



Article

A Comparative Study of Computational Methods for Compressed Sensing Reconstruction of EMG Signal

Lorenzo Manoni [†] , Claudio Turchetti ^{*,†}, Laura Falaschetti [†] and Paolo Crippa [†] DII—Dipartimento di Ingegneria dell'Informazione, Università Politecnica delle Marche,
Via Brecce Bianche 12, I-60131 Ancona, Italy

* Correspondence: c.turchetti@univpm.it; Tel.: +39-071-220-4848

† These authors contributed equally to this work.

Received: 12 July 2019; Accepted: 9 August 2019; Published: 13 August 2019



Abstract: Wearable devices offer a convenient means to monitor biosignals in real time at relatively low cost, and provide continuous monitoring without causing any discomfort. Among signals that contain critical information about human body status, electromyography (EMG) signal is particularly useful in monitoring muscle functionality and activity during sport, fitness, or daily life. In particular surface electromyography (sEMG) has proven to be a suitable technique in several health monitoring applications, thanks to its non-invasiveness and ease to use. However, recording EMG signals from multiple channels yields a large amount of data that increases the power consumption of wireless transmission thus reducing the sensor lifetime. Compressed sensing (CS) is a promising data acquisition solution that takes advantage of the signal sparseness in a particular basis to significantly reduce the number of samples needed to reconstruct the signal. As a large variety of algorithms have been developed in recent years with this technique, it is of paramount importance to assess their performance in order to meet the stringent energy constraints imposed in the design of low-power wireless body area networks (WBANs) for sEMG monitoring. The aim of this paper is to present a comprehensive comparative study of computational methods for CS reconstruction of EMG signals, giving some useful guidelines in the design of efficient low-power WBANs. For this purpose, four of the most common reconstruction algorithms used in practical applications have been deeply analyzed and compared both in terms of accuracy and speed, and the sparseness of the signal has been estimated in three different bases. A wide range of experiments are performed on real-world EMG biosignals coming from two different datasets, giving rise to two different independent case studies.

Keywords: compressed sensing; signal reconstruction; surface electromyography; biosignal; sensors; wireless sensor networks

1. Introduction

Surface electromyography (sEMG) is a technique to capture and measure the electrical potential at the skin surface due to muscle activity [1,2]. The registered EMG signal in a muscle is the collective action potential of all muscular fibers of the motor unit that work together since they are stimulated by the same motor neuron. The muscular contraction is generated by a stimulus that propagates from the brain cortex to the target muscle as an electrical potential, named action potential (AP). sEMG signal is frequently used for the evaluation of muscle functionality and activity, thanks to the non-invasiveness and ease of this technique [3–5]. Common applications are fatigue analysis [6] of rehabilitation exercises [5,7], postural control [8], musculoskeletal disorder analysis [9], gait analysis [10], movement recognition [11], gesture recognition [12], prosthetic control [13–15], to cite only a few. Among these applications monitoring and automatic recognition of human activities are of particular interests both for sport and fitness as well as for healthcare of elderly and impaired people [16,17]. Wireless body area

networks (WBANs) provide an effective and a relatively low-cost solution for biosignal monitoring in real time [18,19]. A WBAN typically consists of one or more low-power, miniaturized, lightweight devices with wireless communication capabilities that operate in the proximity of a human body [20]. However, power consumption represents a major problem for the design and for the widespread of such devices. A large part of the device power consumption is required for the wireless transmission of the signals which are recorded from multiple channels at a high-sampling rate [21]. Standard compression protocols have a high computational complexity and the implementation in the sensor nodes would add a big overhead to the power consumption. Compressed sensing (CS) techniques that lie on the sparsity property of many natural signals, have been successfully applied in the WBAN long term signal monitoring, since CS significantly saves the transmit power by reducing the sampling rate [22–26]. Recent studies have applied CS to sEMG signals for gesture recognition, an innovative application field of the sEMG signal analysis [27,28]. In this context, CS has a great importance in reducing the size of transmitted sEMG data while being able to reconstruct good quality signals and recognize hand movements. The fundamental idea behind CS is that rather than first sampling at a high rate and then compressing the sampled data, as usually done in standard techniques, we would like to find ways to directly sense the data in compressed form, i.e., at a lower sampling rate. To make this possible, CS relies on the concept of sparsity which implies that certain classes of signals, sparse signals, when expressed in a proper basis have only a small number of non-zero coordinates. The CS field grew out of the seminal work of Candes, Romberg, Tao and Donoho [29–35], who showed that a finite dimensional sparse signal can be recovered from several samples much smaller than its length. The CS paradigm combines two fundamental stages, encoding and reconstruction. The reconstruction of a signal acquired with CS represents the most critical and expensive stage as it involves an optimization which seeks the best solution to an undetermined set of linear equations with no prior knowledge of the signal except that it is sparse when represented in a proper basis. To obtain the better performance in the reconstruction of the undersampled signal a large variety of algorithms have been developed in recent years [36]. In the class of computational techniques for solving sparse approximation problems, two approaches are computationally practical and lead to provably exact solutions under some defined conditions: convex optimization and greedy algorithms. Convex optimization is the original CS reconstruction algorithm formulated as a linear programming problem [37]. Unlike convex optimization, greedy algorithms try to solve the reconstruction problem in a less exact manner. In this class, the most common algorithms used in practical applications are orthogonal matching pursuit (OMP) [38–42], compressed sampling matching pursuit (CoSaMP) [43,44], normalized iterative hard thresholding (NIHT) [45]. All these algorithms are applicable in principle to a generic signal; however, in the design and implementation of a sensor architecture is of paramount importance to assess the performance with reference to the specific signal to be acquired. Additionally, the performance of the algorithms can vary very widely, so that a comparative study that demonstrates the practicability of such algorithms are welcomed by designers of low powers WBANs for biosignal monitoring [26].

The aim of this paper is to explore the trade-off in the choice of a compressed sensing algorithm, belonging to the classes of techniques previously described, to be applied in EMG sensor-applications. Thus, the ultimate goal of the paper is to present a comparative study of computational methods for CS reconstruction of EMG signals, in real-world EMG signal acquisition systems, leading to efficient, low-power WBANs. For example, a useful application of this comparative study can be the selection of the best algorithm to be applied in EMG-based gesture recognition. In addition, the effect of this basis used for reconstruction on signal sparseness has been analyzed for three different bases.

This paper is organized as follows. Section 2 summarizes the basic concept of CS theory. Section 3 is mainly focused on CS reconstruction algorithms and in particular gives a complete description of four algorithms: Convex Optimization, OMP, CoSaMP, and NIHT. Section 4 reports a comparative study of the four algorithms performance when applied to real-world EMG signals.

2. Compressed Sensing Background

In this section, we provide an overview of the basic concepts of the CS theory. In Table 1, for ease of reference, a list of the main symbols and definitions used throughout the text are reported. Some of these are currently adopted in the literature while other specific operators will be defined later.

Table 1. Notation.

Symbol	Description
\odot	element-wise product of two vectors, i.e., $c = a \odot b = (a_1 b_1, \dots, a_N b_N)$, $a = (a_1, \dots, a_N)$, $b = (b_1, \dots, b_N)$
\oplus	bitwise XOR between two binary arrays
$[x]_{s_r}$	s_r samples circular shift of vector x
$\text{sgn}(x)$	element-wise sign function of a vector x
B^\dagger	pseudo-inverse of matrix B
$\Lambda = \text{supp}(x)$	support of x , the set of indices $\Lambda = \{j : x_j \neq 0\}$
$ \Lambda = k$	cardinality of the set Λ (the number k of elements in the set)
$\ x\ _0 = \text{supp}(x) $	l_0 -norm of x
$\ x\ _p = (\sum_{i=1}^n x_i ^p)^{1/p}$	l_p -norm of x (for some $0 < p < \infty$)
x_Λ	sub-vector of x indexed by set Λ
B_Λ	sub-matrix of B made by columns indexed by set Λ
$\text{supp}_{k,\Psi}(x)$	returns a set Λ of k indexes corresponding to the largest values $ x_i \ \Psi_i\ _2$, $\Psi = [\Psi_1, \dots, \Psi_N]$
$F(x, \Lambda)$	returns a vector with the same elements of x in the sub-set Λ and 0 elsewhere
$[x]_k = F(x, \text{supp}_{k,\Psi}(x))$	reduced operator

CS theory asserts that rather than acquire the entire signal and then compress, it is usually done in compression techniques, it is possible to capture only the useful information at rates smaller than the Nyquist sampling rate.

The CS paradigm combines two fundamental stages, encoding and reconstruction.

