

ALMA MATER STUDIORUM - UNIVERSITÀ DI BOLOGNA

SCUOLA DI INGEGNERIA E ARCHITETTURA

DIPARTIMENTO di
INGEGNERIA DELL'ENERGIA ELETTRICA E DELL'INFORMAZIONE
“Guglielmo Marconi”
DEI

CORSO DI LAUREA IN INGEGNERIA DELL'AUTOMAZIONE

TESI DI LAUREA
in
FOUNDATIONS OF INDUSTRIAL ROBOTICS

**Classificazione di gesti per il controllo di mani robotiche basato sulla
trasformata Wavelet discreta di segnali EMG**

CANDIDATO
Enrico Guerra

RELATORE:
Chiar.mo Prof. Claudio Melchiorri

CORRELATORE
Dott. Roberto Meattini

Anno Accademico 2017/2018

Sessione II

Alla mia famiglia: Michele,
Lorenza, Caterina e Oscar

Indice

Introduzione	1
1 Segnali EMG	2
1.1 Unità motoria	2
1.2 Generazione segnale e misurazione	2
2 Trasformata Wavelet	5
2.1 Segnali non stazionari	5
2.2 Short Time Fourier Transform (STFT)	5
2.3 Wavelet Transform	6
2.4 Analisi multi risoluzione (MRA)	9
2.5 Discrete Wavelet Transform (DWT)	11
3 Sistema di classificazione	13
3.1 Macchina a vettori di supporto (SVM)	14
3.2 Sistema di classificazione off-line e validazione	15
3.3 Sistema di classificazione on-line	18
3.4 Validazione on-line	20
3.5 Risultati	21
4 Conclusioni	24
Bibliografia	25

Introduzione

Sin dagli albori della robotica, l'interfaccia uomo-robot è sempre stata in continua evoluzione. All'inizio i manipolatori erano visti come semplici "estensioni" del braccio umano ed i primi che furono sviluppati si occupavano del trattamento di materiali pericolosi, in cui il lavoro di controllo era demandato interamente all'uomo. Sono poi stati aggiunti alcuni elementi che aumentassero l'autonomia dei robot, ma la configurazione di interazione uomo-robot è sempre stata quella classica di tipo *master-slave*. Questa consiste nell'utilizzo da parte dell'uomo di un dispositivo di interfaccia (*master*) che controlli un robot remoto (*slave*). Negli ultimi anni, però, sempre più studiosi hanno iniziato a sviluppare sistemi per una maggiore cooperazione e interazione con l'uomo. Questo ha portato alla progettazione di sistemi robotici sempre più ispirati all'essere umano, in modo tale da replicare l'abilità e flessibilità umana nel compiere lavori motori.

Nonostante siano stati fatti recenti sviluppi riguardo l'interfaccia cervello-computer, i sistemi basati sui segnali elettromiografici (EMG) sono tra i più popolari per il controllo di mani robotiche. Oltretutto è stato dimostrato che l'intenzione motoria umana è facilmente riconoscibile attraverso questi segnali. Sebbene queste tecniche non abbiano ancora raggiunto una maturità industriale, questo forte utilizzo riflette la comodità e qualità di impiego delle misurazioni EMG. Alcuni studi si sono focalizzati soprattutto sul controllo della presa (Meattini et al., 2018) e sul sollevamento di oggetti (Meattini et al., 2017), ma è stato anche cercato di utilizzare il segnale EMG per classificare e riconoscere i gesti eseguiti con la mano (Englehart et al., 2001), . A questo scopo, negli anni recenti ha trovato notevole impiego l'utilizzo della trasformata Wavelet discreta (DWT, *Discrete wavelet transform*) per l'estrazione di feature dal segnale che potessero aiutare l'efficacia della classificazione (Phinyomark et al., 2012), rispetto ad altri metodi come il calcolo del valore efficace (RMS). In particolare, si è ottenuto un buon risultato nell'applicazione della mDWT (*marginal Discrete Wavelet Transform*) combinata con classificatori del tipo SVM (*Support Vector Machine*) (Lucas et al., 2008).

In questo lavoro di tesi si è utilizzato quest'ultimo metodo per l'estrazione delle feature e la classificazione dei gesti e si è testato il suo funzionamento prima off-line con dei dati acquisiti in due sessioni, quindi on-line facendo un test con un utente a cui venivano misurati i segnali EMG, mentre eseguiva i gesti, ed una mano robotica che cercasse di replicarli in tempo reale. Nella prima parte dell'elaborato vi sarà una spiegazione più dettagliata di che cosa sono i segnali EMG e una trattazione della trasformata wavelet e della teoria che ne sta dietro. Quindi vi sarà una più ampia parte sulla progettazione e implementazione del sistema di classificazione e sui test effettuati per la validazione del classificatore. Saranno poi presentati i risultati e infine le conclusioni.

Capitolo 1: Segnali EMG

Per elettromiografia si intende la disciplina sperimentale legata alla misura, l'analisi e l'utilizzo del segnale elettrico generato dai muscoli del corpo umano. Il suo obiettivo principale è valutare il funzionamento dei muscoli attraverso l'analisi dei potenziali elettrici da essi generati. Viene indicata comunemente con l'abbreviazione EMG.

1.1 Unità motoria

Con unità motoria (abbreviata in MU, dall'inglese *motor unit*) si intende la più piccola unità funzionale che descrive il controllo neurale della contrazione delle fibre muscolari, elementi caratteristici del tessuto muscolare. Gli elementi che la compongono sono il motoneurone α , il relativo assone e tutte le fibre muscolari da esso innervate, che quindi si contrarranno assieme al segnale di attivazione dato dall'assone. Il numero di MU varia da muscolo a muscolo.

L'attivazione della fibra muscolare avviene per mezzo di un impulso elettrico, propagato dal motoneurone α , che, arrivato alla giunzione neuromuscolare, causa l'emissione di acetilcolina nello spazio tra la zona terminale del nervo e la membrana della fibra muscolare. L'acetilcolina eccita la fibra muscolare, la fibra si depolarizza dalla giunzione e la depolarizzazione si propaga fino ai tendini.

La forza esercitata dal muscolo, nelle contrazioni volontarie, è modulata da due parametri indipendenti fra loro, che sono legati al coinvolgimento delle MU:

- Reclutamento delle MU: maggiore il numero di unità coinvolte, maggiore sarà la forza esercitata;
- Frequenza di attivazione delle MU (*firing*): maggiore è questa frequenza, maggiore sarà la forza.

1.2 Generazione segnale e misurazione

In maniera simile a come l'assone trasmette i potenziali di azione, così il tessuto muscolare conduce i potenziali elettrici. Questa attività elettrica delle fibre muscolari deriva dal reclutamento delle MU e prende il nome, in inglese, di *motor unit action potential* (MUAP). In pratica, lungo la direzione della fibra muscolare, risulta la propagazione di un'onda di depolarizzazione, che, assieme alla conseguente onda di ripolarizzazione, compone quello che viene misurato dagli elettrodi EMG.

Esistono due tipologie di elettrodi EMG: intramuscolari e superficiali. Gli elettrodi superficiali (utilizzati in questo lavoro di tesi), abbreviati in sEMG, consistono di dischi di circa 1 cm di diametro fatti in argento. La misurazione fatta da questo tipo di elettrodi è più riproducibile, ma permette di rilevare unicamente l'attività media dei muscoli più superficiali.

Considerando il modello in fig.1 è possibile capire come la propagazione dei MUAP è registrata dagli elettrodi superficiali. Due punti vengono considerati: I , dove la depolarizzazione è rappresentata nei termini di una sorgente di corrente, e $-I$, dove invece è rappresentata la ripolarizzazione. Si assume che su tutta la fibra la corrente si concentri unicamente in questi due punti, separati da una distanza d . Un elettrodo puntiforme si

trova a distanza r dalla sorgente di corrente. Il potenziale può essere espresso nella seguente maniera:

$$\phi = \frac{I}{4\pi\sigma} \frac{1}{r}$$

Dove sigma è la conduttività del mezzo. Il potenziale netto dell'elettrodo sarà, quindi:

$$\phi_{elettrodo} = \frac{I}{4\pi\sigma} \frac{1}{r_1} - \frac{I}{4\pi\sigma} \frac{1}{r_2} = \frac{I}{4\pi\sigma} \left(\frac{1}{r_1} - \frac{1}{r_2} \right)$$

Dove r_1 e r_2 rappresentano le distanze dell'elettrodo dai due punti I e $-I$. Queste distanze variano nel tempo poiché l'onda si propaga lungo la fibra muscolare, modificando l'andamento temporale. L'onda che il singolo elettrodo misurerà sarà un'onda bifasica, come mostrato in figura.

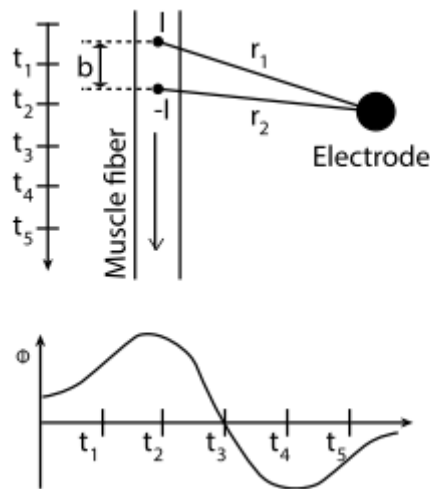


Fig. 1 Modello fibra muscolare-elettrodo, con segnale misurato.

