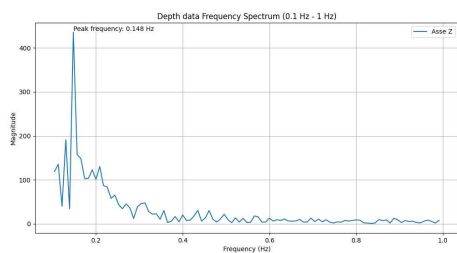
(b) Asse Z



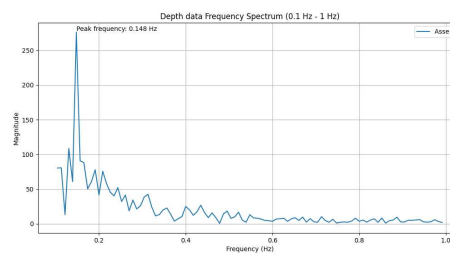
(c) Algoritmo 1



(d) Algoritmo 2



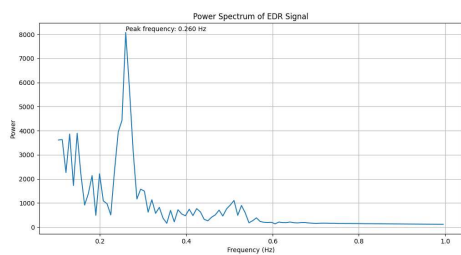
(e) Algoritmo 3



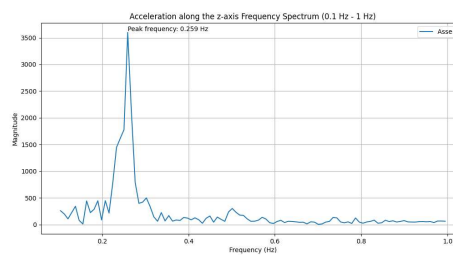
(f) Algoritmo 4

Figura 4.2: Grafici frequenze relative al soggetto 2 ad 1 metro

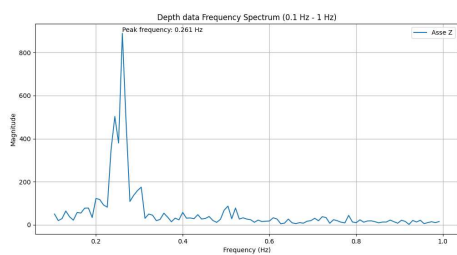




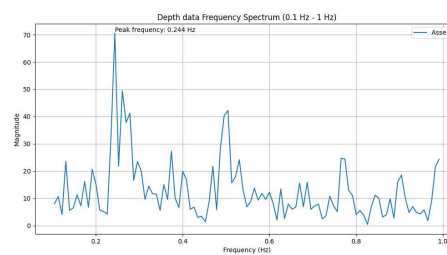
(a) EDR



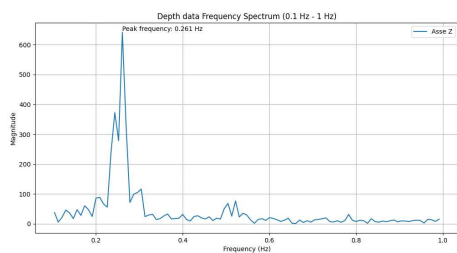
(b) Asse Z



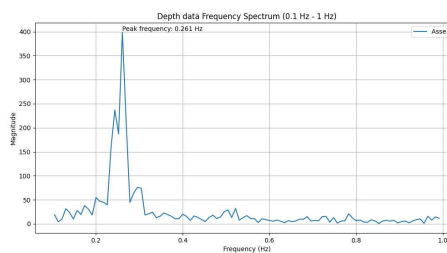
(c) Algoritmo 1



(d) Algoritmo 2



(e) Algoritmo 3



(f) Algoritmo 4

Figura 4.3: Grafici frequenze relative al soggetto 3 ad 1 metro

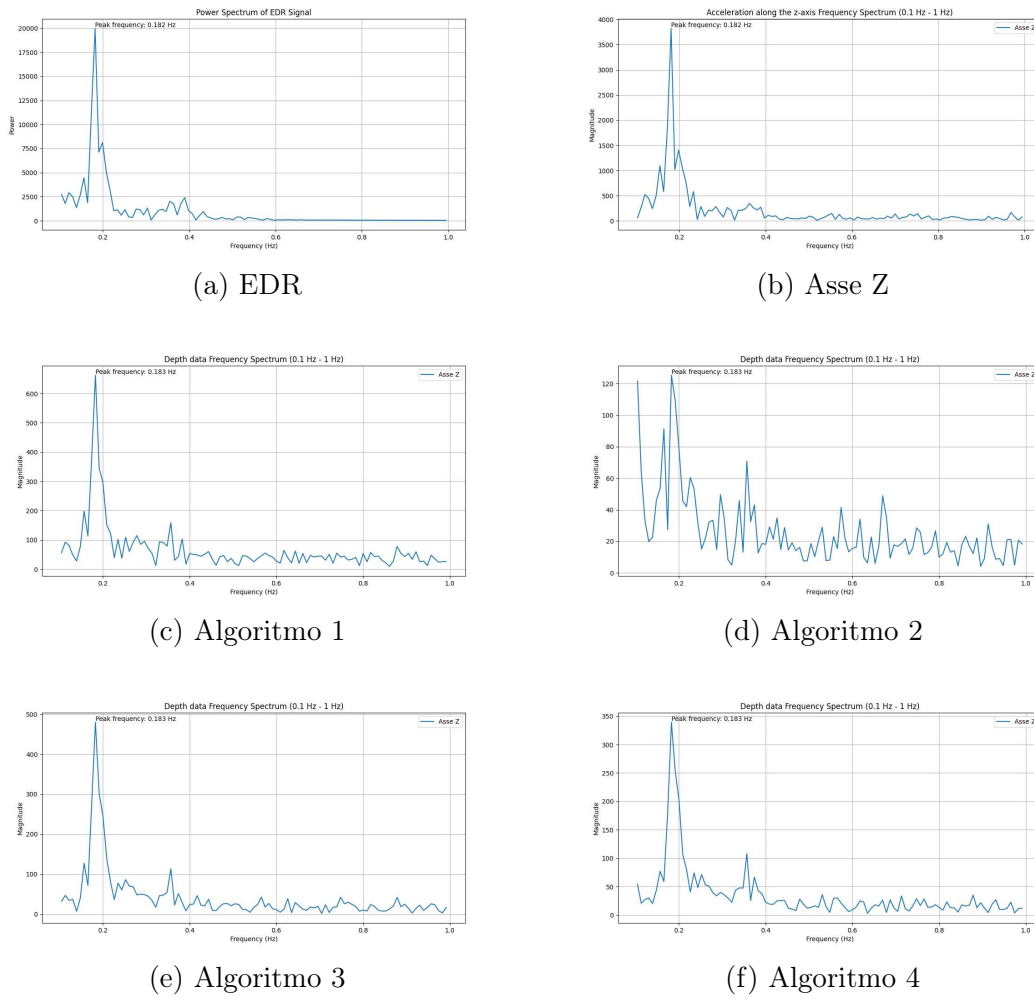


Figura 4.4: Grafici frequenze relative al soggetto 3 a 3 metri

Per ottenere le performance degli algoritmi per le diverse distanze abbiamo calcolato il MAE (Mean Absolute Error) come:

$$\text{MAE}(\text{Algoritmo } X, Y_m) = \frac{1}{n} \sum_{i=1}^n |\text{RRc}_{X,i} - \text{RRt}_i| \quad (4.1)$$

dove:

- $n = 3$  è il numero totale di soggetti,
- $\text{RRc}_{X,i}$  è il valore calcolato della frequenza respiratoria per l'algoritmo  $X$  il soggetto  $i$ , a  $Y_m$  di distanza,
- $\text{RRt}_i$  è il valore "reale" della frequenza respiratoria ottenuta dal segnale EDR per il soggetto  $i$ , a distanza  $Y_m$ .

I valori di MAE sono riportati nelle tabelle seguenti 4.4 e 4.5.

	1m	2m	3m
algoritmo 1(Hz)	0.00133	0.00366	0.00267
algoritmo 2(Hz)	0.00633	0.07333	0.00267
algoritmo 3(Hz)	0.00133	0.00366	0.00267
algoritmo 4(Hz)	0.00133	0.00533	0.00267

Tabella 4.4: MAE in Hz relativo ad ogni algoritmo e distanza

	1m	2m	3m
algoritmo 1(RPM)	0.0798	0.2196	0.1602
algoritmo 2(RPM)	0.3798	4.3998	0.1602
algoritmo 3(RPM)	0.0798	0.2196	0.1602
algoritmo 4(RPM)	0.0798	0.3198	0.1602

Tabella 4.5: MAE in RPM relativo ad ogni algoritmo e distanza

Dai dati elencati dalle tabelle sopra possiamo affermare che l'algoritmo 1 e 3 sono quelli in cui abbiamo riscontrato una maggiore accuratezza. Per quanto riguarda la distanza abbiamo notato una maggiore accuratezza alla distanza di un metro anche se gli algoritmi non sono stati soggetti molto alla variazione della distanza.

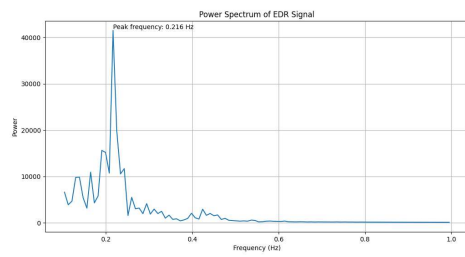
### 4.1.2 Soggetti ruotati

Abbiamo acquisito i dati da 5 soggetti ruotati di un angolo circa di 45 gradi rispetto all'asse della telecamera i risultati sono stati ottenuti utilizzando gli stessi algoritmi del caso precedente. Nella tabella 4.6 sono riportati i dati relativi ai vari soggetti che hanno preso parte alla misurazione.