In the encoding stage the N -dimensional input signal f is encoded into a M -dimensional set of measurements y through a linear transformation by the $M \times N$ measurement matrix Φ where $y = \Phi f$. In this way with $M < N$ the CS data acquisition system directly translates analog data into a compressed digital form.

In the reconstruction stage given by $f = \Psi x$, assuming the signal f to be recovered is known to be sparse in some basis $\Psi = [\Psi_1, \dots, \Psi_N]$, in the sense that all but a few coefficients x_i are zero, the sparsest solution x (fewest significant non-zero x_i) is found. The reconstruction algorithms exhibit better performance when the signal to be reconstructed is exactly k -sparse on the basis Ψ , i.e., with $x_i \neq 0$ for $i \in \Lambda$, $|\Lambda| = k$. Thus, in some algorithms the $N - k$ elements of x that give negligible contributions are discarded. To this end the following operator is defined

$$\Lambda = \text{supp}_{k,\Psi}(x) : |\Lambda| = k, \gamma_i > \gamma_j, \gamma_i = |x_i| \|\Psi_i\|_2, \text{ for } i \in \Lambda, j \notin \Lambda \quad (1)$$

that selects the set Λ of k indexes corresponding to largest values $|x_i| \|\Psi_i\|_2$. The set Λ so derived represents the so-called *set of sparsity*. Another useful definition in this context is the operator $F(x, \Lambda)$ that returns a vector with the same elements of x in the sub-set Λ and zero elsewhere, formally

$$F(x, \Lambda)_\Lambda = x_\Lambda, F(x, \Lambda)_{I_N \setminus \Lambda} = 0, I_N = \{1, 2, \dots, N\}, \quad (2)$$

where $I_N \setminus \Lambda$ means difference of the two sets I_N and Λ . The consecutive application of the two operators gives rise to a k -sparse vector obtained from x by keeping only the components with the largest values of $|x_i| \|\Psi_i\|_2$, and will be synthetically denoted by $[x]_k$, called *reduced operator*. Thus

$$[x]_k = F(x, \text{supp}_{k,\Psi}(x)) \quad (3)$$

A natural formulation of the recovery problem is within an l_0 -norm minimization framework, which seeks a solution x of the problem

$$\min_{x \in \mathbb{R}^N} \|x\|_0 \quad \text{subject to } y = \Phi \Psi x, \quad (4)$$

where $\|\cdot\|_0$ is a counting function that returns the number of non-zero components in its argument. Unfortunately, the l_0 -minimization problem is NP-hard, and hence cannot be used for practical application. A method to avoid using this computationally intractable formulation is to consider an l_1 -minimization problem.

It has been shown [33] that when x is the solution to the convex approximation problem

$$\min_{x \in \mathbb{R}^N} \|x\|_1 \quad \text{subject to } y = \Phi \Psi x \quad (5)$$

then the reconstruction $f = \Psi x$ is exact. More specifically only M measurements in the Φ domain selected uniformly at random, are needed to reconstruct the signal provided M satisfies the inequality

$$M \geq C \nu^2(\Phi, \Psi) S \log N \quad (6)$$

where N represents the signal size, S the index of sparseness, C a constant and $\nu(\Phi, \Psi)$ the coherence between the sensing basis Φ and the representation basis Ψ . Coherence measures the largest correlation between any two elements of Φ and Ψ and is given by $\nu(\Phi, \Psi) = \max_{k,j} |\langle \Phi_k, \Psi_j \rangle|$, with $\nu^2(\Phi, \Psi) \in [0, N]$. Random matrices are largely incoherent ($\nu = 1$) with any fixed basis Ψ . Therefore, as the smaller the coherence the fewer samples are needed, random matrices are the best choice for sensing basis.

The usually adopted performance metric to measure the reduction in the data required to represent the signal f is the compression ratio CR defined as

$$CR = \frac{N}{M}, \quad (7)$$

that is the ratio between the length of the original and compressed signal vectors. Instead sparsity is usually defined as

$$S_N = \frac{k}{N}. \quad (8)$$

Sometimes is more convenient to define sparsity with respect to dimension M , thus giving $S_M = k/M$. Obviously the two equation are related by $S_M = CR S_N$.

3. The Algorithms

As the CS sampling framework includes two main activities, encoding and reconstruction, some specific algorithms must be derived for this purpose.

3.1. Encoding

CS encoder uses a linear transformation to project the vector f into the lower dimensional vector y , through the measurement matrix Φ . In addition of being incoherent with respect to the basis Ψ , measurement matrix Φ must facilitate encoding practical implementation. One widely used approach is to use Bernoulli random matrices $\Phi(i, j) = \pm 1$. This choice allows saving of multiplication in the matrix-product operation $y = \Phi f$. Moreover, simple, fast and low-power digital and analog hardware implementations of the encoder are possible [26].

3.2. Basis Matrix Ψ

A wide range of basis matrices Ψ can be adopted in Equation (4), three of the most familiar bases will be used in this paper, namely DCT, Haar and Daubechies' wavelet (DBW). Although DCT seems

not to be an adequately sparse basis for EMG signal it was used in one of the two case studies because of signal pre-filtering during acquisition as it will be explained in Section 4. Additionally, other recent works [46,47] have demonstrated the validity of DCT basis for CS applied to EMG signal. Matrix Ψ for Haar and DB4 was built using parallel filter bank wavelet implementation [48].

3.3. Reconstruction

CS reconstruction algorithms can be divided into two classes: convex optimization and greedy algorithms.

3.3.1. Convex Optimization

L1-minimization

The CS theory asserts that when f is sufficiently sparse, the recovery via l_1 -minimization is provably exact. Thus, a fundamental algorithm for reconstruction is the convex optimization wherein the l_1 -norm of the reconstructed signal is an efficient measure of sparsity. The CS reconstruction process is described by Equation (5) which can be regarded as a linear programming problem. This approach is also known as basis pursuit.

By assuming $f(k) = [f(kN - N + 1), \dots, f(kN)]$, $k = 1, \dots, L$ is a frame of the EMG signal to be reconstructed, $\Psi = [\Psi_1, \dots, \Psi_N]$ is an $N \times N$ basis matrix, Φ an $M \times N$ Bernoulli matrix, then the constraint in Equation (5) can be rewritten as

$$y = \Phi\Psi x = Ax \quad (9)$$

with $A = \Phi\Psi$. Introducing the Lagrange function

$$L(x, \lambda) = \|x\|_1 + \lambda^T(Ax - y) \quad (10)$$

where T denotes matrix transposition thus to solve the problem (5) is equivalent to determine the stationary point of $L(x, \lambda)$ with respect to both x and λ . A usual technique for this problem is the projected-gradient algorithm based on the following iterative scheme

$$x_{t+1} = x_t - \mu \left. \frac{\partial L}{\partial x} \right|_{x_t} \quad (11)$$

where μ is a parameter that regulates the convergence of the algorithm. By deriving (10) we obtain

$$\frac{\partial L}{\partial x} = \text{sgn}(x) + A^T \lambda \quad (12)$$

then combining Equations (11) and (12) with constraint $Ax_t = y$ and assuming $(AA^T)^{-1}$ is invertible, it results

$$\lambda = -(AA^T)^{-1} A \text{sgn}(x_t) \quad (13)$$

Finally, the following iterative solution

$$x_{t+1} = x_t - \mu(I - A^T(AA^T)^{-1}A) \text{sgn}(x_t) \quad (14)$$

is obtained. To make the convergence parameter μ independent of signal power the following normalized version of the algorithm can be adopted

$$x_{t+1} = x_t - \mu \frac{\|x_t\|_1}{N} P \text{sgn}(x_t) \quad (15)$$

with $P = I - A^\dagger A$ and $A^\dagger = A^T(AA^T)^{-1}$. To initialize the algorithm a vector x_0 given by

$$x_0 = A^\dagger y \quad (16)$$

that solves the following l_2 -minimization problem

$$x_0 = \arg \min_x \|x\|_2^2 \text{ s.t. } Ax = y \quad (17)$$

has been chosen. The parameter μ determines the convergence of the algorithm; thus, to establish a proper choice of its value a convergence criterion should be derived. However, a complete treatment of convergence is a difficult task and is out of the scope of this paper. To face this problem the value of μ has been chosen using a semi-heuristic criterion that bounds the steady-state ripple given by

$$\frac{\|x_{t+1} - x_t\|_2}{\|x_t\|_2} < \epsilon_{max}, \quad t \rightarrow \infty \quad (18)$$

where ϵ_{max} specifies the desired accuracy. In such a way we obtain

$$\mu \leq \frac{\epsilon_{max} N}{\|P \operatorname{sgn}(x_\infty)\|_2} \frac{\|x_\infty\|_2}{\|x_0\|_1} < \frac{\epsilon_{max} N}{\|P \operatorname{sgn}(x_\infty)\|_2} \quad (19)$$