Quest'onda sarà comunque influenzata da vari fattori di misurazione e biologici. Anche la superficie dell'elettrodo influenzerà la registrazione, dovuto al fatto che, nella realtà, ha una superficie finita e delimitata e non puntiforme. Il dato che si ricava principalmente da queste misurazioni sEMG è la differenza di potenziale fra due elettrodi posti sul sito muscolare. Quindi, il risultato che deriva dalla sottrazione di due onde bifasiche di potenziale è un'onda trifasica (fig.2).

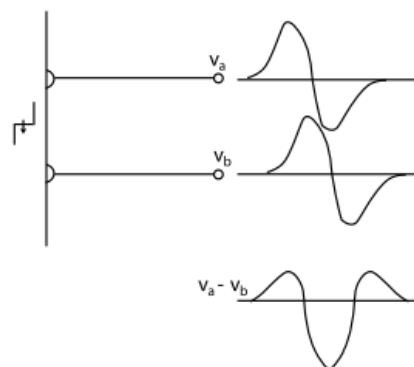


Fig. 2 Andamento del segnale misurato ai capi dei due elettrodi.

Verrà quindi misurato dalla coppia di elettrodi la sovrapposizione dei potenziali di tutte le unità motorie di un determinato muscolo.

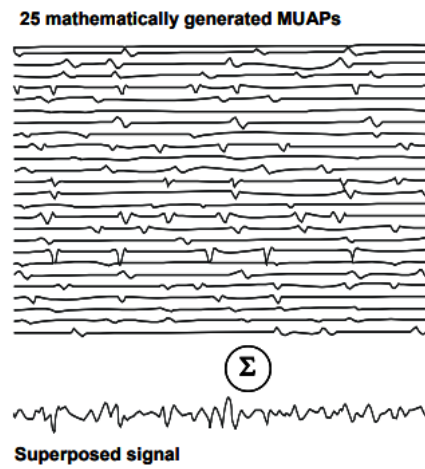


Fig. 3 Esempio simulato di segnale EMG, dato dalla sovrapposizione dei vari MUAP.

Capitolo 2: Trasformata Wavelet

2.1 Segnali non stazionari

La maggior parte dei segnali normalmente misurati e studiati sono funzioni dipendenti dal tempo e vengono rappresentati in diagrammi tempo-ampiezza del segnale. Questo tipo di rappresentazione non è sempre ottimale per lo studio del segnale e le sue applicazioni. Un'altra rappresentazione, molto più utile, è quella data dall'analisi di Fourier, in cui si studia il contenuto frequenziale (spettro delle frequenze) del segnale. Per svolgere questa analisi si utilizza la trasformata di Fourier (abbreviata FT) che riporterà una rappresentazione frequenza-ampiezza del segnale esaminato.

La trasformata di Fourier è reversibile e permette di passare dal segnale originario a quello processato (rappresentazione frequenza-ampiezza) e viceversa. Nonostante questo, nella rappresentazione frequenziale non vi è contenuto informativo temporale e in quella temporale non si hanno informazioni sul contenuto frequenziale; in parole povere, con la trasformata di Fourier non è possibile localizzare temporalmente in un segnale una componente frequenziale presente.

Questa informazione combinata non risulta fondamentale per i segnali stazionari, ossia per quella classe di segnali in cui il contenuto frequenziale non cambia nel tempo. In questo caso specifico, non è necessario conoscere in che momento si manifesta una certa componente frequenziale del segnale perché questa sarà presente lungo tutto l'intervallo esaminato, in qualsiasi momento. Nel caso in cui, invece, venissero esaminati segnali con componenti frequenziali che compaiono in diversi istanti dell'intervallo considerato, la trasformata di Fourier mostrerebbe queste componenti, ma non potrebbe specificare quando queste sono comparse nella finestra temporale selezionata.

Molti dei segnali utilizzati e misurati sono non-stazionari, ad esempio quasi tutti i segnali biologici, tra cui i segnali EMG spiegati nel Capitolo 1.

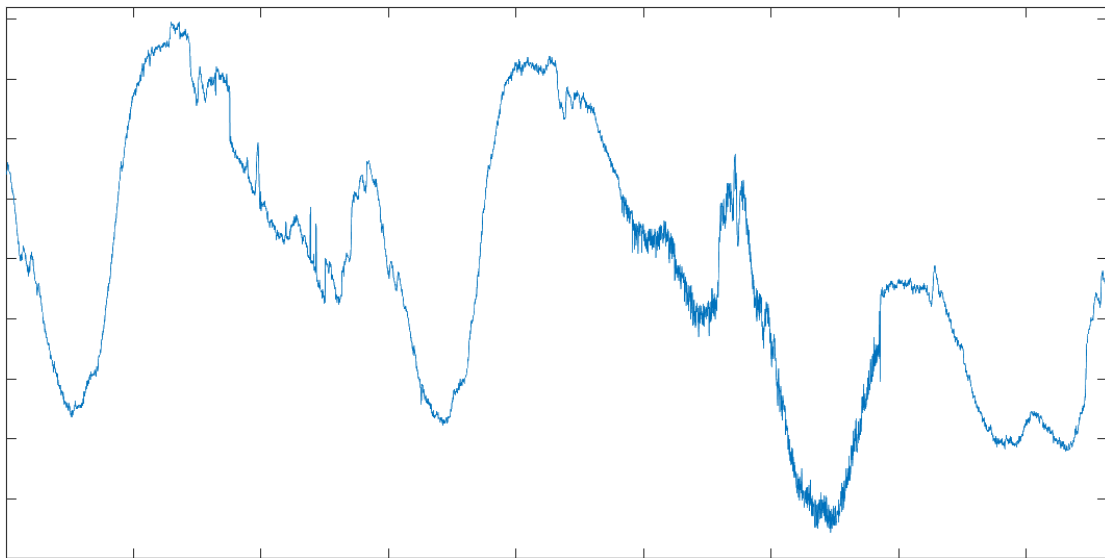


Fig. 4 Esempio di segnale non stazionario presente in MATLAB.

2.2 Short Time Fourier Transform

Nel tentativo di poter applicare la trasformata di Fourier anche ai segnali non-stazionari è stata sviluppata una variante della FT chiamata Short Time Fourier Transform (STFT). In

questa altra versione il segnale viene diviso in tanti piccoli segmenti di uguale lunghezza, in modo tale che gli intervalli che andranno analizzati con la trasformata di Fourier risultino stazionari.

A questo scopo viene scelta una funzione finestra w di larghezza T , diversa da zero solo in un piccolo intervallo temporale. Quindi, verrà man mano traslata lungo tutto l'asse temporale e moltiplicata per il segnale originario, così che il risultato di questa operazione venga dato in pasto alla trasformata di Fourier per ottenere una rappresentazione tempo-frequenza del segnale originario.

$$\mathbf{STFT}\{x(t)\}(\tau, \omega) \equiv X_\omega(\tau, \omega) == \int_{-\infty}^{+\infty} [x(t) * \bar{w}(t - \tau)] e^{-i\omega t} dt$$

Sebbene risolva il problema dato dalla trasformata di Fourier con questo tipo di segnali, la STFT risulta comunque poco pratica a causa del fatto che la sua risoluzione è fissa e non cambia. Infatti, la larghezza della funzione finestra determina se il segnale avrà una buona risoluzione frequenziale oppure una buona risoluzione temporale. Una finestra larga fornirà una buona risoluzione frequenziale (ed una scarsa risoluzione temporale), mentre una finestra stretta otterrà il risultato opposto.

2.3 Wavelet transform

Un metodo alternativo, sviluppato per ovviare a questo problema della STFT, è la *Multi Resolution Analysis* (MRA), che va ad analizzare il segnale a diverse frequenze con differenti risoluzioni. Consiste in un'analisi a risoluzione variabile in cui all'aumentare della frequenza f aumenta anche la banda Δf (contenuto frequenziale del segnale) in modo proporzionale. Si avrà quindi una buona risoluzione frequenziale alle alte frequenze e una buona risoluzione temporale alle basse frequenze.

Ciò è reso possibile costruendo un insieme di funzioni base mediante traslazione e cambiamento di scala di un'unica funzione, $\psi(t)$, detta *mother wavelet*:

$$\psi_{\tau s}(t) = \frac{1}{\sqrt{s}} \psi\left(\frac{t - \tau}{s}\right)$$

A partire dalla mother wavelet è quindi possibile definire lo strumento principe della MRA, la trasformata wavelet continua. Il suo funzionamento è concettualmente simile a quello della STFT, poiché anche in questo caso il segnale originario verrà moltiplicato, lungo tutto l'asse temporale, per una funzione finestra (la *mother wavelet* appena mostrata). Questa è la formulazione della trasformata:

$$\mathbf{CWT}\{x(t)\}(\tau, s) \equiv X_\omega(\tau, s) = \frac{1}{\sqrt{|s|}} \int_{-\infty}^{+\infty} x(t) \bar{\psi}\left(\frac{t - \tau}{s}\right) dt$$

Come si può notare il segnale trasformato è funzione di due variabili, τ e s , che corrispondono alla traslazione e alla scala del nuovo segnale.