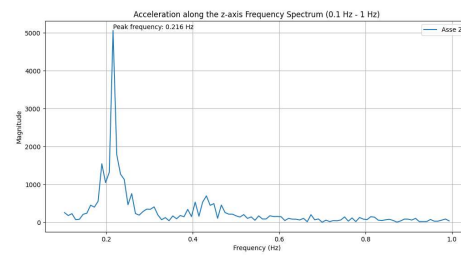
	Soggetto 1	Soggetto 2	Soggetto 3	Soggetto 4	Soggetto 5
Algoritmo 1(Hz)	0.217	0.157	0.261	0.244	0.131
Algoritmo 2(Hz)	0.217	0.157	0.261	0.200	0.131
Algoritmo 3(Hz)	0.217	0.157	0.261	0.113	0.131
Algoritmo 4(Hz)	0.217	0.157	0.270	0.139	0.104
frequenza EDR(Hz)	0.216	0.156	0.260	0.294	0.129
frequenza asse Z(Hz)	0.216	0.156	0.259	0.294	0.138
bpm	100	86	79	59	109

Tabella 4.6: Risultati acquisizioni soggetti ruotati

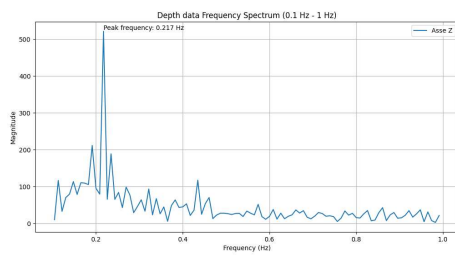
Di seguito i grafici relativi ad ogni soggetto.



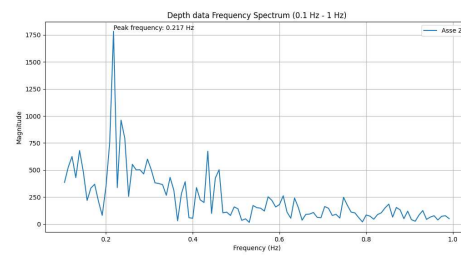
(a) EDR



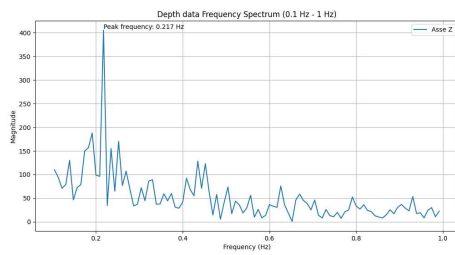
(b) Asse Z



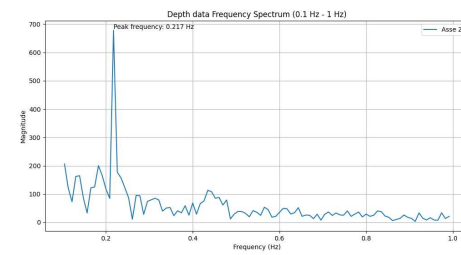
(c) Algoritmo 1



(d) Algoritmo 2

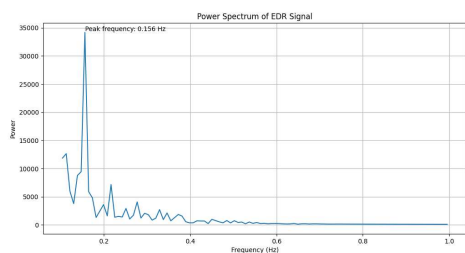


(e) Algoritmo 3

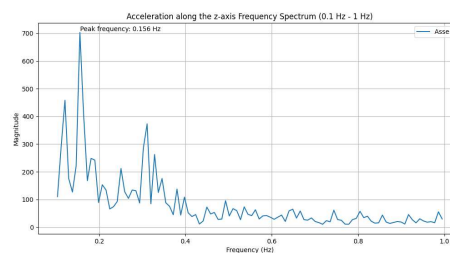


(f) Algoritmo 4

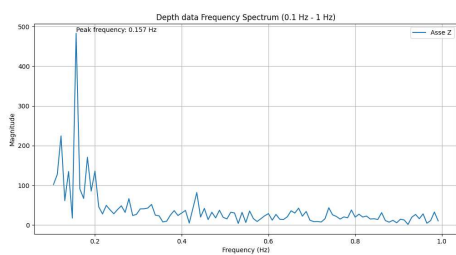
Figura 4.5: Grafici frequenze relative al soggetto 1



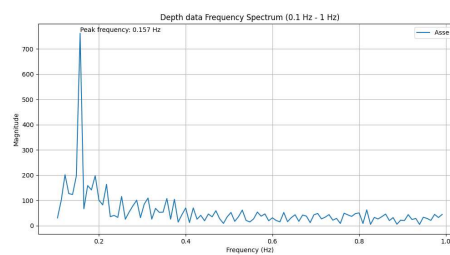
(a) EDR



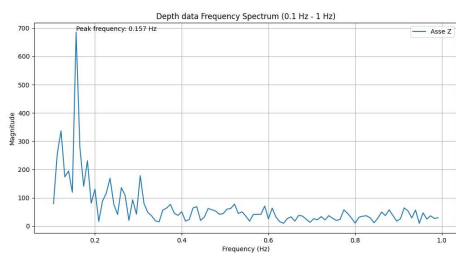
(b) Asse Z



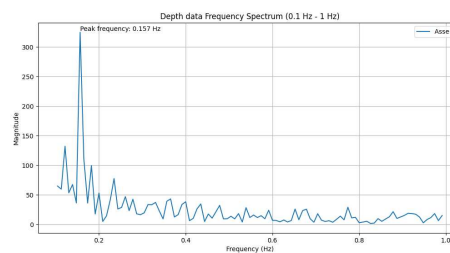
(c) Algoritmo 1



(d) Algoritmo 2

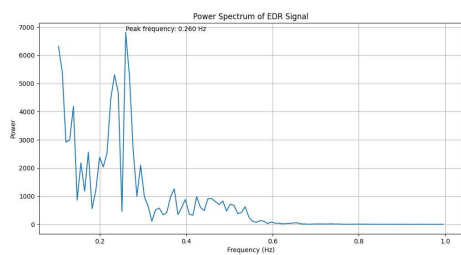


(e) Algoritmo 3

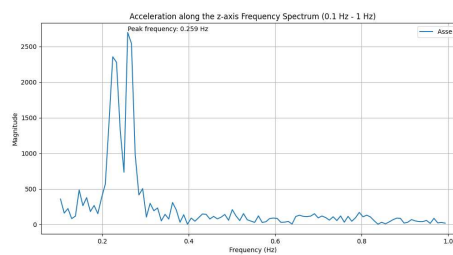


(f) Algoritmo 4

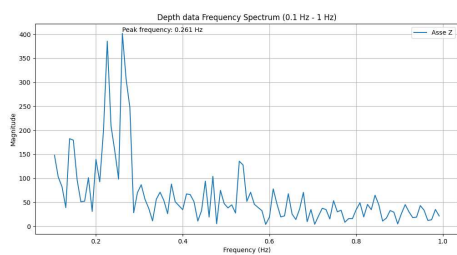
Figura 4.6: Grafici frequenze relative al soggetto 2



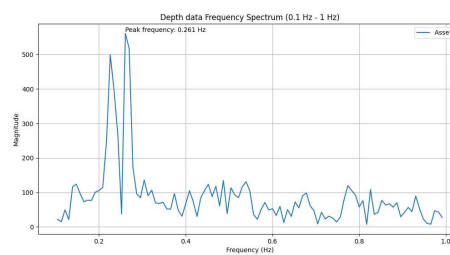
(a) EDR



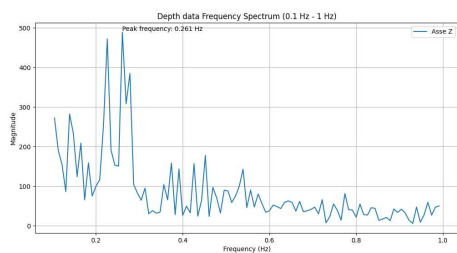
(b) Asse Z



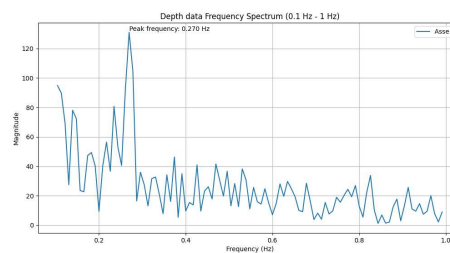
(c) Algoritmo 1



(d) Algoritmo 2

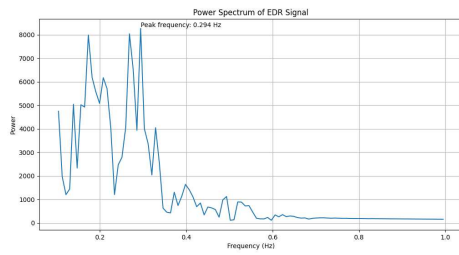


(e) Algoritmo 3

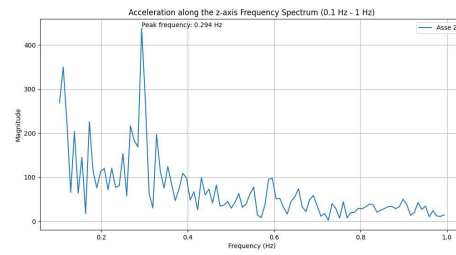


(f) Algoritmo 4

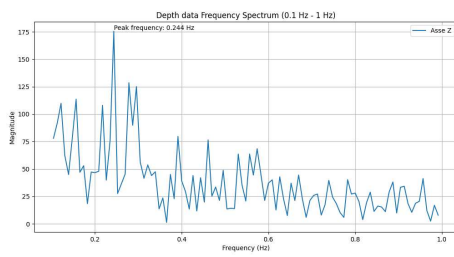
Figura 4.7: Grafici frequenze relative al soggetto 3