which can be reduced to the more practical condition

$$\mu \leq \frac{\epsilon_{max} N}{\|P \operatorname{sgn}(x_0)\|_2} \quad (20)$$

An optimized version of the algorithm with a reduced number of products can be derived as follows. Let us rewrite Equation (15) as

$$x_{t+1} = x_t - \mu \frac{\|x_t\|_1}{N} q_t \quad (21)$$

where

$$q_t = P_{s_t} = \sum_{j=1}^N P_j s_t(j) \quad (22)$$

and

$$s_t = (s_t(1), \dots, s_t(N)) = \operatorname{sgn}(x_t) \quad (23)$$

The variation of q from $(t-1)$ to t

$$q_t - q_{t-1} = \sum_{j=1}^N P_j [s_t(j) - s_{t-1}(j)] \quad (24)$$

only depends on

$$\Delta s_t(j) = s_t(j) - s_{t-1}(j) = \begin{cases} 2 & \text{for } s_{t-1}(j) = -1, s_t(j) = 1 \\ -2 & \text{for } s_{t-1}(j) = 1, s_t(j) = -1 \\ 0 & \text{for } s_{t-1}(j) = s_t(j) \end{cases} \quad j = 1, \dots, N, \quad (25)$$

which can be rewritten in a compact form as

$$\Delta s_t(j) = 2s_t(j)v_t(j), \quad j = 1, \dots, N \quad (26)$$

where

$$v_t(j) = \begin{cases} 1 & \text{for } s_t(j) \neq s_{t-1}(j) \\ 0 & \text{for } s_t(j) = s_{t-1}(j) \end{cases} \quad (27)$$

By defining

$$w_t(j) = (s_t(j) + 1)/2 \in \{0, 1\}, \quad (28)$$

Equation (27) is equivalent to

$$v_t(j) = w_t(j) \oplus w_{t-1}(j) \quad . \quad (29)$$

Finally, from Equations (24), (26) and (29) and defining the set $\Omega_t = \{j/v_t(j) = 1\}$ we have

$$q_t = q_{t-1} + 2 \sum_{j \in \Omega_t} P_j s_t(j) \quad . \quad (30)$$

Thus, the summation in Equation (30) is extended only to the terms for which a sign changing from s_{t-1} to s_t occurs, thus reducing the number of products required at each step.

A pseudo-code of the L_1 algorithm is reported as Algorithm 1.

Algorithm 1 L_1 -minimization

Input: $A = \Phi\Psi$, y , k

Initalize:

$$\begin{cases} P = [p_1 \dots p_N], P = I - A^\dagger A, \\ x_0 = A^\dagger y \\ t = 0 \end{cases}$$

Output: k -sparse coefficient vector x

while $t < N_{iter}$ **do**

$s_t = \text{sgn}(x_t)$, $w_t = (s_t + 1)/2$ // reduce s_t to binary vector w_t

if $t > 0$ **then**

$v_t = w_t \oplus w_{t-1}$

$\Omega_t = \{j/v_t(j) = 1\}$ // define the set of indices Ω_t corresponding to a change from s_{t-1} to s_t

$q_t = q_{t-1} + 2 \sum_{j \in \Omega_t} p_j s_t(j)$

else

$q_t = P s_0$

$\mu = \frac{\epsilon_{max} N}{\|q_0\|_2}$

end if

$x_{t+1} = x_t - \mu \frac{\|x_t\|_1}{N} q_t$

$t = t + 1$

end while

3.3.2. Greedy Algorithms

Orthogonal Matching Pursuit (OMP)

This algorithm solves the reconstruction of k -sparse coefficient vector x , i.e., with $x_i \neq 0$, $i \in \Lambda$, $|\Lambda| = k$. The algorithm tries to find the k directions corresponding to the non-zero x -components, starting from the residual r_0 given by the measurements y . At each step $t \leq k$ the column a_j of A that is mostly correlated with the residual is derived. Then the best coefficients x_t are found by solving the following l_2 -minimization problem

$$x_t = \arg \min_x \|y - A_t x\|_2, \quad (31)$$

thus giving

$$x_t = A_t^\dagger y \quad . \quad (32)$$

Finally, the residual, the difference between the actual measure and the A mathematical description of the algorithm is reported in Algorithm 2.

Algorithm 2 OMP

Input: $A = \Phi\Psi$, y , k
Initialize:

$$\begin{cases} r_0 = y & // \text{residual} \\ A_0 = \emptyset & // \text{columns} \\ t = 0 \end{cases}$$
Output: k -sparse coefficient vector x
while $t < k$ **do**

$$\lambda_t = \arg \max_j |a_j^T r_{t-1}| \quad // \text{find the column of } A \text{ that is most strongly correlated with the residual}$$

$$A_t = [A_{t-1} \ a_{\lambda_t}] \quad // \text{merge the new column}$$

$$x_t = A_t^\dagger y \quad // \text{find the best coefficients } x_t \text{ from (31)}$$

$$r_t = y - A_t x_t \quad // \text{update the residual}$$

$$t = t + 1$$
end while

Compressive Sampling Matching Pursuit (CoSaMP)

Differently from OMP, the CoSaMP algorithm tries to find the k_S columns of A^T that are the most strongly correlated with the residual, thus making a correction of the reconstruction on the basis of residual achieved at each step. The k_S columns are determined by the selection step

$$W_t = \arg \max_{|W|=k_S} \|A_W^T r_t\|_1 \quad (33)$$

where A_W^T is the sub-matrix of A^T made by columns indexed by the set W , and r_t is the residual at current iteration step t . Thus, the algorithm proceeds to estimate the best coefficients h for approximating the residual with the new columns indexed by $T_t = \Lambda_t \cup W_t$. As this step corresponds to an LMS problem, it results

$$h = A_{T_t}^\dagger y \quad (34)$$

Finally, the sparsity operator

$$x = F(h, \Lambda_{t+1}) \quad (35)$$

with $\Lambda_{t+1} = \underset{k, \Psi}{\text{supp}}(h)$ is applied to obtain the sparse vector x . At the end of the algorithm the residual is updated with the new signal reconstruction $A_{\Lambda_{t+1}} x_{\Lambda_{t+1}}$. A pseudo-code of the algorithm is reported in Algorithm 3.

Normalized Iterative Hard Thresholding (NIHT)

The basic idea that underlies NIHT algorithm is that the sparse components to be identified give a large contribution to the gradient of residual. The algorithm tries to find these components by following the gradient of residual $r_t = Ax_t - y$, i.e.,

$$\tilde{x}_{t+1} = x_t - \mu_t \frac{\partial \|r_t\|_2^2}{\partial x_t}, \quad (36)$$

thus obtaining

$$\tilde{x}_{t+1} = x_t - \mu_t A^T (Ax_t - y) \quad (37)$$

The sparse vector x_{t+1} is derived at each iteration t by applying the reduced operator to the estimated vector x_{t+1} ,

$$\Lambda_{t+1} = \underset{k, \Psi}{\text{supp}}(\tilde{x}_{t+1}) \quad (38)$$

$$x_{t+1} = F(\tilde{x}_{t+1}, \Lambda_{t+1}) \quad (39)$$

Algorithm 3 CoSaMP

Input: $A = \Phi\Psi$, y , k
Initialize:
 $\begin{cases} r_0 = y & // \text{residual} \\ t = 0 \end{cases}$
 $\Lambda_0 = \arg \max_{|\Lambda|=k} \|A_{\Lambda}^T r_0\|_1$ // find k columns of A^T that are most strongly correlated with residual r_0
Output: k -sparse coefficient vector x
while $t < N_{iter}$ **do**
 $k_S = \gamma k$, $\gamma \in [0, 1]$ // number of new columns to be selected
 $W_t = \arg \max_{|W|=k_S} \|A_W^T r_t\|_1$ // find k_S columns of A^T that are most strongly correlated with residual r_t
 $T_t = \Lambda_t \cup W_t$ // merge the new columns such that $|T_t| = k + k_S$
 $h = A_{T_t}^{\dagger} y$ // find the best coefficients for residual approximation
 $\Lambda_{t+1} = \underset{k, \Psi}{\text{supp}}(h)$ // find the set of sparsity Λ_{t+1}
 $x = F(h, \Lambda_{t+1})$ // find sparse vector x
 $r_{t+1} = y - A_{\Lambda_{t+1}} x_{\Lambda_{t+1}}$ // update the residual
 $t = t + 1$
end while