La mother wavelet nella WT non viene unicamente traslata come invece accadeva per la funzione finestra nella STFT, ma verrà anche modificata in termini di scala. Il concetto di scala è simile a quello usato nelle cartine geografiche: un valore di scala alto corrisponderà ad una visione globale, poco dettagliata, della funzione, mentre un valore basso corrisponderà ad una visione molto più localizzata e precisa. Allo stesso modo, nella nostra

analisi, alle basse frequenze (valore di scala alto) si avrà una visione globale del segnale, mentre alle alte frequenze (scala bassa) si avrà una visione più dettagliata e ristretta.

In termini matematici, se $f(t)$ è una funzione data, allora $f(t/s)$ corrisponde ad una versione dilatata di $f(t)$, se $s > 1$, oppure ad una versione contratta (o compressa) di $f(t)$, se $s < 1$.

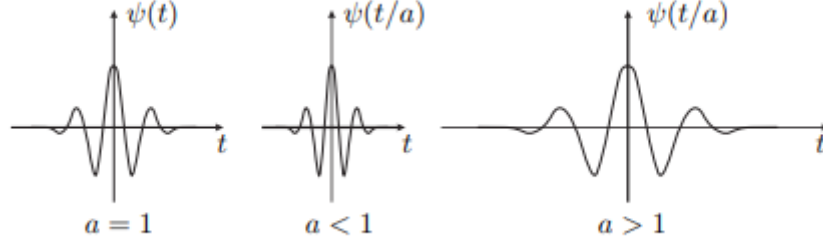


Fig. 5 Esempi di wavelet a diversi valori di scala.

Dal punto di vista computazionale la CWT corrisponde ad un'operazione di moltiplicazione della wavelet con il segnale originario per tutto l'asse temporale (traslazione) e quindi varie ripetizioni di questa operazione con differenti wavelet scalate.

La relazione che permette di ricostruire il segnale a partire da quello trasformato è la seguente:

$$x(t) = \frac{1}{C_\psi} \int \int_{-\infty}^{+\infty} \frac{1}{a^2} CWT[\tau, s] \psi_{\tau s}(t) d\tau ds$$

Perché sia garantita l'invertibilità della trasformata, la costante C_ψ deve essere una quantità finita:

$$C_\psi = \int_0^{+\infty} \frac{|\Psi(f)|^2}{f} df < \infty$$

La funzione $\Psi(f)$ corrisponde alla trasformata di Fourier della mother wavelet $\psi(t)$ e decade rapidamente a zero grazie alla forma tipica della wavelet (proprietà di localizzazione in frequenza). Quindi, l'unica condizione da imporre perché C_ψ sia una quantità finita è che $\Psi(0) = 0$, ossia che il valore medio della wavelet sia nullo, poiché:

$$\Psi(0) = \int_{-\infty}^{+\infty} \psi(t) dt$$

Allo scopo di determinare la capacità di localizzazione tempo-frequenza della trasformata, si calcolano i corrispettivi valori di durata e banda efficace (Δt e Δf):

$$(\Delta t)^2 = \int_{-\infty}^{+\infty} t^2 |\psi(t)|^2 dt$$

$$(\Delta f)^2 = \int_{-\infty}^{+\infty} (f - f_0)^2 |\Psi(f)|^2 df$$

Considerando una cella con $\tau = \tau_0$ e $s = s_0$, si ottiene:

$$\begin{aligned}
(\Delta t)_{s_0 \tau_0}^2 &= \int_{-\infty}^{+\infty} (t - \tau_0)^2 |\psi_{\tau_0 s_0}(t)|^2 dt = \int_{-\infty}^{+\infty} (t - \tau_0)^2 \frac{1}{s_0} \psi^2\left(\frac{t - \tau_0}{s_0}\right) dt \\
&= s_0 \int_{-\infty}^{+\infty} \tau^2 \psi^2(\tau) d\tau = s_0^2 (\Delta t)^2
\end{aligned}$$

$$(\Delta f)_{s_0 \tau_0} = \frac{\Delta f}{s_0}$$

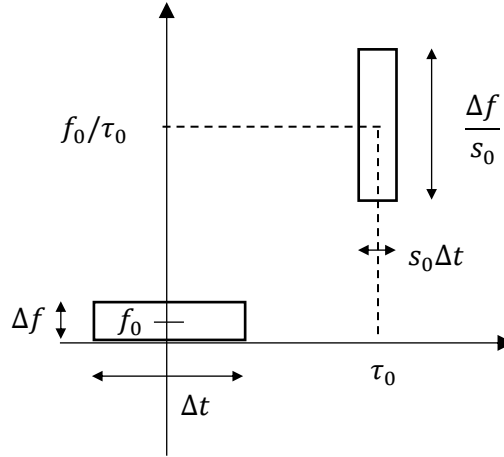


Fig. 6 Finestra di analisi della trasformata in base ai valori di τ e s scelti.

Mostrando che la risoluzione su tutto il piano tempo-frequenza sarà come in figura 7.

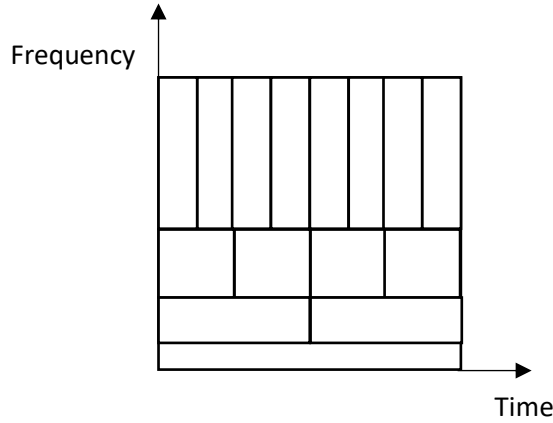


Fig. 7 Andamento delle finestre di analisi lungo il diagramma, tempo-frequenza.

Affinché il piano sia completamente coperto come nella figura sopra, la discretizzazione del parametro τ deve essere legata a quella del parametro s , cosicché funzioni compresse sull'asse temporale vengano traslate con passo piccolo, mentre funzioni espanse siano traslate con passo grande. La discretizzazione è così formulata:

$$\begin{cases} s = s_0^{-m}, & s_0 > 1 \quad m \in \mathbb{Z} \\ \tau = n\tau_0 s_0^{-m}, & \tau_0 > 0 \quad n \in \mathbb{Z} \end{cases}$$

Di particolare interesse nell'elaborazione dei segnali è la discretizzazione su una griglia diadica, che si ottiene dalla definizione di mother wavelet per $s_0 = 2$ e $\tau_0 = 1$:

$$\psi_{m,n}(t) = 2^{\frac{m}{2}} \psi(2^m t - n)$$

Da cui deriva quanto mostrato in figura sopra. Si può osservare, quindi, che le funzioni wavelet presentano una buona risoluzione frequenziale alle basse frequenze ed una buona risoluzione temporale alle alte frequenze.

2.4 Analisi multi risoluzione (MRA)

Considerando il caso più semplice ossia quello in cui le funzioni wavelet sono di tipo Haar, si descrive ora il legame fra WT e analisi multi risoluzione. Dato un generico segnale $x(t)$ se ne ricava una rappresentazione approssimata $x^{(0)}(t)$ utilizzando la seguente famiglia di funzioni $\phi_{0k}(t) = \phi(t - k)$ ¹:

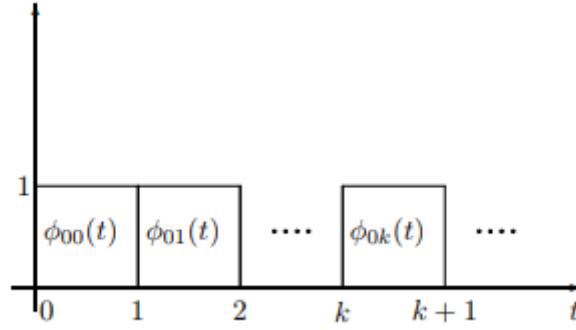


Fig. 8 Rappresentazione delle varie funzioni sull'asse temporale.