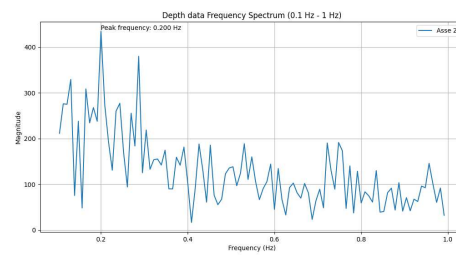
(a) EDR



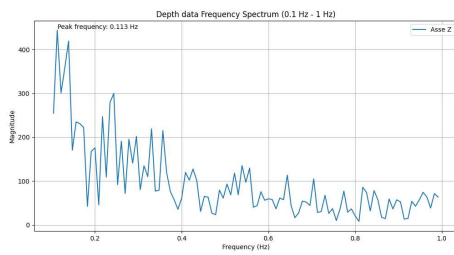
(b) Asse Z



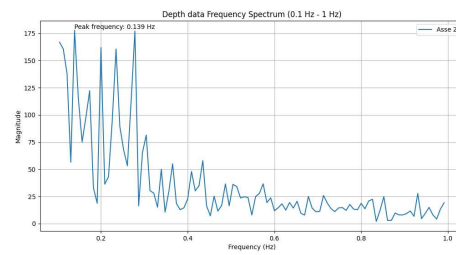
(c) Algoritmo 1



(d) Algoritmo 2



(e) Algoritmo 3



(f) Algoritmo 4

Figura 4.8: Grafici frequenze relative al soggetto 4

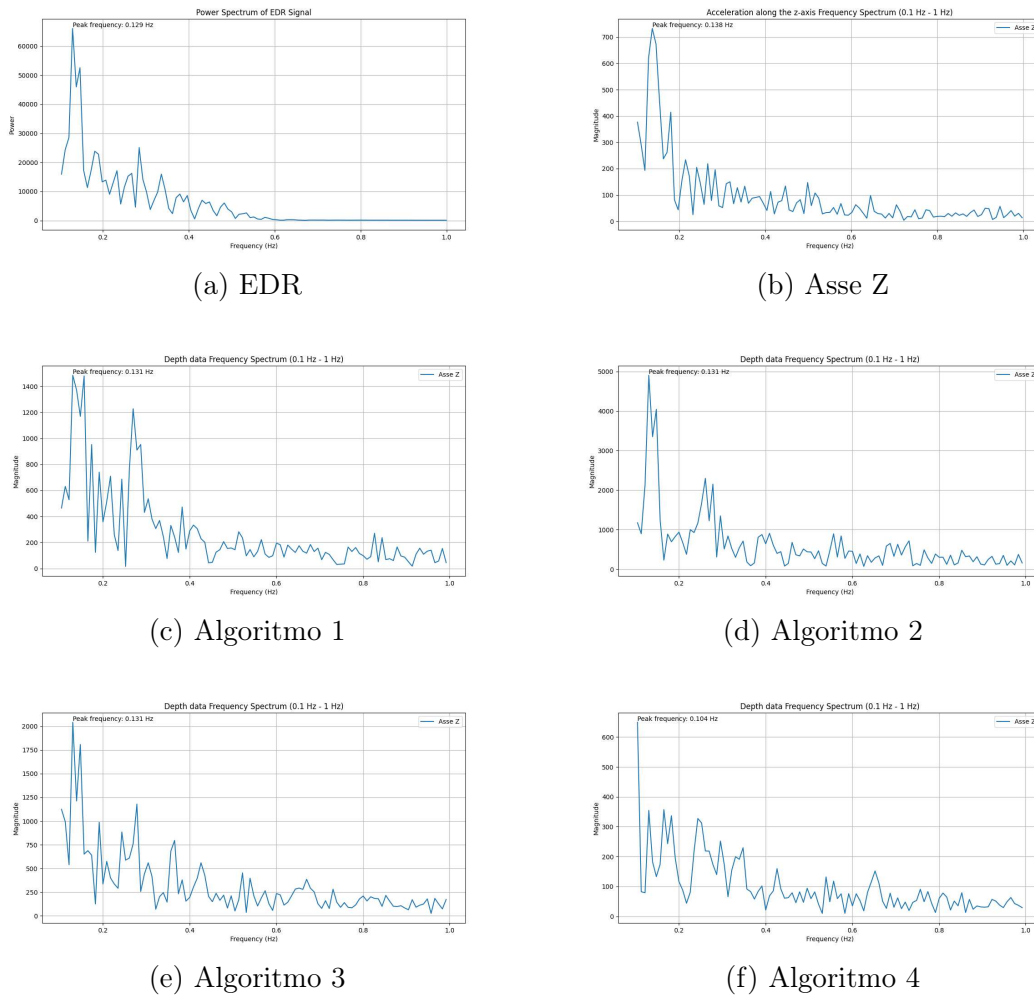


Figura 4.9: Grafici frequenze relative al soggetto 5

Le tabelle 4.7 e 4.8 mostrano per ogni soggetto i valori di frequenza e RPM della differenza tra il respiration rate calcolato per ogni algoritmo rispetto alla respiration rate "reale" ottenuta dal segnale EDR.

Algoritmo	Soggetto 1	Soggetto 2	Soggetto 3	Soggetto 4	Soggetto 5
Algoritmo 1(Hz)	0.001	0.001	0.001	0.050	0.002
Algoritmo 2(Hz)	0.001	0.001	0.001	0.094	0.002
Algoritmo 3(Hz)	0.001	0.001	0.001	0.181	0.002
Algoritmo 4(Hz)	0.001	0.001	0.010	0.155	0.025

Tabella 4.7: Errore in Hz



Algoritmo	Soggetto 1	Soggetto 2	Soggetto 3	Soggetto 4	Soggetto 5
Algoritmo 1(RPM)	0.06	0.06	0.06	3.00	0.12
Algoritmo 2(RPM)	0.06	0.06	0.06	5.64	0.12
Algoritmo 3(RPM)	0.06	0.06	0.06	10.86	0.12
Algoritmo 4(RPM)	0.06	0.06	0.60	9.30	1.50

Tabella 4.8: Errore in RPM

La formula per il MAE in questo caso è data da:

$$\text{MAE}(\text{Algoritmo X}) = \frac{1}{n} \sum_{i=1}^n |\text{RRc}_{X,i} - \text{RRt}_i| \quad (4.2)$$

dove:

- $n = 5$  è il numero totale di soggetti,
- $\text{RRc}_{X,i}$  è il valore calcolato della frequenza respiratoria per l'algoritmo  $X$  e il soggetto  $i$ ,
- $\text{RRt}_i$  è il valore "reale" della frequenza respiratoria ottenuta dal segnale EDR per il soggetto  $i$ .

I risultati ottenuti dal MAE per ogni algoritmo sono riportati nella tabella 4.9.

Algoritmo	MAE (Hz)	MAE (RPM)
Algoritmo 1	0.0110	0.660
Algoritmo 2	0.0198	1.188
Algoritmo 3	0.0372	2.232
Algoritmo 4	0.0384	2.304

Tabella 4.9: MAE (Mean Absolute Error) in Hz e RPM

Dalle tabelle presentate sopra possiamo affermare che per questo tipo di condizione statica l'algoritmo che generalmente ha performato meglio è stato il primo ovvero quello relativo al SingoloPunto.

## 4.2 Condizioni dinamiche

Per quanto riguarda le acquisizioni in condizioni dinamiche abbiamo deciso comunque di considerare nei risultati gli algoritmi utilizzati per le condizioni statiche in modo da poterne misurare l'efficienza in scenari diversi da quelli per cui erano stati concepiti e valutarne le performance. Grande attenzione verrà posta soprattutto verso gli algoritmi sviluppati per le condizioni dinamiche trattati nella sezione 3.6.