As in CoSaMP the initialization is made by choosing the columns of A^T that are most strongly correlated with residual

$$\Lambda_0 = \arg \max_{|\Lambda|=k} \|A_{\Lambda}^T y\|_1 \quad (40)$$

and then estimating the best coefficients

$$x_0 = A_{\Lambda_0}^{\dagger} y \quad (41)$$

for residual approximation. A different step size has been used for each x_t component by defining the step size vector ρ as

$$\rho = \min_j \|a_j\|_2 \left(\frac{1}{\|a_1\|_2}, \dots, \frac{1}{\|a_N\|_2} \right), \quad (42)$$

thus, normalizing the components of the gradient vector $A^T(Ax_t - y)$. In this way the updated equation for x becomes

$$\tilde{x}_{t+1} = x_t - \mu_t q_t \quad (43)$$

where $q_t = \rho \odot A^T(Ax_t - y)$ is the normalized gradient vector and \odot denotes the element-wise product of vectors. The value of μ_t has been estimated by minimizing the residual, i.e., such that

$$\frac{\partial}{\partial \mu_t} \|Ax_{t+1} - y\|_2^2 = 0 \quad (44)$$

or

$$\frac{\partial}{\partial \mu_t} \|A(x_t - \mu_t q_t) - y\|_2^2 = 0 \quad (45)$$

A closed form of μ_t cannot be derived as it depends on the set Λ_{t+1} selected after the updating of \tilde{x}_{t+1} . To circumvent this problem an iterative approach has been used, starting from an initial estimation Λ_{t+1}^* of Λ_{t+1} to compute $\mu_t(\Lambda_{t+1}^*)$ and then updating Λ_{t+1}^* to the true value. In this way from previous Equation (45) we obtain

$$\mu_t = \frac{q_{\Lambda_{t+1}^*}^T A^T (Ax_{\Lambda_{t+1}^*} - y)}{q_{\Lambda_{t+1}^*}^T A^T A q_{\Lambda_{t+1}^*}} = \frac{w^T \epsilon}{w^T w} \quad (46)$$

where

$$\begin{aligned} w &= Aq_{\Lambda_{t+1}^*} \quad \epsilon = Ax_{\Lambda_{t+1}^*} - y \\ \Lambda_{t+1}^* &= \underset{k, \Psi}{\text{supp}}(x_t - \mu_t q). \end{aligned} \quad (47)$$

Algorithm 4 NIHT

Input: $A = \Phi\Psi$, y , k

Initialize:

$$\left\{ \begin{array}{l} \Lambda_0 = \arg \max_{|\Lambda|=k} \|A_{\Lambda}^T y\|_1 \quad // \text{find the columns of } A^T \text{ that are the} \\ \quad \quad \quad \quad \quad \quad // \text{most strongly correlated with residual} \\ x_0 = A_{\Lambda_0}^+ y \quad // \text{find the best coefficients for residual approximation} \\ t = 0 \end{array} \right.$$

Output: k -sparse coefficient vector x

while $t < N_{iter}$ **do**

$$\begin{aligned} r_t &= Ax_t - y \quad // \text{update residual} \\ \rho &= \min_j \|a_j\|_2 \left(\frac{1}{\|a_1\|_2}, \dots, \frac{1}{\|a_N\|_2} \right) \quad // \text{step size vector} \\ q_t &= \rho \odot (A^T r_t) \quad // \text{normalized gradient vector} \\ \Lambda_{t+1} &= \Lambda_{t+1}^* \quad // \text{initialize the set of sparsity } \Lambda_{t+1} \end{aligned}$$

if $t > 0$ **then**

while (stop criterion on Λ_{t+1}) **do**

$$\begin{aligned} \tilde{x}_{t+1} &= x_t - \mu_t (\Lambda_{t+1}) q_t \quad // \text{update } x_t \text{ with step size } \mu_t \text{ given by (46)} \\ \Lambda_{t+1} &= \underset{k, \Psi}{\text{supp}}(\tilde{x}_{t+1}) \quad // \text{update set of sparsity } \Lambda_{t+1} \\ x_{t+1} &= F(\tilde{x}_{t+1}, \Lambda_{t+1}) \quad // \text{find sparse vector } x_{t+1} \end{aligned}$$

end while

else

$$\begin{aligned} \tilde{x}_{t+1} &= x_t - \mu_t (I_N) q_t \\ [x_{t+1}]_k &= F(\tilde{x}_{t+1}, \underset{k, \Psi}{\text{supp}}(\tilde{x}_{t+1})) \quad // \text{find sparse vector } x_{t+1} \end{aligned}$$

end if

$t = t + 1$

end while

4. Comparative Study

To quantify the performance of the CS algorithms previously described a comparative study has been conducted on two different sets of EMG signals, giving rise to case study A and case study B.

A similar study of CS applied to EMG signal was performed in [49]. In that work sparsity is enforced to the signal with a time-domain thresholding technique, and reconstruction SNR is measured with respect to the sparsified signal. In this work, to have an estimation of overall information loss of the signal, after enforcing sparsity with reduced operator $[x]_k$ for each basis, we measured SNR with respect to the original signal x .

4.1. Case Study A

The signals used in this study refer to three different muscles, namely *biceps brachii*, *deltoideus medius*, and *triceps brachii*. They were recorded by the sEMG acquisition set-up described in [16] and following the protocol outlined in [50,51]. The EMG signal was high-pass filtered at 5 Hz and low-pass

filtered at 500 Hz before being sampled at 2 kHz. The algorithms were applied to frames of length $N = 1024$, which is a large value enough to limit SNR variations among frames. In the simulations the index k and the compression factor $CF = M/N$, i.e., the inverse of CR , were varied. The performance has been measured based on the following equivalent signal to noise ratio

$$SNR = 20 \log_{10} \frac{\|y\|_2}{\|y - y_{rec}\|_2} \quad , \quad (48)$$

where y_{rec} is the reconstruction signal, by averaging the results obtained with different frames.

4.1.1. Basis Selection

Figure 1 compares for the three muscles the reconstruction error in three frames extracted from the data set, as achieved with convex optimization using three different bases, DCT, Haar, and DB4. Since the signal was pre-filtered at 500 Hz, this can lead to an improvement of sparsity in the frequency domain making DCT worth testing.

Figures 2 and 3 report the SNR as a function of frame, sparsity and iteration respectively, for the same muscles of Figure 1. DB4 basis clearly shows the best reconstruction performance in all the conditions considered in these figures.

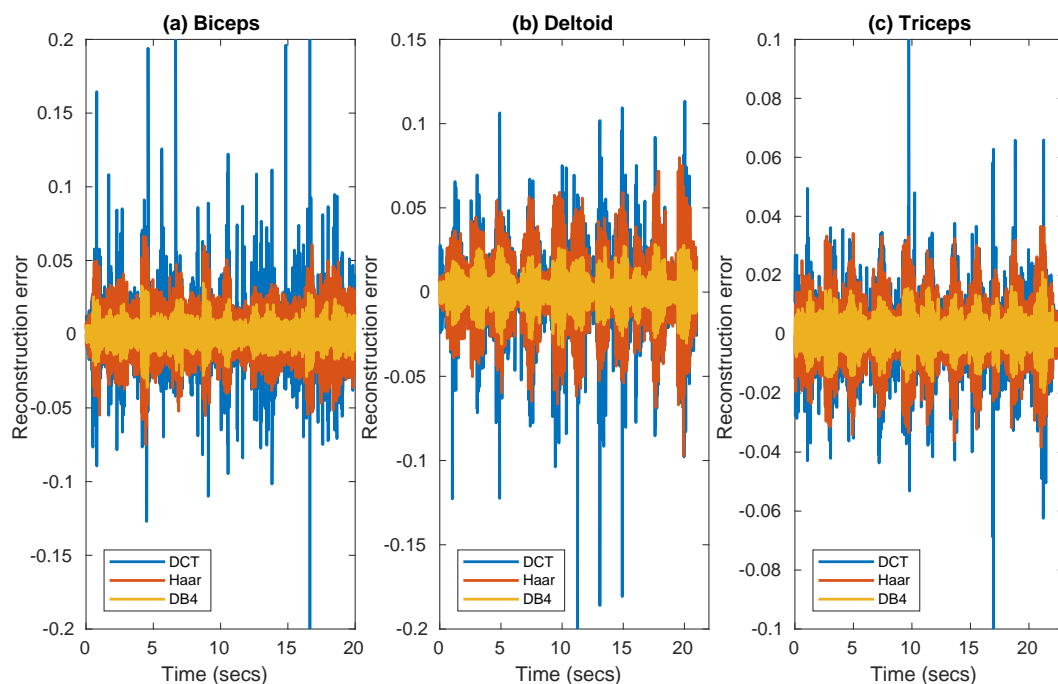


Figure 1. Reconstruction error of EMG signal as achieved with convex optimization, using DCT, Haar, and DB4 bases in frames corresponding to three muscles (a) Biceps (b) Deltoideus (c) Triceps.