L'espansione in serie del segnale $x(t)$ può essere scritta nella seguente forma:

$$x^{(0)}(t) = \sum_k c_{0,k} \phi_{0,k}(t)$$

I cui coefficienti $c_{0,k}$ sono dati dalla relazione:

$$c_{0,k} = \langle x(t), \phi_{0,k}(t) \rangle = \int_{-\infty}^{+\infty} x(t) \phi_{0,k}(t) dt = \int_k^{k+1} x(t) dt$$

Se si volesse ottenere una rappresentazione del segnale a risoluzione maggiore bisognerebbe considerare delle funzioni più strette, $\phi_{1,k}(t) = \sqrt{2} \phi(2t - k)$ così da ottenere una versione più accurata di $x(t)$:

$$x^{(1)}(t) = \sum_k c_{1,k} \phi_{1,k}(t)$$

I coefficienti $c_{1,k}$ questa volta sono dati da:

$$c_{1,k} = \int_{-\infty}^{+\infty} x(t) \phi_{1,k}(t) dt = \sqrt{2} \int_{\frac{k}{2}}^{\frac{k+1}{2}} x(t) dt$$

La nuova rappresentazione conterrà sicuramente anche quella più grossolana vista prima, il che ci permette di definire la seguente famiglia di funzioni:

$$\phi_{j,k}(t) = 2^{j/2} \phi(2^j t - k)$$

Le funzioni $\phi_{j,k}(t)$ vengono dette funzioni di scaling poiché forniscono una rappresentazione del segnale a diverse risoluzioni $x^{(0)}(t), x^{(1)}(t), \dots, x^{(n)}(t)$ al variare di j . Queste funzioni generano n spazi vettoriali V_j ed ognuno di questi è costituito da tutti quei segnali che possono essere espressi tramite combinazioni lineari delle funzioni $\phi_{j,k}(t)$. Questo insieme di segnali a diversa risoluzione, $x^{(0)}(t), \dots, x^{(n)}(t)$ ottenuto risulta essere fortemente ridondante, essendo i vari spazi sottoinsiemi l'uno dell'altro in base alla risoluzione a cui si riferiscono:

$$V_0 \subset V_1 \subset \dots \subset V_n$$

Al fine di non avere questa ridondanza si consideri un segnale appartenente allo spazio V_1 . Risulta che:

$$x^{(1)}(t) = x^{(0)}(t) + (x^{(1)}(t) - x^{(0)}(t)), \quad \text{con } x^{(1)}(t) \in V_1$$

Facendo riferimento alla differenza $D^0(t) = x^{(1)}(t) - x^{(0)}(t)$, la si può esprimere concentrandosi al generico intervallo $[k, k+1)$:

$$x^{(1)}(t) - x^{(0)}(t) = \begin{cases} \sqrt{2}c_{1,2k} - c_{0,k}, & k \leq t < k + \frac{1}{2} \\ \sqrt{2}c_{1,2k+1} - c_{0,k}, & k + \frac{1}{2} \leq t < k + 1 \end{cases}$$

Tenendo conto della relazione fra i coefficienti $c_{0,k} = \frac{1}{\sqrt{2}}(c_{1,2k} + c_{1,2k+1})$ e riscrivendola, con $d_{0,k} = c_{1,2k} - c_{1,2k+1}$:

$$x^{(1)}(t) - x^{(0)}(t) = \begin{cases} d_{0,k}, & k \leq t < k + \frac{1}{2} \\ -d_{0,k}, & k + \frac{1}{2} \leq t < k + 1 \end{cases}$$

La differenza sopra può quindi essere riscritta come D^0 :

$$D^0 = x^{(1)}(t) - x^{(0)}(t) = \sum_k d_{0,k} \psi_{0,k}(t)$$

Dove $\psi_{0,k}(t)$ rappresenta l'insieme delle traslazioni della mother wavelet di Haar. Le funzioni wavelet ortogonali di Haar formano quindi l'insieme delle funzioni che generano D^0 . In conclusione, il segnale appartenente allo spazio V_1 può essere riscritto come:

$$x^{(1)}(t) = x^{(0)}(t) + D^0 = \sum_k c_{0,k} \phi_{0,k}(t) + \sum_k d_{0,k} \psi_{0,k}(t)$$

Da cui, iterando il procedimento, si ottiene una rappresentazione non ridondante:

$$\begin{aligned}
x^{(j+1)}(t) &= x^{(j)}(t) + D^j = x^{(j-1)}(t) + D^{j-1} + D^j = x^{(0)}(t) + D^0 + \dots + D^j \\
&= x^{(0)}(t) + \sum_j \sum_k d_{j,k} \psi_{j,k}(t)
\end{aligned}$$

2.5 Discrete Wavelet Transform (DWT)

Si può dimostrare che la funzione di scaling e la mother wavelet possono essere espresse come combinazione lineare attraverso opportuni coefficienti h_n e g_n .

$$\phi(t) = \sqrt{2} \sum_n h_n \phi(2t - n)$$

$$\psi(t) = \sqrt{2} \sum_n g_n \psi(2t - n)$$

Dove:

$$h_n = \langle \phi(t), \phi_1(t) \rangle = \sqrt{2} \int_{-\infty}^{+\infty} \phi(t) \phi(2t - n) dt$$

$$g_n = \langle \psi(t), \phi_1(t) \rangle = \sqrt{2} \int_{-\infty}^{+\infty} \psi(t) \phi(2t - n) dt$$

Si mostra ora la struttura mediante banco di filtri che permette di calcolare i coefficienti wavelet per un segnale discreto, ottenendo la rappresentazione nota come Discret Wavelet Transform. Considerando ancora una volta un segnale appartenente a V_1 si può dimostrare che la relazione fra $c_{1,n}$, $c_{0,n}$ è:

$$c_{0,n} = \sum_k c_{1,k} h_{k-2n} = \sum_k c_{1,k} \tilde{h}_{2n-k}$$

Dove $\tilde{h}_n = -h_n$. La sequenza di coefficienti $c_{0,n}$ si ottiene, pertanto, mediante filtraggio e decimazione per 2 (*subsample*) della sequenza $c_{1,n}$. Si può dimostrare anche che:

$$d_{0,n} = \sum_k c_{1,k} g_{k-2n} = \sum_k c_{1,k} \tilde{g}_{2n-k}$$

Il legame tra i coefficienti è mostrato nello schema a blocchi in figura 9.

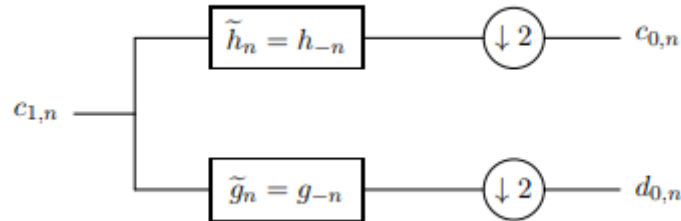


Fig. 9 Schema di filtraggio e subsample per un livello di decomposizione.

Provando a ripetere questo processo anche per il caso di una funzione appartenente a V_2 non cambia il legame fra i coefficienti e lo schema di decomposizione a due livelli diventa quello in figura 10. Lo schema può quindi essere iterato per tutti i livelli di decomposizione desiderati.

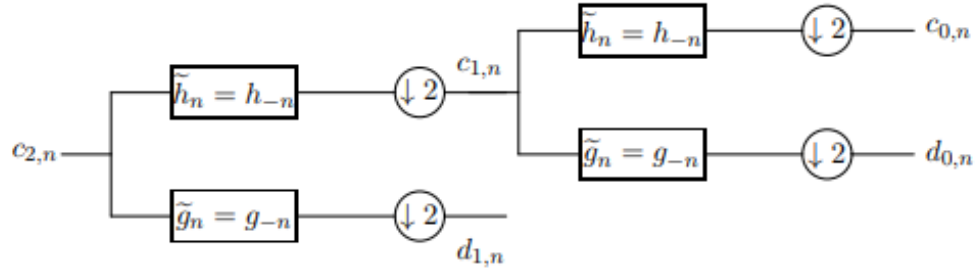


Fig. 10 Schema di filtraggio e subsample per due livelli di decomposizione.

Capitolo 3: Sistema di classificazione

Dopo questa parte più teorica, posta per dare una conoscenza degli strumenti utilizzati, si passa ora alla descrizione del processo seguito per la classificazione dei gesti.

Lo scopo principale di questo processo di classificazione, e del lavoro intero di tesi, è quello di riuscire ad ottenere un sistema di interfacciamento ad una mano robotica basato su segnali sEMG, di facile utilizzo.

Il sistema di classificazione utilizzato è basato su una metodologia di apprendimento supervisionato (*supervised learning*) che opera in questo modo: si punta ad istruire uno specifico algoritmo in modo da consentirgli di riconoscere, in maniera autonoma, la classe di appartenenza di particolari dati, sulla base di una serie di esempi ideali che gli vengono precedentemente forniti. All'atto pratico, il processo di classificazione si realizza in due fasi distinte: una fase iniziale, detta di *training*, per “allenare” il sistema, ed il riconoscimento vero e proprio della classe dei dati. Nel primo stadio, quello di *training*, il sistema viene munito di un set di esempi sufficiente per permettergli di elaborare un modello da utilizzare per il riconoscimento, poi si procede a testare l'efficacia del classificatore valutando le sue risposte a dei dati di cui è già nota la classe all'utente.

In questo studio, questo processo di realizzazione di un sistema di classificazione è stato svolto in due modalità differenti: una prima elaborazione e classificazione off-line, con dei dati registrati precedentemente in due sessioni distinte, ed una seconda on-line, in cui si generava il modello e si valutava l'efficacia del classificatore in “real time”.