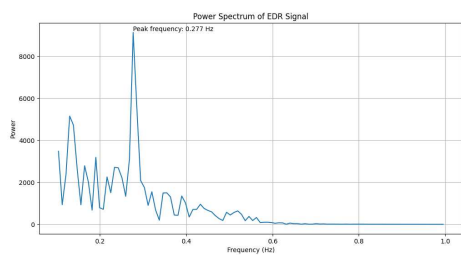
### 4.2.1 Movimento frontale

L'unico soggetto che abbiamo preso in esperimento ovvero il soggetto 3 è stato fatto muovere avanti e indietro in maniera periodica frontalmente alla telecamera. Per questo tipo di movimento è stata prevista una singola acquisizione. I risultati dell'acquisizione sono riportati nella tabella 4.10. In molti dei grafici presentati di seguito

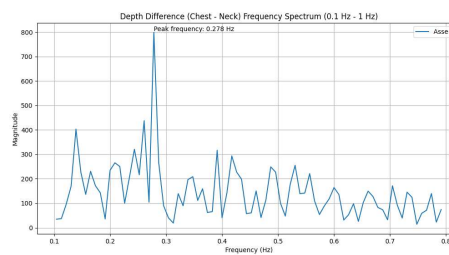
	Acquisizione
YOLO1(Hz)	0.104
YOLO2(Hz)	0.104
YOLO3(Hz)	0.104
YOLO4(Hz)	0.104
YOLO1 - Collo(Hz)	0.278
YOLO2 - Collo(Hz)	0.139
YOLO3 - Collo(Hz)	0.278
YOLO4 - Collo(Hz)	0.139
YOLO1 - Addome(Hz)	0.278
YOLO3 - Addome(Hz)	0.278
YOLO4 - Addome(Hz)	0.278
YOLO1 - Spalle(Hz)	0.200
YOLO2 - Spalle(Hz)	0.278
YOLO3 - Spalle(Hz)	0.200
YOLO4 - Spalle(Hz)	0.278
frequenza EDR(Hz)	0.277
frequenza asse Z(Hz)	0.277
bpm	72

Tabella 4.10: Risultati acquisizione movimenti frontali per il soggetto 3

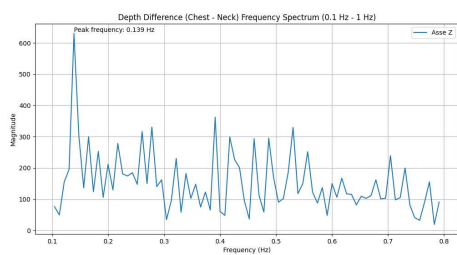
è possibile notare due picchi di frequenza principali: uno a 2.77 Hz, corrispondente alla frequenza respiratoria, e l'altro vicino a 0.13 Hz, probabilmente legato alla frequenza del movimento del soggetto. Quando quest'ultimo picco è più grande, invece di rilevare la frequenza respiratoria, si rileva la frequenza del movimento, riducendo così l'accuratezza della misurazione. Questo fenomeno è visibile anche nei grafici relativi ad altre condizioni dinamiche.



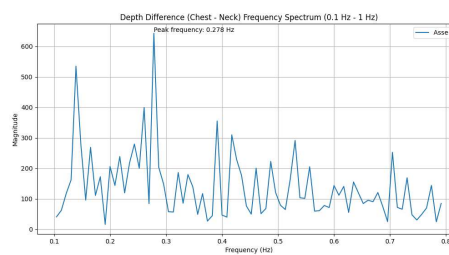
(a) EDR



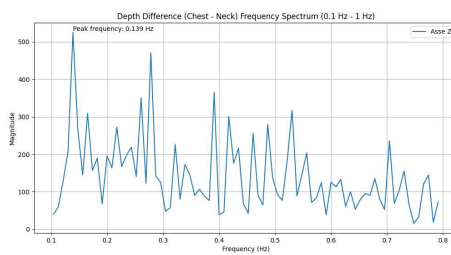
(b) YOLO1 - Collo



(c) YOLO2 - Collo



(d) YOLO3 - Collo



(e) YOLO4 - Collo

Figura 4.10: Grafici differenza con collo

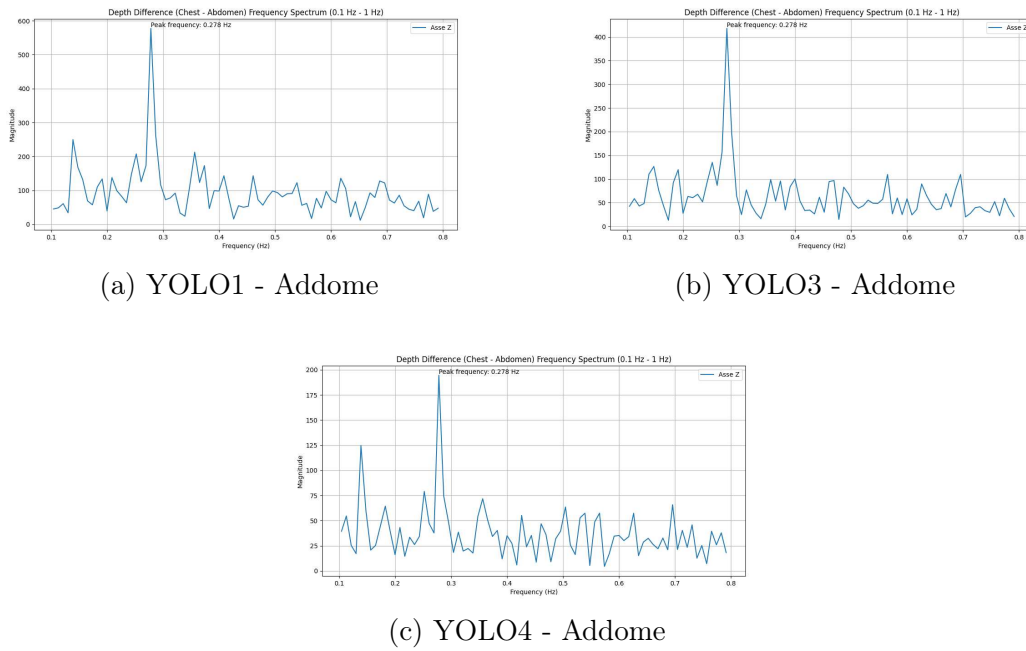


Figura 4.11: Grafici differenza addome

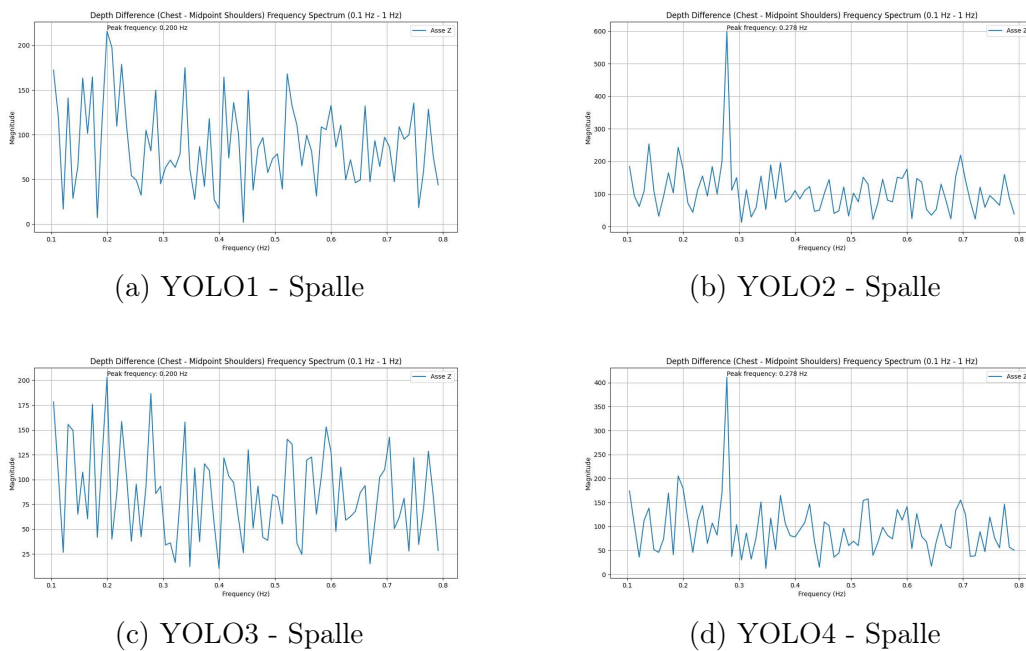


Figura 4.12: Grafici differenza spalle

Nelle tabelle 4.11 e 4.12 è possibile osservare gli errori in frequenza e in RPM dei vari algoritmi rispetto sempre al valore considerato come "reale" della frequenza del segnale EDR.

Algoritmo	Differenza in Hz
YOLO1	0.173
YOLO2	0.173
YOLO3	0.173
YOLO4	0.173
YOLO1 - Collo	0.001
YOLO2 - Collo	0.138
YOLO3 - Collo	0.001
YOLO4 - Collo	0.138
YOLO1 - Addome	0.001
YOLO3 - Addome	0.001
YOLO4 - Addome	0.001
YOLO1 - Spalle	0.077
YOLO2 - Spalle	0.001
YOLO3 - Spalle	0.077
YOLO4 - Spalle	0.001

Tabella 4.11: Errore in Frequenza (Hz)

Algoritmo	Differenza in RPM
YOLO1	10.38
YOLO2	10.38
YOLO3	10.38
YOLO4	10.38
YOLO1 - Collo	0.06
YOLO2 - Collo	8.28
YOLO3 - Collo	0.06
YOLO4 - Collo	8.28
YOLO1 - Addome	0.06
YOLO3 - Addome	0.06
YOLO4 - Addome	0.06
YOLO1 - Spalle	4.62
YOLO2 - Spalle	0.06
YOLO3 - Spalle	4.62
YOLO4 - Spalle	0.06

Tabella 4.12: Errore in RPM

In questo caso specifico avendo un solo soggetto di studio non ha senso calcolare il MAE. Dai risultati ottenuti possiamo affermare che gli algoritmi con una maggiore accuratezza sono stati tutti quelli relativi alla differenza con il segnale addominale e in parte quelli in cui sottraiamo il segnale del collo e delle spalle. Come ci si

aspettava gli algoritmi ideati nel caso statico sono stati quelli in cui si è riscontrato un errore maggiore.

### 4.2.2 Rotazione del torso

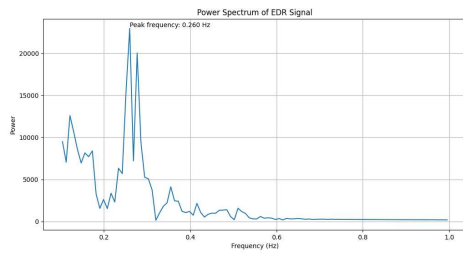
Abbiamo acquisito i dati da quattro soggetti che seppur rimanendo statici compivano rotazioni del torso. Per il soggetto 4 sono state eseguite due acquisizioni per arricchire con ulteriori dati il dataset.