4.1.2. Comparison of Algorithms Performance

As the ultimate goal of this paper is to study and compare the CS methods for the reconstruction of EMG signals, a large experimentation has been carried out with the algorithms previously described.

Figures 4–8 report the performance achieved with the four algorithms L1, OMP, CoSaMP, and NIHT under different experimental conditions. In particular the behavior of SNR as a function of sparsity $S_M = k/M$ for the four algorithms and the three bases is shown in Figure 4, where a constant value $CF = 0.5$ of compression factor is used. Here the sparsity S_M with respect to the dimension M has been adopted as for $k > M$ the behavior is not of particular significance. It is evident from these results the superiority of DB4 with respect to other bases as already pointed out in the previously figures. Concerning algorithm performance, all the algorithms show a pronounced peak near the

value of $k/M = 0.4 - 0.5$. This behavior is due to the fact that the SNR is measured with respect to the original signal, and as k/M decreases the fidelity between x and $[x]_k$ deteriorates. Moreover, while for OMP, CoSaMP, and NIHT, the SNR falls rapidly as k/M increases, the L1 algorithm remains nearly constant beyond the maximum.

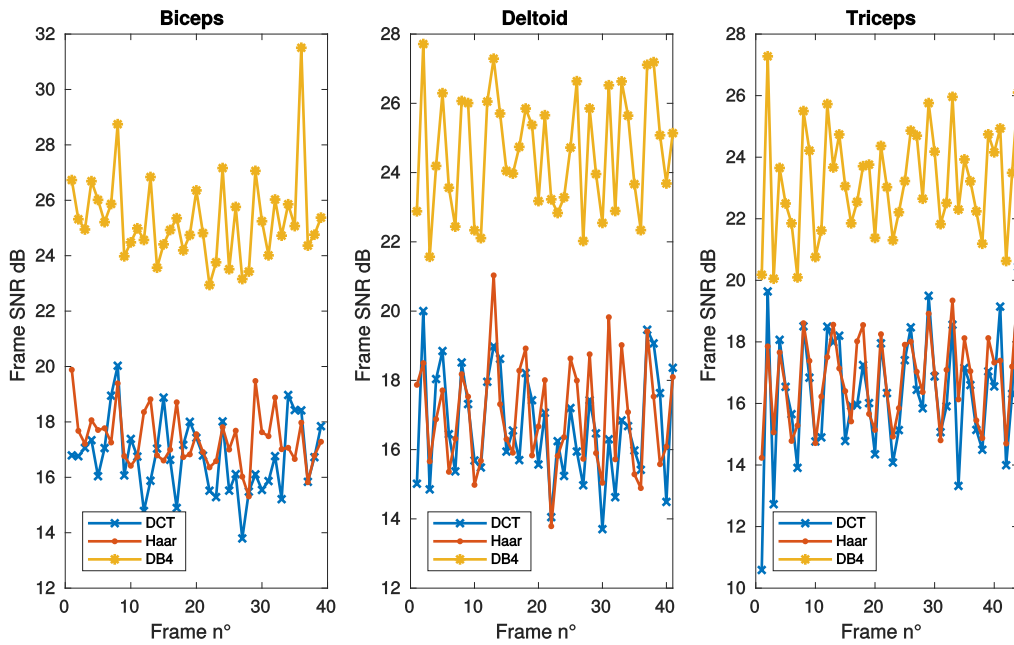


Figure 2. SNR vs. frame number, as achieved with convex optimization, using DCT, Haar, and DB4 bases, for the same muscles of Figure 1.

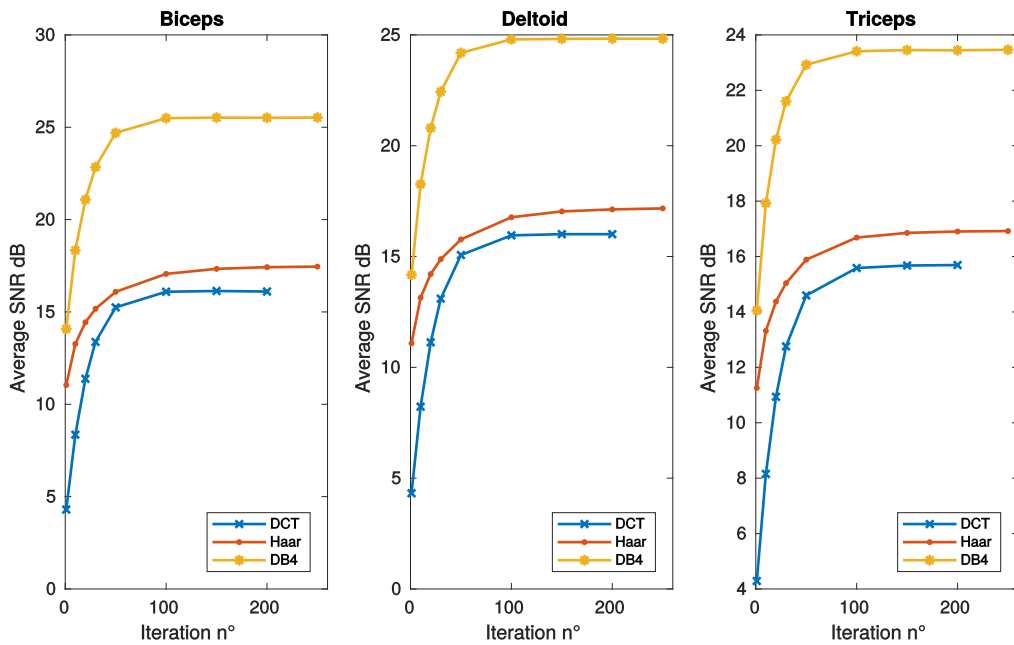


Figure 3. SNR vs. algorithm iterations, as achieved with convex optimization, for the same bases and muscles of Figure 1.

Figure 5 reports the SNR as a function of sparsity $S_M = k/M$ for different values of CF . In these cases, L1 and OMP have the better performance as they show a similar behavior. Figure 6 depicts the

SNR as a function of compression factor CF and different values of S_M . Also, in this case L1 and OMP show the better performance.

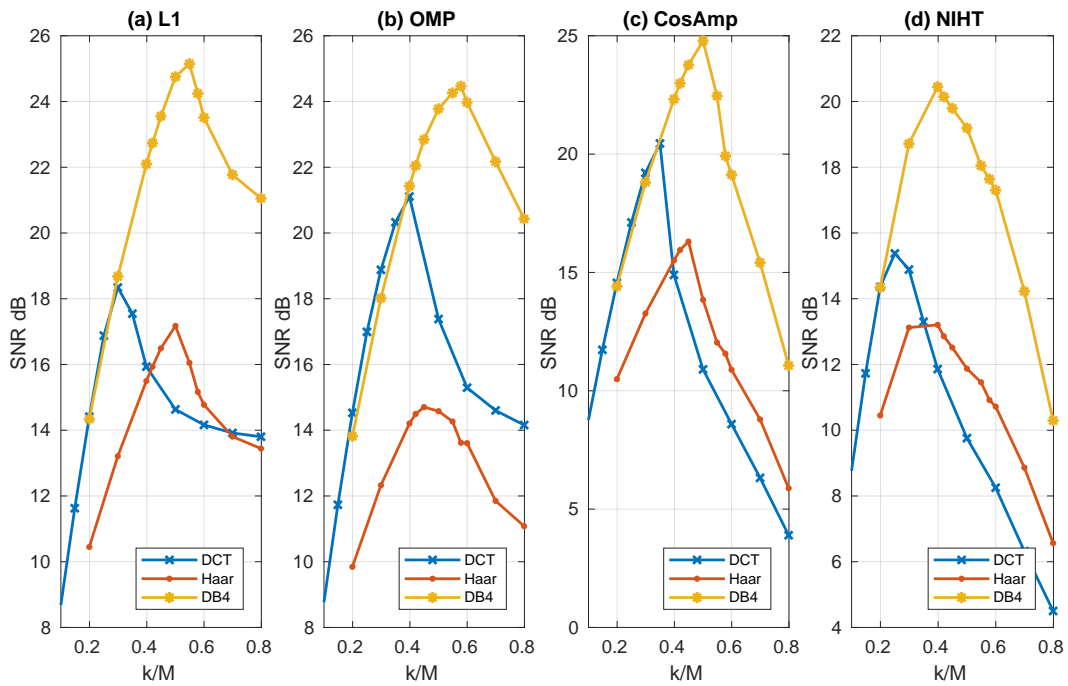


Figure 4. Case Study A—SNR as a function of sparsity $S_M = k/M$ with a compression factor $CF = 0.5$, for the four algorithms, (a) L1, (b) OMP, (c) CoSaMP, (d) NIHT, using the same bases of Figure 1.