Per l'acquisizione del segnale, in entrambi i casi, sono stati posti sul braccio di un utente 8 coppie di elettrodi a gel di tipo sEMG. Sono stati situati in una zona dell'avambraccio prossima al gomito, dove sono presenti i muscoli *Flexor Digitorum Superficiali* e *Extensor Digitorum Communis*, con una disposizione tale da coprire tutta la circonferenza del braccio, come mostrato in figura. Questi elettrodi, poi, sono stati collegati ad una scheda di acquisizione dati appositamente progettata per l'acquisizione di segnali biologici, la quale era già disponibile al LAR (Laboratorio di Automatica e Robotica dell'Università di Bologna). Compito della scheda è stato quello di misurare i potenziali muscolari per mezzo

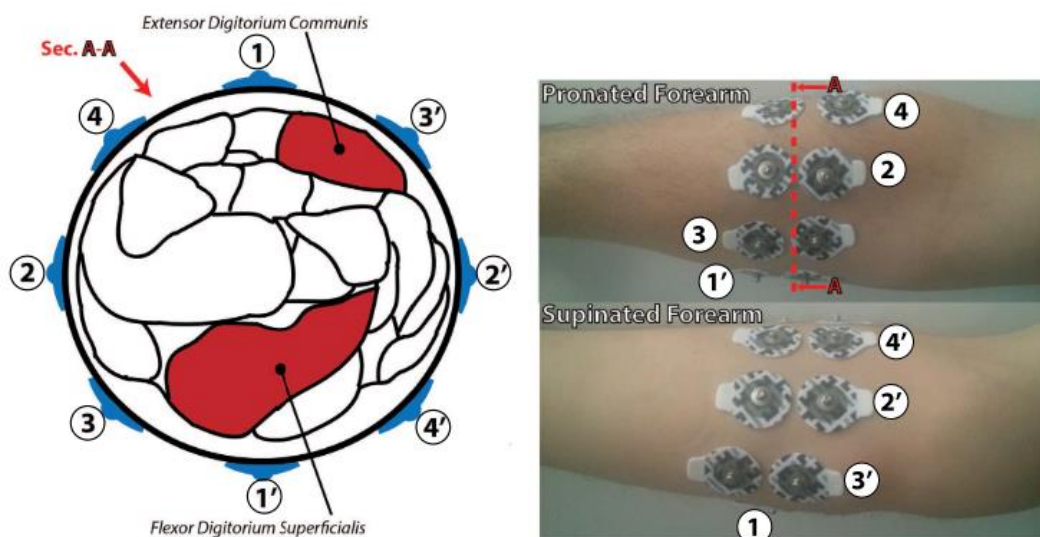


Fig. 11 Posizionamento elettrodi, da Meattini e al. (2018), vedi bibliografia.

degli elettrodi e trasmettere, via bluetooth, il segnale multi-canale ad un computer che potesse utilizzarlo o salvarlo.

Dopo il processo di acquisizione il segnale veniva trattato ed elaborato allo scopo di estrarre sue “caratteristiche” che potessero essere facilmente impiegabili per l’elaborazione di un sistema di classificazione efficiente. Queste “caratteristiche” del segnale, in inglese, prendono il nome di *features* e vengono classicamente chiamate in questa maniera in letteratura. Allo scopo di estrarre queste informazioni, il segnale veniva inizialmente filtrato e “smussato” con particolari funzioni, al fine di eliminare le interferenze esterne ed avere una versione più pulita del segnale, i dettagli verranno forniti nel paragrafo 3.2. Poi si passava alla vera e propria estrazione di features. Il segnale veniva diviso in tanti intervalli temporali di lunghezza prestabilita da cui si ottenevano i coefficienti di dettaglio con l’utilizzo della trasformata Wavelet discreta. Poi, con un processo utilizzato anche da (Lucas et al., 2008) e che verrà spiegato al paragrafo 3.2, da questi coefficienti si ricavano una serie di vettori contenenti le feature da utilizzare per il *training* del classificatore.

Nella prima fase, quella di classificazione off-line, si è valutato maggiormente quali wavelet o quali intervalli di tempo utilizzare per ottenere la migliore efficienza del classificatore. Nella seconda fase si sono testati i risultati di questa prima analisi generando uno schema Simulink che, dopo una sessione di training, potesse trattare il segnale, estrarne le feature e riconoscere il gesto eseguito dall’utente, per poi controllare una mano robotica in modo che potesse replicarlo.

Per classificare i dati ottenuti si è utilizzata una metodologia di apprendimento automatico che prende il nome di “Macchina a vettori di supporto”, abbreviata in SVM (dall’inglese *Support Vector Machine*).

3.1 SVM

In *machine learning* per macchina a vettori di supporto si intende un modello di apprendimento supervisionato composto da specifici algoritmi per la classificazione e la regressione di dati

I modelli SVM furono inizialmente definiti per la classificazione di classi di oggetti linearmente separabili. Per ogni gruppo di elementi divisi in due classi una SVM identifica l’iperpiano avente il massimo margine di separazione. Dato un *training set* composto di coppie elemento-*label* assegnata (x_i, y_i) , con $i = 1, \dots, l$ e x_i appartenente a \mathbb{R} , e y appartenente a $\{1, -1\}^l$, la macchina a vettori di supporto richiede la soluzione del seguente problema di ottimizzazione:

$$\min_{w, b, \xi} \frac{1}{2} \mathbf{w}^T \mathbf{w} + C \sum_{i=1}^l \xi_i$$

Tenendo conto che:

$$\begin{aligned} y_i(\mathbf{w}^T \phi(x_i) + b) &\geq 1 - \xi_i, \\ \xi_i &\geq 0 \end{aligned}$$

Qui i vettori di training x_i sono mappati in uno spazio avente dimensione maggiore dalla funzione ϕ . La SVM trova un iperpiano lineare che separi le classi di elementi, con il

massimo margine possibile in questo spazio multidimensionale. $C > 0$ è il *penalty parameter* del termine errore. Invece, $K(\mathbf{x}_i, \mathbf{x}_j) \equiv \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j)$ è chiamata funzione *kernel*.

Nella libreria libSVM (Chang et al., 2011. Vedi bibliografia) utilizzata in questo lavoro di tesi sono presenti quattro tipologie di funzioni kernel:

- Linear: $K(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i^T \mathbf{x}_j$.
- Polynomial: $K(\mathbf{x}_i, \mathbf{x}_j) = (\gamma \mathbf{x}_i^T \mathbf{x}_j + r)^d, \gamma > 0$.
- Radial basis function (RBF): $K(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2), \gamma > 0$.
- Sigmoid: $K(\mathbf{x}_i, \mathbf{x}_j) = \tanh(\gamma \mathbf{x}_i^T \mathbf{x}_j + r)$.

γ, r e d sono parametri delle funzioni. Per la loro scelta eseguiremo un processo di ottimizzazione sul training set che permetta di ottenere di questi parametri i migliori per rendere più efficiente il classificatore.

3.2 Sistema di classificazione off-line e validazione

Il primo test fatto consisteva nella creazione e nella prova del classificatore off-line.

Per prima cosa sono state fatte due sessioni di acquisizione del segnale 8-dimensionale dei canali sEMG. Dopo aver posto sull'avambraccio dell'utilizzatore 8 coppie di elettrodi in una zona prossima al gomito, come mostrato precedentemente in figura 11, si è proceduto con il seguente protocollo di acquisizione. I 6 gesti da ripetere, per poi essere riconosciuti dal classificatore, erano: mano aperta, presa di potenza, 3 dita aperte, presa ulnare, presa di precisione, indice puntato; a queste posizioni va aggiunto un settimo gesto corrispondente alla posa neutrale, di riposo, delle dita della mano (fig.12). La sequenza dei 6 gesti

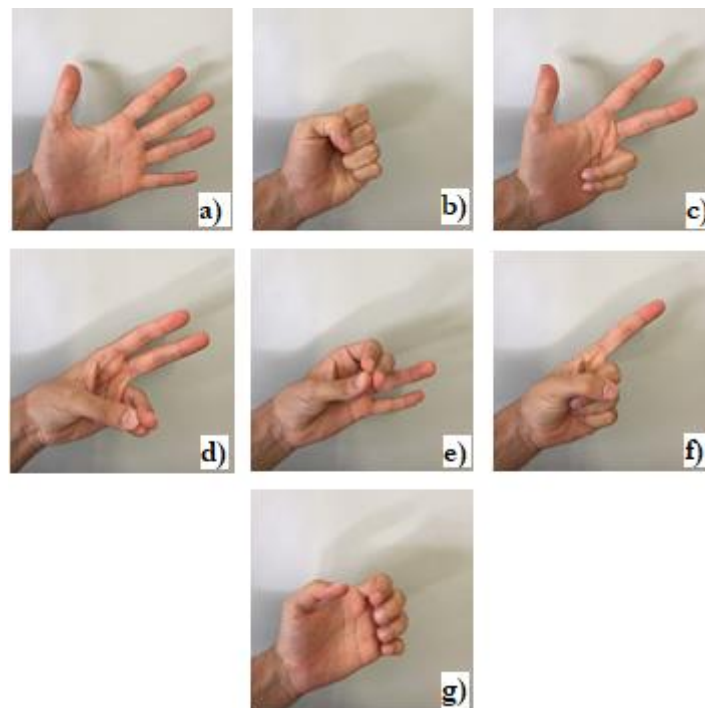


Fig. 12 Lista di gesti classificati: a) mano aperta, b) presa di potenza, c) tre dita aperte, d) presa ulnare, e) presa di precisione, f) indice alzato, g) posa neutrale.

comprendeva il mantenimento della posizione, di ognuno di questi, per una durata di 5 s, alternando ogni gesto con un intervallo di 5 s di riposo per le dita, in posa neutrale. La

durata totale della sequenza era quindi di circa 1 minuto. È stata ripetuta per 5 volte a sessione, così da avere un set di dati di una lunghezza sufficientemente grande per la creazione del modello SVM. In ogni sessione veniva fornito all'utente un segnale visivo che lo avvisasse di eseguire un gesto oppure di tenere la mano in posa neutrale.