Nella tabella 4.13 vengono mostrati tutti i risultati ottenuti.

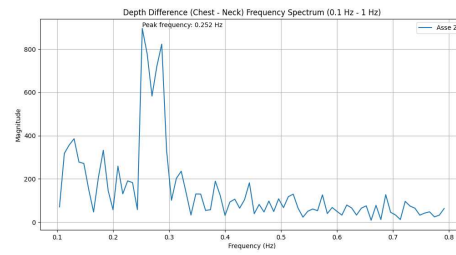
	Soggetto 1	Soggetto 2	Soggetto 3	Soggetto 4	Soggetto4(2)
YOLO1(Hz)	0.113	0.131	0.322	0.322	0.139
YOLO2(Hz)	0.139	0.131	0.130	0.322	0.139
YOLO3(Hz)	0.139	0.131	0.331	0.322	0.139
YOLO4(Hz)	0.131	0.131	0.165	0.322	0.139
YOLO1 - Collo(Hz)	0.244	0.252	0.157	0.635	0.139
YOLO2 - Collo(Hz)	0.139	0.278	0.487	0.322	0.139
YOLO3 - Collo(Hz)	0.244	0.252	0.165	0.635	0.122
YOLO4 - Collo(Hz)	0.139	0.287	0.165	0.322	0.139
YOLO1 - Addome(Hz)	0.244	0.252	0.165	0.635	0.148
YOLO3 - Addome(Hz)	0.244	0.261	0.331	0.331	0.139
YOLO4 - Addome(Hz)	0.244	0.131	0.165	0.635	0.139
YOLO1 - Spalle(Hz)	0.157	0.131	0.165	0.322	0.139
YOLO2 - Spalle(Hz)	0.244	0.261	0.322	0.635	0.139
YOLO3 - Spalle(Hz)	0.139	0.139	0.165	0.635	0.122
YOLO4 - Spalle(Hz)	0.252	0.131	0.165	0.322	0.139
frequenza EDR(Hz)	0.250	0.260	0.338	0.138	0.146
frequenza asse Z(Hz)	0.249	0.259	0.328	0.138	0.146
bpm	106	92	84	68	74

Tabella 4.13: Risultati acquisizioni rotazione torso

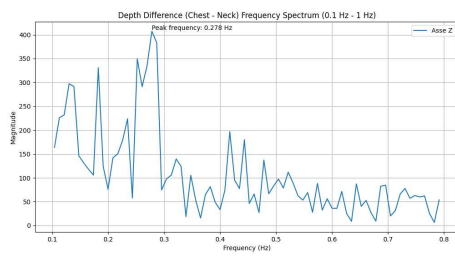
I grafici seguenti sono relativi alle acquisizioni svolte sul soggetto 2. Anche in questo caso si può notare la presenza di due picchi di frequenza principali uno che rappresenta la frequenza respiratoria l'altro la frequenza di rotazione.



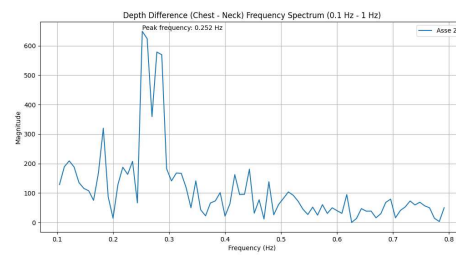
(a) EDR



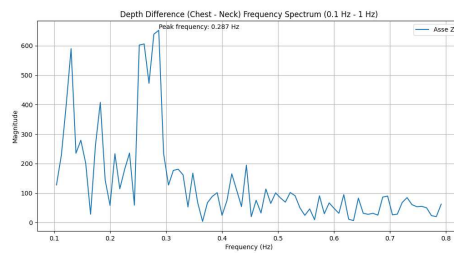
(b) YOLO1 - Collo



(c) YOLO2 - Collo

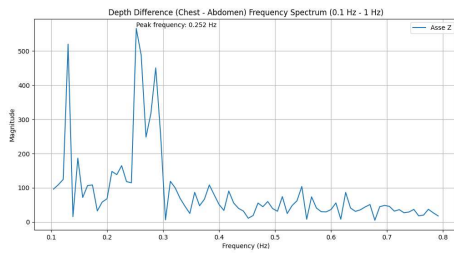


(d) YOLO3 - Collo

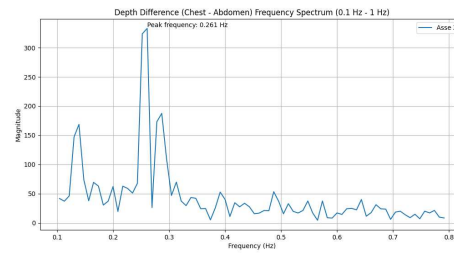


(e) YOLO4 - Collo

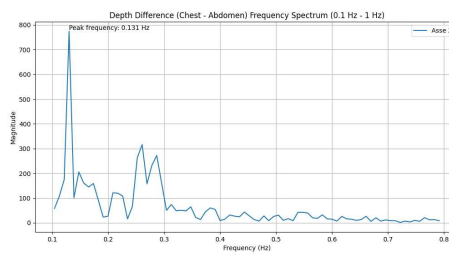
Figura 4.13: Grafici differenza con collo



(a) YOLO1 - Addome

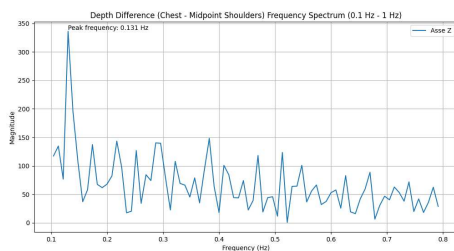


(b) YOLO3 - Addome

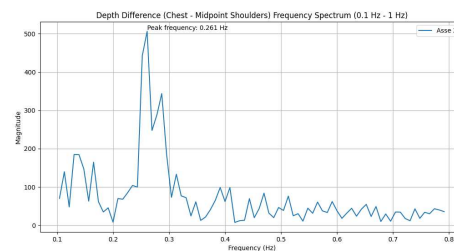


(c) YOLO4 - Addome

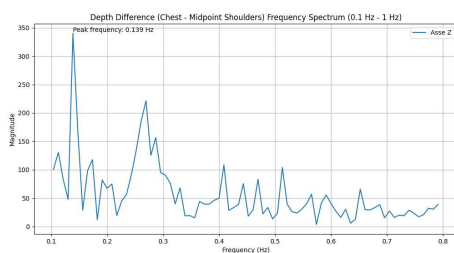
Figura 4.14: Grafici differenza addome



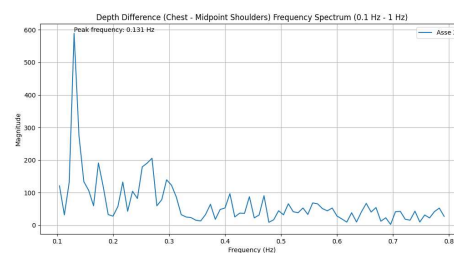
(a) YOLO1 - Spalle



(b) YOLO2 - Spalle



(c) YOLO3 - Spalle



(d) YOLO4 - Spalle

Figura 4.15: Grafici differenza spalle

Nelle tabelle 4.14 e 4.15 sono riportati gli errori in Hz e RPM che abbiamo riscontrato nei vari soggetti per ogni algoritmo in relazione al valore in frequenza del segnale EDR.



Algoritmo	Soggetto 1	Soggetto 2	Soggetto 3	Soggetto 4	Soggetto4(2)
YOLO1	0.137	0.129	0.016	0.184	0.007
YOLO2	0.111	0.129	0.208	0.184	0.007
YOLO3	0.111	0.129	0.007	0.184	0.007
YOLO4	0.119	0.129	0.173	0.184	0.007
YOLO1 - Collo	0.006	0.008	0.181	0.497	0.007
YOLO2 - Collo	0.111	0.018	0.149	0.184	0.007
YOLO3 - Collo	0.006	0.008	0.173	0.497	0.024
YOLO4 - Collo	0.111	0.027	0.173	0.184	0.007
YOLO1 - Addome	0.006	0.008	0.173	0.497	0.002
YOLO3 - Addome	0.006	0.001	0.007	0.193	0.007
YOLO4 - Addome	0.006	0.129	0.173	0.497	0.007
YOLO1 - Spalle	0.093	0.129	0.173	0.184	0.007
YOLO2 - Spalle	0.006	0.001	0.016	0.497	0.007
YOLO3 - Spalle	0.111	0.121	0.173	0.497	0.024
YOLO4 - Spalle	0.002	0.129	0.173	0.184	0.007

Tabella 4.14: Errore in Hz

Algoritmo	Soggetto 1	Soggetto 2	Soggetto 3	Soggetto 4	Soggetto4(2)
YOLO1	8.22	7.74	0.96	11.04	0.42
YOLO2	6.66	7.74	12.48	11.04	0.42
YOLO3	6.66	7.74	0.42	11.04	0.42
YOLO4	7.14	7.74	10.38	11.04	0.42
YOLO1 - Collo	0.36	0.48	10.86	29.82	0.42
YOLO2 - Collo	6.66	1.08	8.94	11.04	0.42
YOLO3 - Collo	0.36	0.48	10.38	29.82	1.44
YOLO4 - Collo	6.66	1.62	10.38	11.04	0.42
YOLO1 - Addome	0.36	0.48	10.38	29.82	0.12
YOLO3 - Addome	0.36	0.06	0.42	11.58	0.42
YOLO4 - Addome	0.36	7.74	10.38	29.82	0.42
YOLO1 - Spalle	5.58	7.74	10.38	11.04	0.42
YOLO2 - Spalle	0.36	0.06	0.96	29.82	0.42
YOLO3 - Spalle	6.66	7.26	10.38	29.82	1.44
YOLO4 - Spalle	0.12	7.74	10.38	11.04	0.42

Tabella 4.15: Errore in RPM

Per calcolare il MAE in questo caso utilizziamo la formula 4.2. Di seguito i valori di MAE ottenuti dal calcolo.