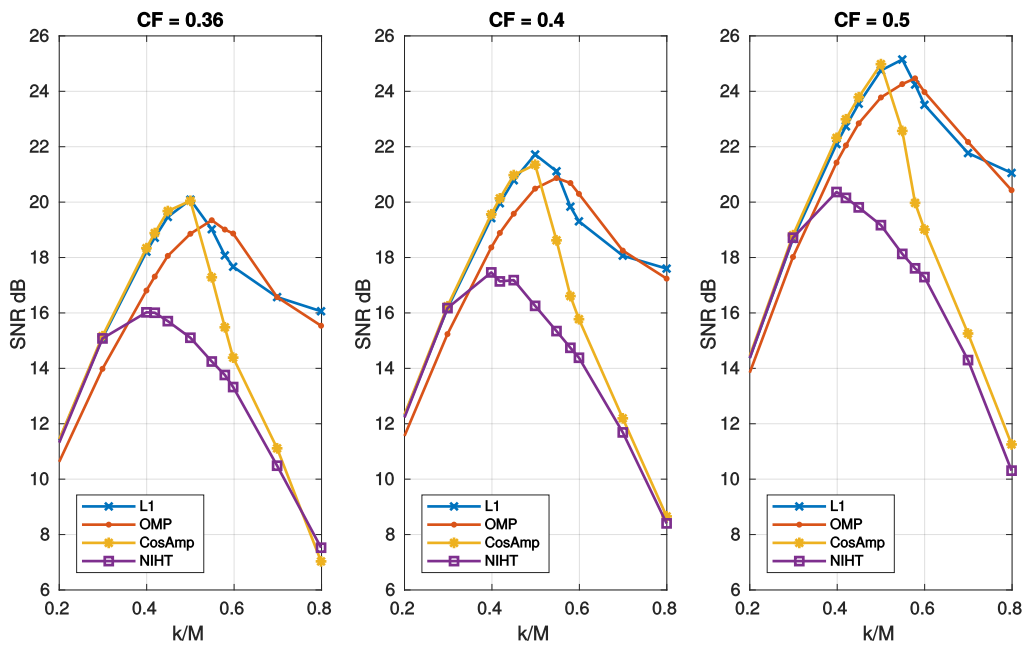


Figure 5. Case Study A—SNR as a function of sparsity $S_M = k/M$ and compression factor CF , for the four algorithms, L1, OMP, CoSaMP, NIHT, using the DB4 basis.

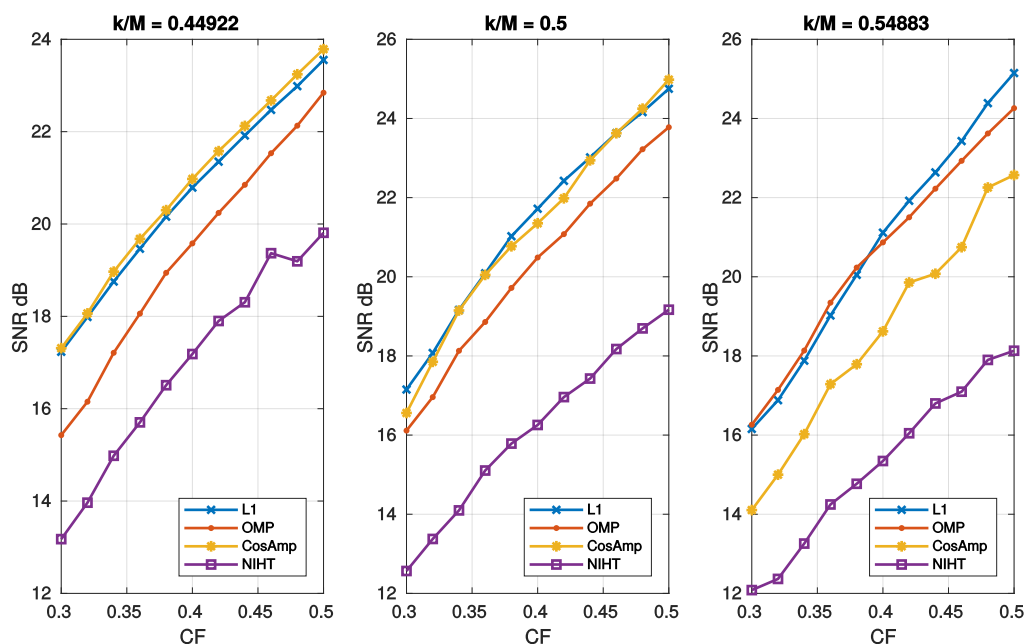


Figure 6. Case Study A—SNR as function of compression factor $CF = M/N$ for the four algorithms and three values of sparsity $S_M = k/M$.

4.1.3. Noise Tolerance

Real data CS acquisition systems are inherently noisy, thus to simulate a more realistic situation some experiments have been conducted with a noise superimposed to the signal. The effect of a noisy signal y on CS reconstruction corresponds to an error x_e in the sparse solution x given in this case by

$$x = x_{NF} + x_e \tag{49}$$

where x_{NF} denotes the noise-free solution. The error term x_e can be particularized for the four algorithms as follows:

$$\begin{aligned} x_{e,L1} &= A^\dagger n \\ x_{e,OMP} &= A_{\Lambda_k}^\dagger n \\ x_{e,CosAmp} &= A_T^\dagger n \\ x_{e,NIHT} &= A_{\Lambda_0}^\dagger n \end{aligned} \tag{50}$$

where n is the noise superimposed to y . It is straightforward to show that the following inequality

$$\frac{\|f - f_{x_e}\|_2}{\|f\|_2} \geq \left| \frac{\|f - f_{x_e}^{NF}\|_2}{\|f\|_2} - \frac{\|\psi x_e\|_2}{\|f\|_2} \right| \tag{51}$$

holds, thus giving the relationship

$$SNR \leq \frac{SNR_{NF} SNR_{noise}}{SNR_{NF} - SNR_{noise}} \tag{52}$$

where SNR_{NF} and $SNR_{noise} = \|f\|_2 / \|\psi x_e\|_2$ refer to the x_{NF} and x_e components, respectively. For high values of noise SNR degenerates to SNR_{noise} thus worsening the noise-free performance. It is worth noticing that for $L1$ the SNR_{noise} is independent of k/M , as it results from Equation (50) and the definition of SNR_{noise} . This implies that reducing the SNR_{noise} does not affect the reconstruction,

thus resulting almost constant with k/M . For the other algorithms x_e increases with k/M , so that a maximum for the SNR is expected. Figure 7 reports the SNR as a function of sparsity S_M for three values of noise superimposed, while Figure 8 is the noisy version of Figure 6, in which a value of SNR = 25 dB for the measurement signal y is used. The experimental results confirm the considerations stated above for L1, which shows the worst behavior when the measure SNR decreases. As for OMP, CoSaMP, and NIHT, their performances are almost independent of k/M for low values of it, while suddenly worsen when k/M exceeds a critical value of about 0.5. Finally, Figure 9 reports the computational cost and execution time on MATLAB respectively as functions of sparsity $S_M = k/M$. The execution time was computed using MATLAB tic-toc functions. These figures clearly show that L1-minimization outperforms the other algorithms.

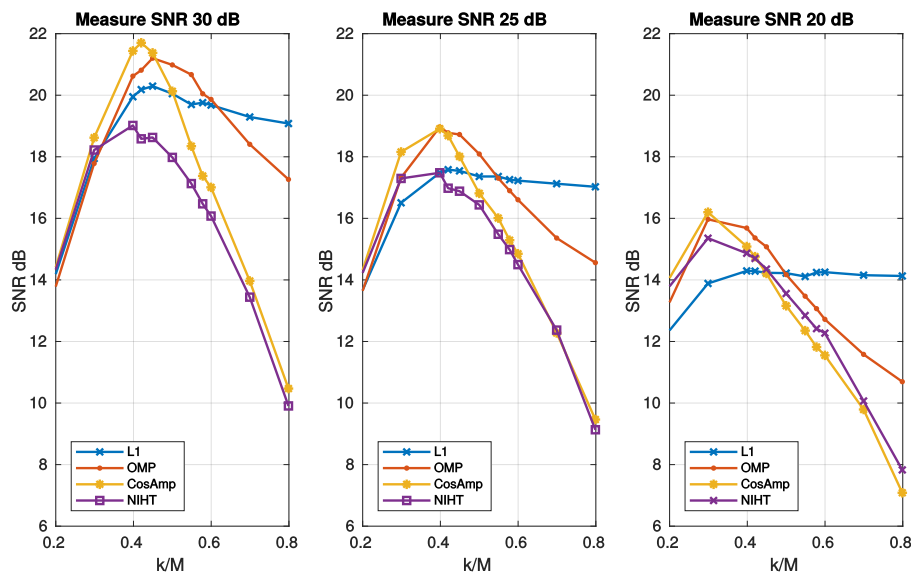


Figure 7. Case Study A—SNR as a function of sparsity $S_M = k/M$ for the four algorithms and three values of noise superimposed to the signal.