Acquisiti i dati, sono poi stati filtrati. Inizialmente sono stati passati in un filtro elimina banda (notch) per eliminare interferenze esterne a 50 Hz, date soprattutto dai numerosi apparecchi elettronici presenti nel laboratorio in cui è stata eseguita la misurazione. Quindi per mezzo di un filtro passa-alto sono state eliminate tutte le frequenze al di sotto di quella di taglio del filtro, pari a 20 Hz . Successivamente, dai dati è stato estratto il valore efficace su una finestra mobile, di lunghezza 200 ms, con sovrapposizione (*overlap*) massima. Questo corrisponde ad eseguire il valore efficace su un intervallo finito di campioni, spostare questo intervallo di un campione, calcolare il valore efficace del nuovo intervallo e, quindi, ripetere questo procedimento per tutta la lunghezza del segnale, ottenendo così un nuovo segnale di uguale lunghezza del precedente ma più “smussato”, senza grosse oscillazioni fra campioni consecutivi.

$$RMS(k) = \sqrt{\frac{1}{N} \sum_{i=0}^N X(k-i)^2}$$

Successivamente, al segnale venivano eliminati i campioni corrispondenti ai transitori fra i vari gesti (non utili ai fini della classificazione) e nel frattempo veniva creato il vettore contenente la classe di appartenenza di ogni campione (*label*) .

A questo punto, con apposite funzioni MATLAB, sono stati estratti i coefficienti per mezzo della trasformata Wavelet. In particolare, si è deciso di operare fino al quarto di livello di decomposizione. Per l'estrazione sono state usate tre finestre di lunghezza diversa: 100 ms, 200 ms e 400 ms; oltre a questo è stata fatta un' ulteriore estrazione, per lunghezza di finestra, con overlap pari alla lunghezza di metà finestra.

Le wavelet utilizzate e testate per questa prima fase sono state 9, appartenenti a 3 famiglie di wavelet ortogonali (uniche utilizzabili nella DWT):

- Daubechies (di grado 2, 3 e 4);
- Symlet (di grado 2, 3 e 4);
- Coiflet (di grado 2, 3 e 4).

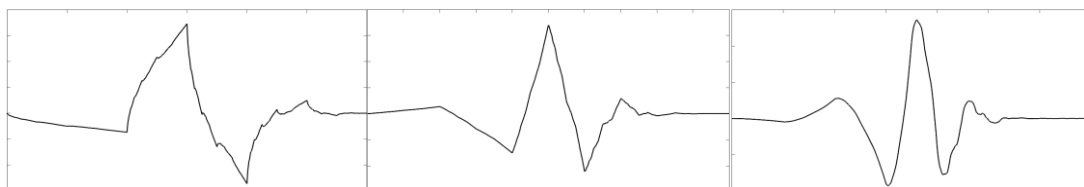


Fig. 13 Wavelet della famiglia Daubechies: 2, 3 e 4.

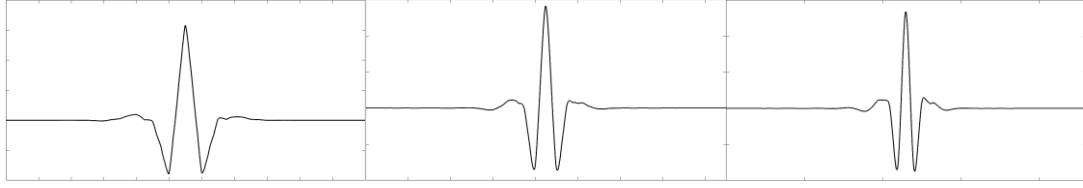


Fig. 14 Wavelet della famiglia Coiflet: 2, 3 e 4.

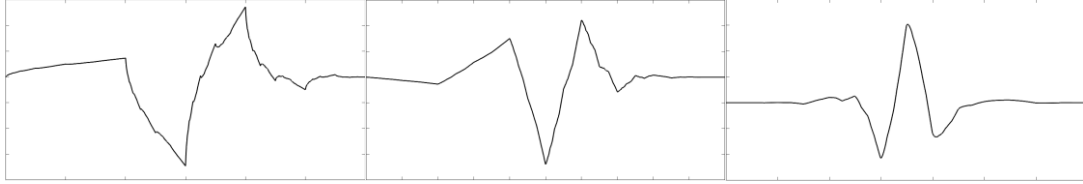


Fig. 15 Wavelet della famiglia Symlet: 2, 3 e 4.

Ottenuti i coefficienti per ognuno degli otto canali, ne sono stati calcolati i marginali allo stesso modo di come fatto da Lucas et al., 2008. Si è fatta una sommatoria, per livello di decomposizione, dei valori assoluti di tutti i coefficienti presenti, così da ottenere un valore per livello:

$$m_{x_k}(s) = \sum_{u=0}^{N/2^s-1} |d_{x_k}(s, u)|, \quad s = 1, \dots, 4$$

$m_{x_k}(s)$ è un vettore di marginali, x_k corrisponde al segnale del canale, d_{x_k} è invece l'insieme dei coefficienti di dettaglio del segnale.

Calcolati i marginali per ogni intervallo in cui si era suddiviso il segnale, si è costruita la matrice delle feature da utilizzare per la creazione della SVM. In questa matrice ogni riga corrispondeva ad un intervallo di tempo e conteneva i marginali per ognuno dei 4 livelli di decomposizione e per tutti gli 8 canali, quindi 32 elementi. In più, è stato aggiunto il vettore delle *label* come ultima colonna della matrice, in modo tale che l'ultimo elemento di ogni riga corrispondesse alla classe (ossia il gesto eseguito) assegnata a quell'intervallo.

In seguito, si è passati alla classificazione vera e propria e alla validazione off-line. Si è per questo operata una permutazione casuale delle righe della matrice delle feature, così da non avere eventuali distorsioni nel classificatore.

Quindi si è diviso il set di elementi in dieci gruppi di uguali dimensioni. Nove di questi gruppi sono stati utilizzati per ottenere un *training set* ed il restante gruppo è divenuto il *validation set*. Quest'ultimo è stato utilizzato successivamente per testare l'efficacia della SVM ottenuta con il training set. Il processo di divisione dei dati in training e validation set e successivo confronto del modello, ottenuto dal primo, con i dati del secondo prende il nome di *cross-validation*. Per la generazione del SVM i dati andavano precedentemente normalizzati, in modo tale che l'algoritmo per la generazione del classificatore ricevesse valori paragonabili fra di loro, e si è operato come segue:

$$\frac{x_i - \bar{x}_i}{\sigma_i}, \quad \text{con } i = 1, \dots, 32$$

Con \bar{x}_i pari al valor medio di x_i e σ_i pari alla sua deviazione standard.

Prima però di ottenere un modello, veniva fatto già un processo di cross-validation all'interno del training set, quindi solo con quei dati, in modo tale da trovare i parametri d e γ della funzione kernel migliori per la generazione del SVM.

Fatta questa operazione, un altro dei dieci gruppi è stato eletto a validation set e i restanti elementi usati come training set e si è iterato questo procedimento di divisione e successiva cross-validation per ognuno dei dieci gruppi in cui era stata divisa la matrice delle feature.

I risultati dei vari test effettuati sono stati sommati e raccolti in delle matrici di confusione che servono per mappare le corrispondenze fra i valori predetti dal classificatore e i valori reali. Le matrici si presentano come in figura 16.

		Confusion Matrix						
True class	A	334	3	109	9	5		10
	B		424	2	8	9	18	9
	C	100	2	269	43	24	2	30
	D	11	10	19	344	49	6	31
	E	2	16	20	38	322	8	64
	F		13	1	4	5	420	27
	G	17	1	24	15	33	6	370
		A	B	C	D	E	F	G
		Predicted class						

Fig. 16 Esempio di matrice di confusione

In una qualsiasi riga della matrice compaiono tutti gli elementi che realmente appartengono alla classe della riga, mentre, su una colonna, avremo tutti quegli elementi che il classificatore ha predetto come appartenenti alla classe della colonna. Per questo, presa una qualsiasi cella (m, n) della matrice avremo che:

- Se $m = n$, allora nella cella compare il numero di elementi che sono stati classificati correttamente come appartenenti alla classe $m(= n)$.
- Se $m \neq n$, nella cella si presenta il numero di volte che gli elementi della classe m sono stati predetti come elementi di classe n .