Algoritmo	MAE (Hz)	MAE (RPM)
YOLO1	0.0946	5.676
YOLO2	0.1278	7.668
YOLO3	0.0876	5.256
YOLO4	0.1224	7.344
YOLO1 - Collo	0.1398	8.388
YOLO2 - Collo	0.0938	5.628
YOLO3 - Collo	0.1416	8.496
YOLO4 - Collo4	0.1004	6.024
YOLO1 - Addome	0.1372	8.232
YOLO3 - Addome	0.0428	2.568
YOLO4 - Addome	0.1624	9.744
YOLO1 - Spalle	0.1172	7.032
YOLO2 - Spalle	0.1054	6.324
YOLO3 - Spalle	0.1852	11.112
YOLO4 - Spalle	0.0990	5.940

Tabella 4.16: MAE (Mean Absolute Error) in Hz e RPM

Dai valori ottenuti possiamo evincere che generalmente c'è stato un peggioramento nell'accuratezza degli algoritmi riscontrando valori più alti di MAE rispetto a quelli ottenuti nel caso statico. Questo è giustificato dal fatto che il movimento del soggetto va ad impattare notevolmente sul movimento dovuto alla respirazione. In questo tipo di contesto dinamico in particolare ha riscontrato una maggiore accuratezza l'algoritmo della differenza tra l'algoritmo YOLO3 (SuperficiePuntiUp) e l'addome anche se gli algoritmi base nel caso statico almeno per quanto riguarda SingoloPunto e SuperficiepuntiUp hanno performato inaspettatamente bene in questo ambito. Mentre per quanto riguarda gli algoritmi di differenza con spalle e collo hanno avuto accuratezze minori principalmente per il fatto che con la rotazione del torso hanno avuto più difficoltà ad individuare i keypoints del collo e spalle nel momento in cui il busto si trovava girato.

### 4.2.3 Movimento casuale

Infine abbiamo proceduto a fare delle acquisizioni per ogni soggetto che si spostava in maniera casuale all'interno del campo visivo della telecamera. Sono riportati i risultati per ogni individuo nelle tabelle seguenti. Per il soggetto 1 abbiamo eseguito 3 prove:

	Acquisizione 1	Acquisizione 2	Acquisizione 3
YOLO1(Hz)	0.104	0.130	0.104
YOLO2(Hz)	0.104	0.130	0.104
YOLO3(Hz)	0.104	0.130	0.104
YOLO4(Hz)	0.104	0.130	0.104
YOLO1 - Collo(Hz)	0.122	0.339	0.183
YOLO2 - Collo(Hz)	0.148	0.104	0.104
YOLO3 - Collo(Hz)	0.200	0.339	0.183
YOLO4 - Collo(Hz)	0.200	0.348	0.104
YOLO1 - Addome(Hz)	0.104	0.113	0.104
YOLO3 - Addome(Hz)	0.104	0.113	0.104
YOLO4 - Addome(Hz)	0.183	0.113	0.104
YOLO1 - Spalle(Hz)	0.113	0.104	0.296
YOLO2 - Spalle(Hz)	0.104	0.113	0.104
YOLO3 - Spalle(Hz)	0.287	0.113	0.104
YOLO4 - Spalle(Hz)	0.287	0.113	0.104
frequenza EDR(Hz)	0.215	0.344	0.301
frequenza asse Z(Hz)	0.215	0.344	0.302
bpm	88	104	104

Tabella 4.17: Risultati acquisizioni movimenti casuali per il soggetto 1

Per il soggetto 2 abbiamo fatto 2 acquisizioni:

	Acquisizione 1	Acquisizione 2
YOLO1(Hz)	0.113	0.104
YOLO2(Hz)	0.113	0.104
YOLO3(Hz)	0.113	0.104
YOLO4(Hz)	0.113	0.104
YOLO1 - Collo(Hz)	0.157	0.261
YOLO2 - Collo(Hz)	0.157	0.104
YOLO3 - Collo(Hz)	0.157	0.261
YOLO4 - Collo(Hz)	0.157	0.104
YOLO1 - Addome(Hz)	0.122	0.252
YOLO3 - Addome(Hz)	0.226	0.270
YOLO4 - Addome(Hz)	0.122	0.174
YOLO1 - Spalle(Hz)	0.165	0.487
YOLO2 - Spalle(Hz)	0.130	0.104
YOLO3 - Spalle(Hz)	0.157	0.157
YOLO4 - Spalle(Hz)	0.165	0.131
frequenza EDR(Hz)	0.225	0.313
frequenza asse Z(Hz)	0.225	0.313
bpm	82	81

Tabella 4.18: Risultati acquisizioni movimenti casuali per il soggetto 2

Per il soggetto 3 abbiamo eseguito 3 acquisizioni:

	Acquisizione 1	Acquisizione 2	Acquisizione 3
YOLO1(Hz)	0.113	0.104	0.122
YOLO2(Hz)	0.113	0.104	0.122
YOLO3(Hz)	0.113	0.104	0.122
YOLO4(Hz)	0.113	0.104	0.122
YOLO1 - Collo(Hz)	0.287	0.296	0.270
YOLO2 - Collo(Hz)	0.287	0.296	0.244
YOLO3 - Collo(Hz)	0.287	0.296	0.270
YOLO4 - Collo(Hz)	0.287	0.296	0.122
YOLO1 - Addome(Hz)	0.296	0.174	0.139
YOLO3 - Addome(Hz)	0.296	0.296	0.444
YOLO4 - Addome(Hz)	0.287	0.139	0.139
YOLO1 - Spalle(Hz)	0.113	0.139	0.261
YOLO2 - Spalle(Hz)	0.287	0.139	0.444
YOLO3 - Spalle(Hz)	0.287	0.139	0.122
YOLO4 - Spalle(Hz)	0.287	0.296	0.183
frequenza EDR(Hz)	0.286	0.301	0.268
frequenza asse Z(Hz)	0.285	0.301	0.268
bpm	74	68	69

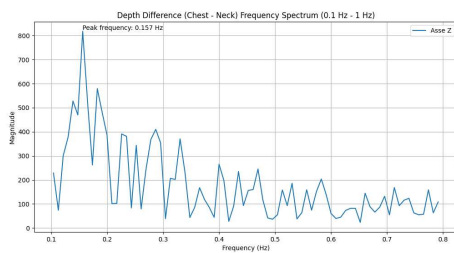
Tabella 4.19: Risultati acquisizioni movimenti casuali per il soggetto 3

Infine per il soggetto 4 abbiamo eseguito solo 1 acquisizione:

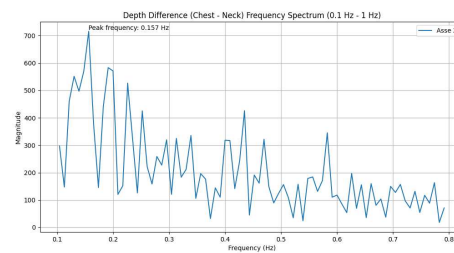
	Acquisizione 1
YOLO1(Hz)	0.104
YOLO2(Hz)	0.104
YOLO3(Hz)	0.104
YOLO4(Hz)	0.104
YOLO1 - Collo(Hz)	0.339
YOLO2 - Collo(Hz)	0.122
YOLO3 - Collo(Hz)	0.113
YOLO4 - Collo(Hz)	0.374
YOLO1 - Addome(Hz)	0.165
YOLO3 - Addome(Hz)	0.122
YOLO4 - Addome(Hz)	0.165
YOLO1 - Spalle(Hz)	0.148
YOLO2 - Spalle(Hz)	0.435
YOLO3 - Spalle(Hz)	0.148
YOLO4 - Spalle(Hz)	0.104
frequenza EDR(Hz)	0.344
frequenza asse Z(Hz)	0.344
bpm	65

Tabella 4.20: Risultati acquisizione movimenti casuali per il soggetto 4

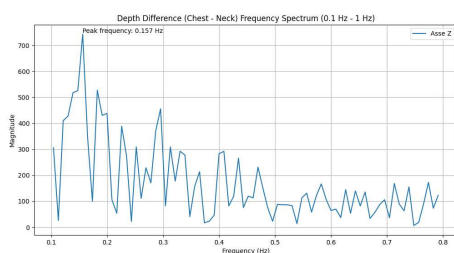
Di seguito sono mostrati alcuni grafici relativi alla prima acquisizione del soggetto 2.