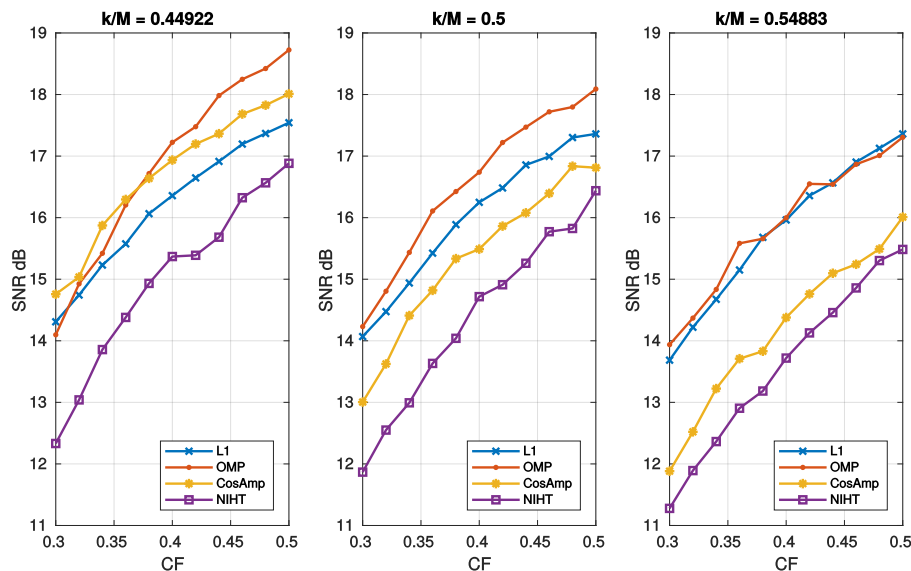


Figure 8. Case Study A—SNR as a function of compression factor $CF = M/N$ for the four algorithms and three values of sparsity $S_M = k/M$. A value of SNR = 25 dB for the measurement signal y is used.

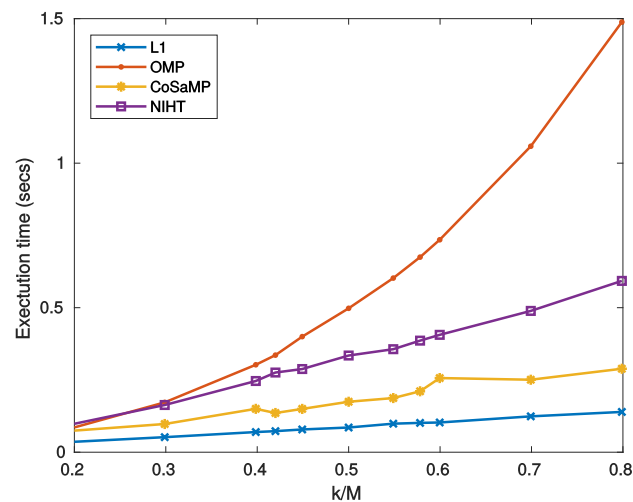


Figure 9. Case Study A—Execution time on MATLAB as a function of sparsity $S_M = k/M$.

4.2. Case Study B

The EMG signals used in this case study come from PhysioBank [52], a large and growing archive of well-characterized digital recordings of physiological signals and related data for use by biomedical research community. In particular, the data come from the ‘Neuroelectric and Myoelectric Databases’ of PhysioBank archives. A class of this database, named ‘Examples of Electromyograms’ [53], has been used; it contains short EMG recordings from three subjects (one without neuromuscular disease, one with myopathy, one with neuropathy). The signals are sampled at a frequency of 4 kHz and the frame has a length $N = 1024$, the same as case study A. As the signal from this dataset was not low-pass filtered, it contains all typical EMG frequency components, therefore this time we discarded DCT and Haar, using only DB4 basis. We chose to add this case study to analyse performances when signal has the lowest sparsity as possible which is the worst scenario for the reconstruction performance.

Figure 10 reports the execution time as a function of sparsity S_M for the four algorithms. Figure 11 compares the SNR as a function of sparsity S_M for three values of CF , as obtained with the four algorithms. As shown in these figures, the obtained results have a similar behaviour of those achieved in case study A.

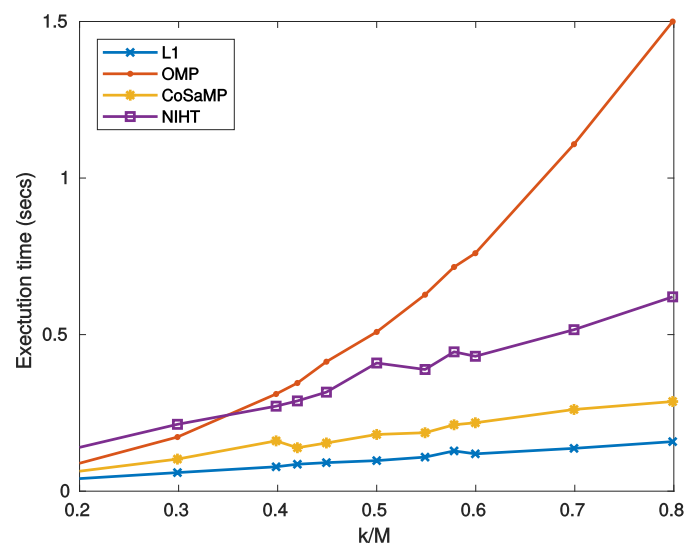


Figure 10. Case Study B—Execution time on MATLAB as a function of sparsity $S_M = k/M$.

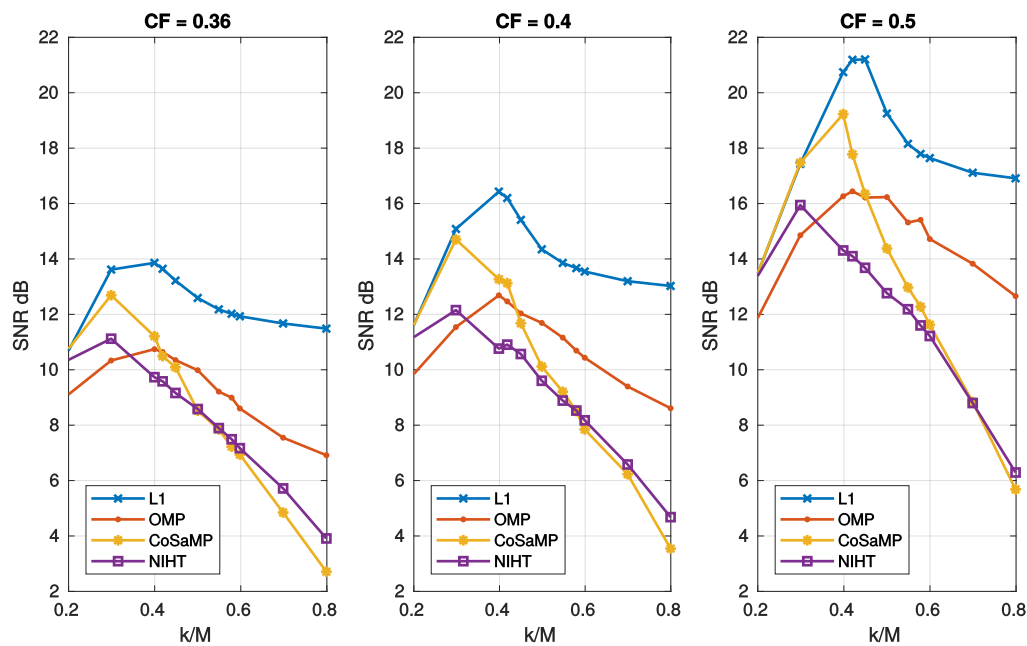


Figure 11. Case Study B—SNR as a function of sparsity $S_M = k/M$ and compression factor CF , for the four algorithms, L1, OMP, CoSaMP, NIHT, using the DB4 basis.

Finally, based on the experimental results previously reported a qualitative assessment of the four reconstruction algorithms can be derived that explores the trade-off in the choice of a CS reconstruction algorithm for EMG sensor application. To this end Table 2 summarizes the performance, in terms of accuracy, noise tolerance, and speed, of the four reconstruction algorithms.