Sulla diagonale compariranno quindi i risultati positivi classe per classe, mentre nel resto della matrice si può osservare come, e quante volte, sbaglia il classificatore. Da queste matrici che sono state ottenute si è quindi calcolato, per ognuna di queste, il *Balanced Error Rate* (BER). Questo parametro serve per fornire una stima dell'efficacia del classificatore, basandosi sui risultati della validazione. Data una matrice $m \times n$, il cui generico elemento corrisponde a a_{kj} , il Balanced Error Rate si calcola sommando per ogni classe il rapporto fra valori sbagliati e valori reali di quella classe, nella seguente maniera:

$$BER = 1/N \sum_{k=1}^N \frac{\sum_{i=1, i \neq k}^3 a_{ki}}{\sum_{j=1}^3 a_{kj}}$$

Con N pari al numero totale di classi in cui è suddiviso il segnale.

3.3 Sistema di classificazione on-line

La classificazione on-line aveva come scopo quello di testare il classificatore su segnali acquisiti sul momento per replicare dal vivo con una mano robotica il gesto eseguito dall'utente

Per realizzare questo scopo si è implementato un processo di trattamento del segnale ed elaborazione del classificatore identico a quello usato nel sistema di classificazione off-line mostrato nel paragrafo 3.2.

Anche in questo schema è stata implementata una modalità per la sessione di training, così da poter generare un modello sul momento per poi eseguire la validazione. Non si è pensato di utilizzare modelli generati off-line con i dati delle sessioni precedenti. Infatti, pur sfruttando valori di finestra di estrazione e wavelet già vagliati nella classificazione off-line, le misurazioni EMG possono variare di volta in volta in maniera spesso imprevedibile a causa di differenti fattori (stanchezza dell'utente, disposizione degli elettrodi, sudorazione del braccio, ecc.).

A questo scopo, lo schema Simulink sviluppato ha permesso in un primo tempo di svolgere una sessione di “allenamento” per l'acquisizione del segnale e la generazione del modello SVM, per poi passare alla modalità di validazione con controllo della mano robotica. L'utente mentre ripeteva la sequenza già spiegata al paragrafo 3.2, ha avuto come guida un segnale di riferimento che gli indicasse quando eseguire un gesto o quando mantenere a riposo le dita. La sequenza è stata, quindi ripetuta per 6 volte.

Anche in questo schema, i dati raccolti sono stati passati per due filtri, prima dell'estrazione delle feature o dell'acquisizione per il training. I filtri erano un notch a 50 Hz ed un passa-alto con frequenza di taglio pari a 20 Hz. A seguire, è stato calcolato il valore efficace del segnale su una finestra mobile di 200 ms con sovrapposizione della finestra (*overlap*) massima.

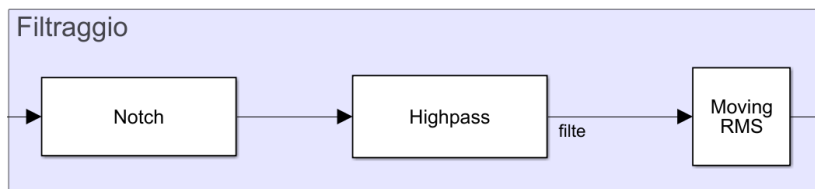


Fig. 17 Blocco di filtraggio nello schema Simulink

Dopo questo processo di filtraggio il segnale veniva salvato e utilizzato per il calcolo del SVM. Attraverso un apposito script sviluppato per questo scopo, il sistema estraeva le feature (utilizzando DWT e marginali), normalizzava i dati, li mischiava e procedeva con un processo di cross-validation per la scelta dei migliori parametri della funzione kernel e sviluppava un modello.

Per il processo di validazione lo schema è stato pensato per riprodurre le stesse operazioni di estrazione delle feature, svolte off-line o per il training. Per prima cosa il segnale veniva trasformato per mezzo della DWT e i coefficienti ottenuti venivano sommati e raccolti così da ottenere i marginali e comporre un vettore di 32 elementi da poter dare in osservazione al classificatore.

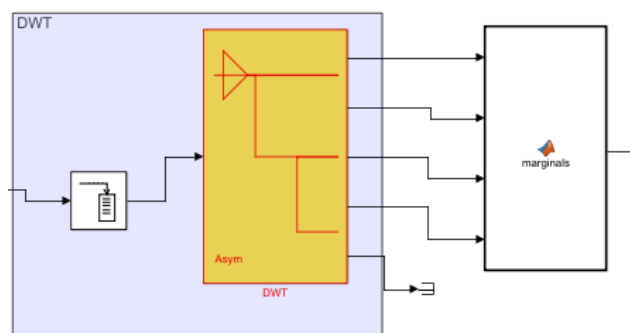


Fig. 18 Blocco per l'estrazione delle features nello schema Simulink

Quindi venivano normalizzati, utilizzando i valori di media e di deviazione standard ottenuti nella fase di training, e poi venivano passati al classificatore che effettuava il suo lavoro di riconoscimento.

L'uscita del classificatore andava a comandare la mano robotica UB Hand (University of Bologna Hand), sviluppata e posseduta dal Laboratorio di Automatica e Robotica (LAR) dell'Università di Bologna. La si è programmata in modo che potesse replicare i 6 gesti e che mantenesse la posizione precedente nel caso fosse attivo un comando specifico, in validazione questo comando faceva riferimento alla posa neutrale riconosciuta dal classificatore. Si è impostato anche un comando che permettesse di forzare la mano in una posizione di home per mantenerla ferma quando si fermava la validazione.



Fig. 19 UB hand dell'Università di Bologna

3.4 Validazione on-line

Per la validazione on-line si è deciso di utilizzare la terza wavelet della famiglia Daubechies per la DWT, con lunghezza di sovrapposizione della finestra pari a mezzo intervallo. Come finestra per l'estrazione dei coefficienti si è scelto un intervallo di lunghezza 192 ms

(192 campioni), invece di 200 ms poiché il blocco che esegue la trasformata accetta unicamente intervalli che siano multipli di 2^N dove N è pari al livello di decomposizione, nel nostro caso $N = 4$, quindi unicamente multipli di 16.

Per la validazione on-line si è collegata l'uscita del classificatore al comando per i gesti della mano. Così facendo se l'uscita del classificatore fosse corrisposta ad un gesto dei 6 la mano UB hand IV lo avrebbe replicato; nel caso, invece, l'uscita fosse stata la posa neutrale la mano avrebbe mantenuto il gesto che faceva in precedenza.

Il protocollo utilizzato per questo processo di validazione è stato il seguente: l'utente, collegato alla scheda per l'acquisizione, eseguiva una sequenza casuale dei 6 gesti, mentre la mano cercava di replicarlo; prima e dopo ogni gesto eseguiva una posa neutrale e dopo ogni micro sequenza neutrale-gesto-neutrale la mano veniva messa in posizione di home.

Sequenze di gesti					
5	3	6	2	1	4
4	1	6	5	2	3
2	1	5	3	6	4
1	3	4	5	2	6
3	4	5	2	6	1
6	3	1	5	2	4
1	4	3	6	2	5
3	6	1	5	4	2
3	2	4	1	5	6
4	1	6	5	3	2

Tabella 1

3.5 Risultati

Nella classificazione off-line come detto si sono valutate 3 diverse finestre per l'estrazione: 100 ms, 200 ms e 400 ms. Per ognuna di queste sono state provate 9 wavelets. Queste valutazioni sono poi state ripetute con gli stessi dati, sovrapponendo le finestre di estrazione per una lunghezza pari a metà della loro lunghezza. Così facendo sono state effettuate in totale 18 validazioni off-line. Per valutare quali combinazioni finestra-wavelet sono risultate più performanti si sono ottenute le *matrix of occurrence rate* dalle matrici di confusione. In queste matrici sono stati rappresentati i risultati delle matrici di confusione

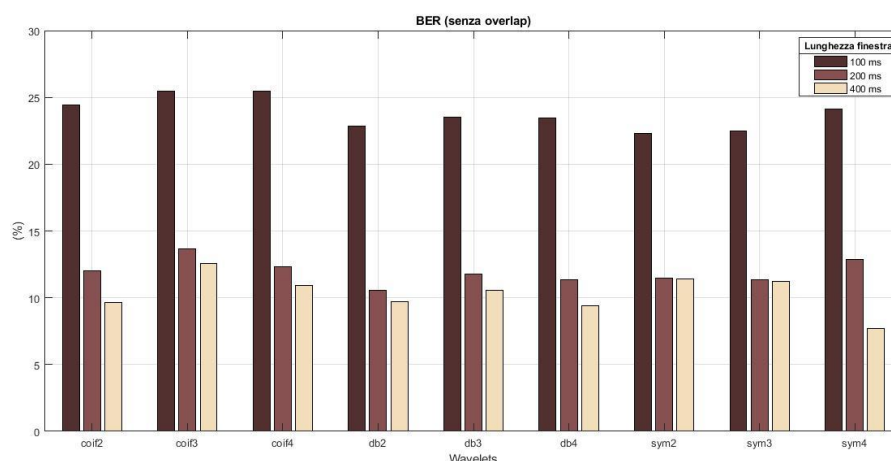


Grafico 1 Balanced error rate delle varie combinazioni senza overlap

in percentuale sul totale riga per riga. Per poi avere un confronto più pratico si sono calcolati e mappati assieme tutti i balanced error rate per ogni combinazione.