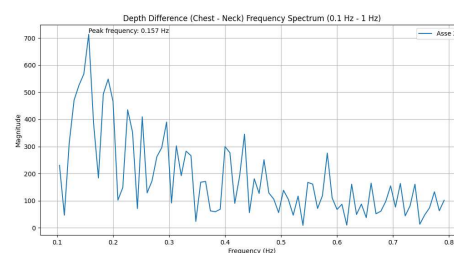
(a) YOLO1 - Collo



(b) YOLO2 - Collo

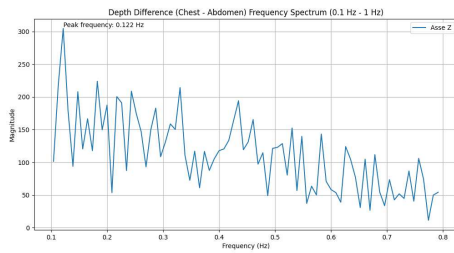


(c) YOLO3 - Collo

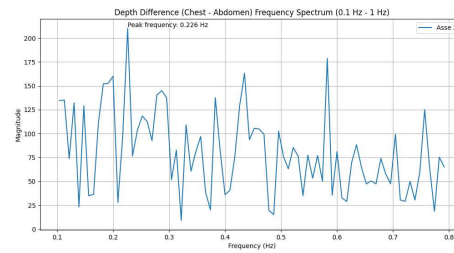


(d) YOLO4 - Collo

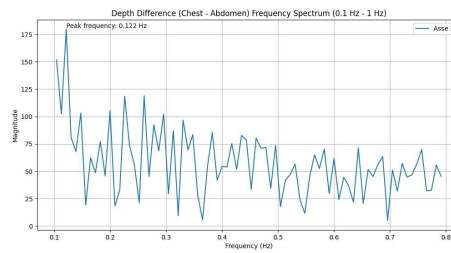
Figura 4.16: Grafici differenza con collo



(a) YOLO1 - Addome

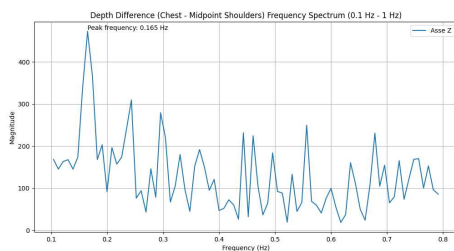


(b) YOLO3 - Addome

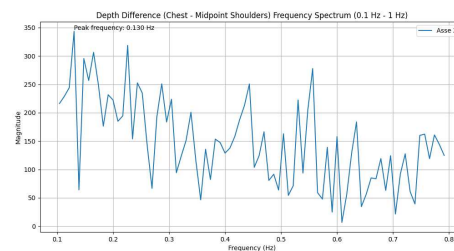


(c) YOLO4 - Addome

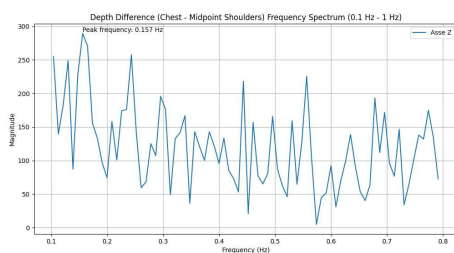
Figura 4.17: Grafici differenza addome



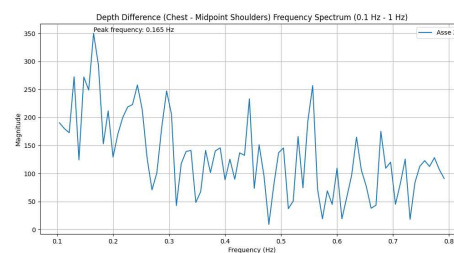
(a) YOLO1 - Spalle



(b) YOLO2 - Spalle



(c) YOLO3 - Spalle



(d) YOLO4 - Spalle

Figura 4.18: Grafici differenza spalle

Fondendo le acquisizioni svolte dai diversi soggetti ci ricaviamo due tabelle dove riportiamo la differenza della frequenza respiratoria in Hz e RPM tra i valori ricavati dai vari algoritmi e quelli ottenuti dal segnale EDR.

Algoritmi	S1(1)	S1(2)	S1(3)	S2(1)	S2(2)	S3(1)	S3(2)	S3(3)	S4(1)
YOLO1	0.111	0.214	0.197	0.112	0.209	0.173	0.197	0.146	0.240
YOLO2	0.111	0.214	0.197	0.112	0.209	0.173	0.197	0.146	0.240
YOLO3	0.111	0.214	0.197	0.112	0.209	0.173	0.197	0.146	0.240
YOLO4	0.111	0.214	0.197	0.112	0.209	0.173	0.197	0.146	0.240
YOLO1 - Collo	0.093	0.005	0.118	0.068	0.052	0.001	0.005	0.002	0.005
YOLO2 - Collo	0.067	0.240	0.197	0.068	0.209	0.001	0.005	0.024	0.222
YOLO3 - Collo	0.015	0.005	0.118	0.068	0.052	0.001	0.005	0.002	0.231
YOLO4 - Collo	0.015	0.004	0.197	0.068	0.209	0.001	0.005	0.146	0.030
YOLO1 - Addome	0.111	0.231	0.197	0.103	0.061	0.010	0.127	0.129	0.179
YOLO3 - Addome	0.111	0.231	0.197	0.001	0.043	0.010	0.005	0.176	0.222
YOLO4 - Addome	0.032	0.231	0.197	0.103	0.139	0.001	0.162	0.129	0.179
YOLO1 - Spalle	0.102	0.240	0.005	0.060	0.174	0.173	0.162	0.007	0.196
YOLO2 - Spalle	0.111	0.231	0.197	0.095	0.209	0.001	0.162	0.176	0.091
YOLO3 - Spalle	0.072	0.231	0.197	0.068	0.156	0.001	0.162	0.146	0.196
YOLO4 - Spalle	0.072	0.231	0.197	0.060	0.182	0.001	0.005	0.085	0.240

Tabella 4.21: Errore in Hz

Algoritmi	S1(1)	S1(2)	S1(3)	S2(1)	S2(2)	S3(1)	S3(2)	S3(3)	S4(1)
YOLO1	6.66	12.84	11.82	6.72	12.54	10.38	11.82	8.76	14.40
YOLO2	6.66	12.84	11.82	6.72	12.54	10.38	11.82	8.76	14.40
YOLO3	6.66	12.84	11.82	6.72	12.54	10.38	11.82	8.76	14.40
YOLO4	6.66	12.84	11.82	6.72	12.54	10.38	11.82	8.76	14.40
YOLO1 - Collo	5.58	0.30	7.08	4.08	3.12	0.06	0.30	0.12	0.30
YOLO2 - Collo	4.02	14.40	11.82	4.08	12.54	0.06	0.30	1.44	13.32
YOLO3 - Collo	0.90	0.30	7.08	4.08	3.12	0.06	0.30	0.12	13.86
YOLO4 - Collo	0.90	0.24	11.82	4.08	12.54	0.06	0.30	8.76	1.80
YOLO1 - Addome	6.66	13.86	11.82	6.18	3.66	0.60	7.62	7.74	10.74
YOLO3 - Addome	6.66	13.86	11.82	0.06	2.58	0.60	0.30	10.56	13.32
YOLO4 - Addome	1.92	13.86	11.82	6.18	8.34	0.06	9.72	7.74	10.74
YOLO1 - Spalle	6.12	14.40	0.30	3.60	10.44	10.38	9.72	0.42	11.76
YOLO2 - Spalle	6.66	13.86	11.82	5.70	12.54	0.06	9.72	10.56	5.46
YOLO3 - Spalle	4.32	13.86	11.82	4.08	9.36	0.06	9.72	8.76	11.76
YOLO4 - Spalle	4.32	13.86	11.82	3.60	10.92	0.06	0.30	5.10	14.40

Tabella 4.22: Errore in RPM

Sfruttiamo ancora una volta la formula 4.2 per calcolare il MAE tra tutte le acquisizioni fatte e riportiamo i risultati nella tabella sottostante.



Algoritmo	MAE (Hz)	MAE (RPM)
YOLO1	0.1777	10.662
YOLO2	0.1777	10.662
YOLO3	0.1777	10.662
YOLO4	0.1777	10.662
YOLO1 - Collo	0.0388	2.328
YOLO2 - Collo	0.1148	6.888
YOLO3 - Collo	0.0552	3.312
YOLO4 - Collo	0.0750	4.500
YOLO1 - Addome	0.1276	7.656
YOLO3 - Addome	0.1106	6.636
YOLO4 - Addome	0.1303	7.818
YOLO1 - Spalle	0.1243	7.458
YOLO2 - Spalle	0.1414	8.484
YOLO3 - Spalle	0.1366	8.196
YOLO4 - Spalle	0.1192	7.152

Tabella 4.23: MAE (Mean Absolute Error) in Hz e RPM

Dai risultati di MAE nella tabella 4.23 si può osservare una migliore performance per quanto riguarda gli algoritmi di differenza con il collo ad eccezione di quello con il segnale addominale (Collo - YOLO2).

# Capitolo 5

## Sviluppi futuri

Alla luce dei risultati ottenuti e delle considerazioni effettuate, sono stati identificati diversi sviluppi futuri e possibili miglioramenti per il nostro progetto:

- **Aumento del Dataset:** Una delle priorità iniziali dovrebbe essere l'espansione del dataset, aumentando il numero di soggetti di studio. È fondamentale includere acquisizioni in scenari diversificati e in varie condizioni di illuminazione. Questo ci permetterà di valutare l'efficienza e le performance dei vari algoritmi in contesti più ampi e diversificati, avvicinandoci il più possibile alle situazioni reali per le quali sono stati progettati.
- **Miglioramento degli algoritmi:** Viste le varie inefficienze degli algoritmi proposti nel caso dinamico l'ideale sarebbe riuscire a realizzare un algoritmo robusto, non influenzato dal movimento del soggetto e che possa dare risultati corretti in diverse situazioni di movimento.
- **Filtrare le frequenze:** Poiché in alcuni casi abbiamo ottenuto dei picchi di frequenza a frequenze inferiori a 0.1Hz cosa improbabile visto il range di frequenze di respirazione normali, sarebbe utile applicare una maschera che rimuova tutte le frequenze al di sotto di 0.1Hz.
- **Analisi delle prestazioni in relazione ai requisiti di tempo:** I risultati ottenuti in questo progetto sono frutto di acquisizioni e processamenti che avvengono in un tempo successivo, si potrebbero valutare le prestazioni degli algoritmi in relazione ai requisiti di tempo nel caso in cui volessimo ottenere dei risultati in tempo reale. Questo potrebbe implicare la valutazione della latenza di processamento o la loro accuratezza nel caso in cui volessimo ottenere il processamento real-time.