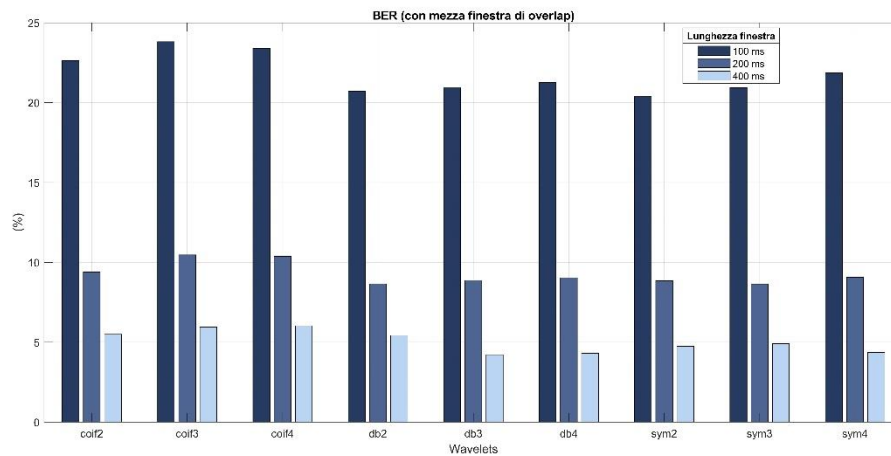


Grafico 2 Balanced error rate delle varie combinazioni con mezza finestra di overlap

Da una rapida analisi si può facilmente notare come all'aumentare delle dimensioni della finestra aumenti anche la precisione del classificatore e che avere la sovrapposizione delle finestre migliori complessivamente le performance dei classificatori. Per quanto riguarda l'intervallo da 400 ms, quello con valori di BER più bassi, si nota che la wavelet migliore risulta essere la Daubechies 3. La sua matrix of occurrence rate è quella in figura 20

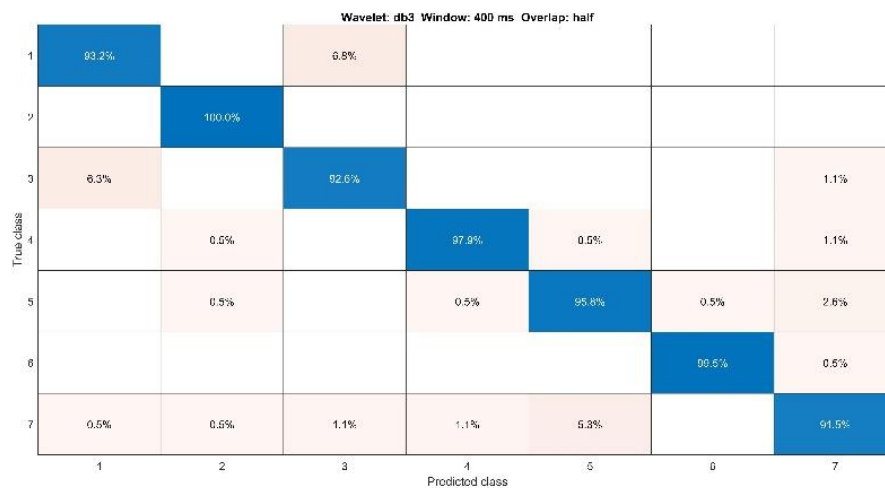


Fig. 20 Matrix of occurrence rate della classificazione fatta con wavelet db3, finestra di 400 ms e overlap presente.

Nonostante gli ottimi risultati ottenuti in questo caso, per la validazione on-line si è deciso di non utilizzare una finestra di analisi così grande (400 ms) perché comportava un ritardo troppo elevato della risposta del classificatore e conseguente replicazione del gesto da parte della mano robotica. Si è scelta quindi una finestra di 200 ms. Nonostante questo, il BER ottenuto con la wavelet Daubechies 3 con questa lunghezza di finestra era comunque paragonabile a quello corrispondente ad una finestra di 400 ms.



Fig. 21 Matrix of occurrence rate della classificazione fatta con wavelet db3, finestra di 200 ms e overlap presente.

Per quanto riguarda la validazione on-line, come già detto al paragrafo 3.3, si è cercata di eseguire una serie di sequenze casuali dei 6 gesti prestabiliti: mano aperta (open hand), presa di potenza (power grasp), tre dita aperte (three fingers), presa ulnare (ulnar pinch), presa di precisione (precision grip) e indice alzato (index pointing). Ognuno dei gesti nelle sequenze era anticipato e seguito da una posa neutrale. Si è andato a valutare quindi se il gesto eseguito fosse replicato dalla mano e se anche le posizioni neutrali (non riprodotte dalla mano con un gesto specifico, ma soltanto mantenendo la posizione precedente) fossero rilevate adeguatamente.

Evidenziando quanti errori di classificazione sono stati fatti per gesto, i risultati ottenuti sono quindi quelli riportati in tab.2. Si può notare come le performance siano diminuite rispetto ai risultati ottenuti off-line, ma nonostante questo si è ottenuto un buon grado di riconoscimento da parte del classificatore, in particolare per quanto riguarda le posizioni PW e Rest. La posizione IP è risultata di più difficile catalogazione rispetto alle altre.

Gesto	Accuratezza (%)
<i>Mano aperta (OH)</i>	70%
<i>Presa di potenza (PW)</i>	100%
<i>Tre dita aperte (3F)</i>	70%
<i>Presa ulnare (UP)</i>	70%
<i>Presa di precisione (PG)</i>	70%
<i>Indice alzato (IP)</i>	50%
<i>Posa neutrale (Rest)</i>	91,7%

Conclusioni

Si è generato uno schema per la classificazione di gesti della mano attraverso segnali EMG allo scopo di controllare mano robotiche. Per fare questo si è approfondito come si generasse questo tipo di segnale e come operasse al fine di trattarlo ed estrarre le informazioni migliori per il lavoro di elaborazione del classificatore. Si è studiato il potente strumento che è la trasformata wavelet e come questa possa essere applicata per l'estrazione di feature del segnale EMG. Dopo questa prima fase di indagine, è stato elaborato uno schema per la classificazione di 6 gesti (7 con la posa neutrale) che applicasse i concetti appresi e che potesse funzionare sia per validazioni off-line che on-line. In particolare, nel caso on-line si è testato che lo schema funzionasse anche assieme al controllo di una mano robotica. Sono stati evidenziati i risultati ottenuti off-line per i vari valori di larghezza della finestra di estrazione dei coefficienti e per le varie wavelets utilizzate. Si è poi fatta una sessione on-line per avere una validazione che comprendesse il vero scopo del lavoro di tesi e si è riportato come si mantengano buoni livelli di accuratezza nonostante si evidenzino un calo delle prestazioni rispetto all'off-line.

Bibliografia

- Meattini, R., Benatti, S., Scarcia, U., Benini, L., & Melchiorri, C. (2015, December). Experimental evaluation of a sEMG-based human-robot interface for human-like grasping tasks. In *Robotics and Biomimetics (ROBIO), 2015 IEEE International Conference on* (pp. 1030-1035). IEEE.
- Meattini, R., Palli, G., & Melchiorri, C. (2017, July). Experimental evaluation of a sEMG-based control for elbow wearable assistive devices during load lifting tasks. In *Rehabilitation Robotics (ICORR), 2017 International Conference on* (pp. 140-145). IEEE.
- Englehart, K., Hudgin, B., & Parker, P. A. (2001). A wavelet-based continuous classification scheme for multifunction myoelectric control. *IEEE Transactions on Biomedical Engineering*, 48(3), 302-311.
- Lucas, M. F., Gaufriau, A., Pascual, S., Doncarli, C., & Farina, D. (2008). Multi-channel surface EMG classification using support vector machines and signal-based wavelet optimization. *Biomedical Signal Processing and Control*, 3(2), 169-174.
- Polikar, R. (1996). The wavelet tutorial.
- Mallat, S. G. (1989). A theory for multiresolution signal decomposition: the wavelet representation. *IEEE transactions on pattern analysis and machine intelligence*, 11(7), 674-693.
- Daubechies, I. (1992). *Ten lectures on wavelets* (Vol. 61). Siam.
- Meattini, R. (2017). Electromyography Based Human-Robot Interfaces for the Control of Artificial Hands and Wearable Devices
- Burges, C. J. (1998). A tutorial on support vector machines for pattern recognition. *Data mining and knowledge discovery*, 2(2), 121-167.
- Chang, C. C., & Lin, C. J. (2011). LIBSVM: a library for support vector machines. *ACM transactions on intelligent systems and technology (TIST)*, 2(3), 27.
- Verdoliva L. (2014). La Trasformata Wavelet. *Appunti di Elaborazione dei Segnali Multimediali*