# Capitolo 6

## Conclusione

La frequenza respiratoria è un parametro che troppo spesso viene trascurato anche se la sua valutazione è cruciale per la diagnosi e la gestione delle malattie. Negli ultimi anni, sono stati intensificati gli studi per consentire un monitoraggio non invasivo di questo parametro, con un ampio impiego di telecamere di profondità come nel nostro caso. Questo tipo di approccio permette di ridurre il contatto fisico con il paziente aumentando il comfort e monitorando la respirazione in una vasta gamma di contesti clinici.

L'obiettivo principale di questo studio è stato sviluppare diversi algoritmi che ci permettano di rilevare la frequenza respiratoria di un soggetto attraverso l'uso di una telecamera RGB-D. Per valutarne l'efficienza sono state prese in considerazione un'ampia varietà di condizioni sia statiche a partire dal soggetto immobile su una sedia a quello in piedi rotato che dinamiche tra cui movimenti frontali, casuali e rotazione del torso.

Dai risultati ottenuti nelle varie acquisizioni emerge un'ottima accuratezza degli algoritmi ai diversi contesti in cui abbiamo operato anche se con alcune problematiche nel caso dinamico principalmente causate dall'influenza del movimento del soggetto sul segnale respiratorio.

In conclusione, il nostro lavoro costituisce una buona base di partenza per ulteriori sviluppi nell'ambito del monitoraggio della frequenza respiratoria utilizzando tecniche di rilevamento a distanza, contribuendo così a migliorare la qualità della diagnosi e il trattamento delle malattie respiratorie.

# Bibliografia

- [1] Anthony P Addison, Paul S Addison, Philip Smit, Dominique Jacquiel, and Ulf R Borg. Noncontact respiratory monitoring using depth sensing cameras: A review of current literature. *Sensors*, 21(4):1135, 2021.
- [2] Paul S Addison, André Antunes, Dean Montgomery, Philip Smit, and Ulf R Borg. Robust non-contact monitoring of respiratory rate using a depth camera. *Journal of Clinical Monitoring and Computing*, 37(4):1003–1010, 2023.
- [3] Hirooki Aoki and Noriaki Fujita. Non-contact respiration measurement by using rgb-d camera. In *2022 Joint 12th International Conference on Soft Computing and Intelligent Systems and 23rd International Symposium on Advanced Intelligent Systems (SCIS&ISIS)*, pages 1–4. IEEE, 2022.
- [4] Gary G Berntson, John T Cacioppo, and Karen S Quigley. Respiratory sinus arrhythmia: Autonomic origins, physiological mechanisms, and psychophysiological implications. *Psychophysiology*, 30(2):183–196, 1993.
- [5] Patrícia Bota, Rafael Silva, Carlos Carreiras, Ana Fred, and Hugo Plácido da Silva. Biosppy: A python toolbox for physiological signal processing. *SoftwareX*, 26:101712, 2024.
- [6] Peter H Charlton, David A Birrenkott, Thomas Bonnici, Marco AF Pimentel, Alistair EW Johnson, Jorge Alastruey, Lionel Tarassenko, et al. Breathing rate estimation from the electrocardiogram and photoplethysmogram: A review. *IEEE Reviews in Biomedical Engineering*, 9:38–51, 2016.
- [7] Michelle A Cretikos, Rinaldo Bellomo, Ken Hillman, Jack Chen, Simon Finfer, and Arthas Flabouris. Respiratory rate: the neglected vital sign. *Medical Journal of Australia*, 188(11):657–659, 2008.
- [8] Roberto De Fazio, Marco Stabile, Massimo De Vittorio, Ramiro Velázquez, and Paolo Visconti. An overview of wearable piezoresistive and inertial sensors for respiration rate monitoring. *Electronics*, 10(17):2178, 2021.
- [9] SciPy Developers. Scipy documentation, 2024. Accessed: 29 June 2024.

- [10] Hulya Dogan and Ramazan Ozgur Dogan. A comprehensive review of computer-based techniques for r-peaks/qrs complex detection in ecg signal. *Archives of Computational Methods in Engineering*, 30(6):3703–3721, 2023.
- [11] John F Fieselmann, Michael S Hendryx, Charles M Helms, and Douglas S Wakefield. Respiratory rate predicts cardiopulmonary arrest for internal medicine inpatients. *Journal of general internal medicine*, 8:354–360, 1993.
- [12] Python Software Foundation. `asyncio` — asynchronous i/o, 2024. Accessed: 29 June 2024.
- [13] Rahel Gilgen-Ammann, Theresa Schweizer, and Thomas Wyss. Rr interval signal quality of a heart rate monitor and an ecg holter at rest and during exercise. *European journal of applied physiology*, 119(7):1525–1532, 2019.
- [14] DR Goldhill, AF McNarry, G Mandersloot, and A McGinley. A physiologically-based early warning score for ward patients: the association between score and outcome. *Anaesthesia*, 60(6):547–553, 2005.
- [15] Shan He, Zixiong Han, Cristóvão Iglesias, Varun Mehta, and Miodrag Bolic. A real-time respiration monitoring and classification system using a depth camera and radars. *Frontiers in Physiology*, 13:799621, 2022.
- [16] Katrina Hinde, Graham White, and Nicola Armstrong. Wearable devices suitable for monitoring twenty four hour heart rate variability in military populations. *Sensors*, 21(4):1061, 2021.
- [17] Emily Lam, Danielle DuPlessis, Michael Hutchison, Shannon Scratch, and Elaine Biddiss. Validation of the polar h10 accelerometer in a sports-based environment. *Engineering Proceedings*, 27(1):71, 2022.
- [18] Dominique Makowski, Tam Pham, Zen J Lau, Jan C Brammer, François Lepinasse, Hung Pham, Christopher Schölzel, and SH Annabel Chen. Neurokit2: A python toolbox for neurophysiological signal processing. *Behavior research methods*, pages 1–8, 2021.
- [19] Carlo Massaroni, Andrea Nicolò, Daniela Lo Presti, Massimo Sacchetti, Sergio Silvestri, and Emiliano Schena. Contact-based methods for measuring respiratory rate. *Sensors*, 19(4):908, 2019.
- [20] Polar. *Polar H10 Heart Rate Sensor: Manuale utente*, n.d. Accessed: 29 June 2024.
- [21] PyPI. `bleak`, 2024. Accessed: 29 June 2024.
- [22] PyPI. `pyrealsense2`, 2024. Accessed: 29 June 2024.

- 
- [23] Intel RealSense. Intel realsense d455 depth camera, 2024. Accessed on 25 giugno 2024, indirizzo: <https://www.intelrealsense.com/depth-camera-d455/>.
- [24] Tomáš Skála, Marek Vícha, Martin Rada, Jan Vácha, Jakub Flašík, and Miloš Táborský. Feasibility of evaluation of polar h10 chest-belt ecg in patients with a broad range of heart conditions. *Cor Vasa*, 64:411–422, 2022.
- [25] Juan Terven, Diana-Margarita Córdova-Esparza, and Julio-Alejandro Romero-González. A comprehensive review of yolo architectures in computer vision: From yolov1 to yolov8 and yolo-nas. *Machine Learning and Knowledge Extraction*, 5(4):1680–1716, 2023.
- [26] Yunlu Wang, Menghan Hu, Yuwen Zhou, Qingli Li, Nan Yao, Guangtao Zhai, Xiao-Ping Zhang, and Xiaokang Yang. Unobtrusive and automatic classification of multiple people’s abnormal respiratory patterns in real time using deep neural network and depth camera. *IEEE Internet of Things Journal*, 7(9):8559–8571, 2020.
- [27] George Yuan, Nicole A Drost, and R Andrew McIvor. Respiratory rate and breathing pattern. *McMaster Univ. Med. J*, 10(1):23–25, 2013.

# Ringraziamenti

Desidero innanzitutto ringraziare il mio Relatore, prof. Ennio Gambi, per la sua disponibilità e per avermi dato l'opportunità di svolgere questo progetto di tirocinio. Un grazie di cuore va anche al dott. Nocera Antonio, per averci seguito con dedizione durante tutte le fasi del percorso, fornendoci preziosi consigli e materiale su cui lavorare.

Un ringraziamento speciale va ai miei genitori, Cristina e Cesare, che hanno reso possibile tutto questo supportandomi sia dal punto di vista psicologico che finanziario. Avete saputo incoraggiarmi nei momenti difficili e gioire insieme a me per ogni successo conseguito. Grazie di cuore per tutti i sacrifici che avete fatto per permettermi di costruire un futuro migliore. So di poter contare sempre su di voi.

Ringrazio anche tutti i miei amici, con i quali ho condiviso momenti indimenticabili nel corso di questi tre anni di università e non solo. Grazie per aver reso più leggeri i periodi di esami con la vostra costante presenza e per essere sempre riusciti strapparmi un sorriso.

Un pensiero speciale va a Cristina, sebbene tu sia entrata nella mia vita da poco riesci ad illuminare con il tuo sorriso ogni mia giornata. Grazie al tuo amore e ai tuoi continui incoraggiamenti, mi hai dato la forza di non arrendermi mai e di credere sempre nelle mie potenzialità rendendomi ogni giorno una persona migliore. Sono fortunato ad averti al mio fianco.

Infine, un ringraziamento insolito ma doveroso è rivolto a me stesso. Con impegno e dedizione ho raggiunto sempre i traguardi che mi sono imposto, incluso questo.

*Christian*