The L1 minimization algorithm has an excellent behavior for accuracy, noise tolerance, and speed, thus outperforming the other algorithms. Among these, CoSaMP shows the best trade-off between accuracy and speed.

Table 2. Comparison, in terms of accuracy and speed, of the four algorithms L1, OMP, CoSaMP, NIHT.

Algorithm	Accuracy	Noise Tolerance	Speed	Computational Cost
L1	Excellent	Excellent	Excellent	$\mathcal{O}(N^2 N_{iter})$
OMP	Good	Good	Bad	$\mathcal{O}(Mk^3)$
CoSaMP	Fair	Fair	Good	$\mathcal{O}(Mk^2 N_{iter})$
NIHT	Bad	Fair	Fair	$\mathcal{O}(MNk)$

5. Conclusions

This paper presents a comprehensive comparative study of four of the most common algorithms for CS reconstruction of EMG signals, namely L1-minimization, OMP, CoSaMP, and NIHT. The study has been conducted using a wide range of EMG biosignals coming from two different datasets. Concerning algorithm accuracy, all the algorithms show a pronounced peak of SNR near the value of $k/M = 0.4 - 0.5$. However, while for OMP, CoSaMP, and NIHT, the SNR falls rapidly, the L1 algorithm remains nearly constant beyond the maximum. As for the effect of noise on CS reconstruction, L1-minimization shows a behavior that is almost independent of k/M . The results on computational cost and execution time on MATLAB show that L1-minimization outperforms the other algorithms. Finally, Table 2 summarizes the performance, in terms of accuracy, noise tolerance, speed, and computational cost of the four algorithms.

Author Contributions: Investigation, L.M., C.T., L.F., and P.C.; Methodology, L.M., C.T., L.F., and P.C.; Writing—original draft, L.M., C.T., L.F., and P.C.

Funding: This work was supported by a Università Politecnica delle Marche Research Grant.

Conflicts of Interest: The authors declare no conflict of interest. The funding sponsors had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, and in the decision to publish the results.

References

1. Naik, G.R.; Selvan, S.E.; Gobbo, M.; Acharyya, A.; Nguyen, H.T. Principal Component Analysis Applied to Surface Electromyography: A Comprehensive Review. *IEEE Access* **2016**, *4*, 4025–4037. [[CrossRef](#)]
2. Merlo, A.; Farina, D.; Merletti, R. A fast and reliable technique for muscle activity detection from surface EMG signals. *IEEE Trans. Biomed. Eng.* **2003**, *50*, 316–323. [[CrossRef](#)] [[PubMed](#)]
3. Fukuda, T.Y.; Echeimberg, J.O.; Pompeu, J.E.; Lucareli, P.R.G.; Garbelotti, S.; Gimenes, R.; Apolinário, A. Root mean square value of the electromyographic signal in the isometric torque of the quadriceps, hamstrings and brachial biceps muscles in female subjects. *J. Appl. Res.* **2010**, *10*, 32–39.
4. Nawab, S.H.; Roy, S.H.; Luca, C.J.D. Functional activity monitoring from wearable sensor data. In Proceedings of the 26th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, San Francisco, CA, USA, 1–5 September 2004; Volume 1, pp. 979–982.
5. Lee, S.Y.; Koo, K.H.; Lee, Y.; Lee, J.H.; Kim, J.H. Spatiotemporal analysis of EMG signals for muscle rehabilitation monitoring system. In Proceedings of the 2013 IEEE 2nd Global Conference on Consumer Electronics, Tokyo, Japan, 1–4 October 2013; pp. 1–2.
6. Biagetti, G.; Crippa, P.; Curzi, A.; Orcioni, S.; Turchetti, C. Analysis of the EMG Signal During Cyclic Movements Using Multicomponent AM–FM Decomposition. *IEEE J. Biomed. Health Inform.* **2015**, *19*, 1672–1681. [[CrossRef](#)] [[PubMed](#)]
7. Chang, K.M.; Liu, S.H.; Wu, X.H. A wireless sEMG recording system and its application to muscle fatigue detection. *Sensors* **2012**, *12*, 489–499. [[CrossRef](#)]
8. Ghasemzadeh, H.; Jafari, R.; Prabhakaran, B. A Body Sensor Network With Electromyogram and Inertial Sensors: Multimodal Interpretation of Muscular Activities. *IEEE Trans. Inf. Technol. Biomed.* **2010**, *14*, 198–206. [[CrossRef](#)] [[PubMed](#)]
9. Du, W.; Omisore, M.; Li, H.; Ivanov, K.; Han, S.; Wang, L. Recognition of Chronic Low Back Pain during Lumbar Spine Movements Based on Surface Electromyography Signals. *IEEE Access* **2018**, *6*, 65027–65042. [[CrossRef](#)]
10. Spulber, I.; Georgiou, P.; Eftekhar, A.; Toumazou, C.; Duffell, L.; Bergmann, J.; McGregor, A.; Mehta, T.; Hernandez, M.; Burdett, A. Frequency analysis of wireless accelerometer and EMG sensors data: Towards discrimination of normal and asymmetric walking pattern. In Proceedings of the 2012 IEEE International Symposium on Circuits and Systems, Seoul, Korea, 20–23 May 2012; pp. 2645–2648.
11. Zhang, X.; Chen, X.; Li, Y.; Lantz, V.; Wang, K.; Yang, J. A Framework for Hand Gesture Recognition Based on Accelerometer and EMG Sensors. *IEEE Trans. Syst. Man Cybern. Part A Syst. Hum.* **2011**, *41*, 1064–1076. [[CrossRef](#)]
12. Rahimi, A.; Benatti, S.; Kanerva, P.; Benini, L.; Rabaey, J.M. Hyperdimensional biosignal processing: A case study for EMG-based hand gesture recognition. In Proceedings of the 2016 IEEE International Conference on Rebooting Computing (ICRC), San Diego, CA, USA, 17–19 October 2016; pp. 1–8.
13. Brunelli, D.; Tadesse, A.M.; Vodermayr, B.; Nowak, M.; Castellini, C. Low-cost wearable multichannel surface EMG acquisition for prosthetic hand control. In Proceedings of the 2015 6th International Workshop on Advances in Sensors and Interfaces (IWASI), Gallipoli, Italy, 18–19 June 2015; pp. 94–99.
14. Yang, D.; Jiang, L.; Huang, Q.; Liu, R.; Liu, H. Experimental Study of an EMG-Controlled 5-DOF Anthropomorphic Prosthetic Hand for Motion Restoration. *J. Intell. Robot. Syst.* **2014**, *76*, 427–441. [[CrossRef](#)]
15. Oskoei, M.A.; Hu, H. Myoelectric control systems—A survey. *Biomed. Signal Process. Control* **2007**, *2*, 275–294. [[CrossRef](#)]
16. Biagetti, G.; Crippa, P.; Falaschetti, L.; Turchetti, C. Classifier Level Fusion of Accelerometer and sEMG Signals for Automatic Fitness Activity Diarization. *Sensors* **2018**, *18*, 2850. [[CrossRef](#)] [[PubMed](#)]

17. Roy, S.H.; Cheng, M.S.; Chang, S.S.; Moore, J.; Luca, G.D.; Nawab, S.H.; Luca, C.J.D. A Combined sEMG and Accelerometer System for Monitoring Functional Activity in Stroke. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2009**, *17*, 585–594. [[CrossRef](#)] [[PubMed](#)]
18. Varshney, U. Pervasive Healthcare and Wireless Health Monitoring. *Mob. Netw. Appl.* **2007**, *12*, 113–127. [[CrossRef](#)]
19. Movassaghi, S.; Abolhasan, M.; Lipman, J.; Smith, D.; Jamalipour, A. Wireless Body Area Networks: A Survey. *IEEE Commun. Surv. Tutor.* **2014**, *16*, 1658–1686. [[CrossRef](#)]
20. Cavallari, R.; Martelli, F.; Rosini, R.; Buratti, C.; Verdone, R. A Survey on Wireless Body Area Networks: Technologies and Design Challenges. *IEEE Commun. Surv. Tutor.* **2014**, *16*, 1635–1657. [[CrossRef](#)]
21. Zhang, Y.; Zhang, F.; Shakhsher, Y.; Silver, J.D.; Klinefelter, A.; Nagaraju, M.; Boley, J.; Pandey, J.; Shrivastava, A.; Carlson, E.J.; et al. A Batteryless 19 μ W MICS/ISM-Band Energy Harvesting Body Sensor Node SoC for ExG Applications. *IEEE J. Solid-State Circuits* **2013**, *48*, 199–213. [[CrossRef](#)]
22. Craven, D.; McGinley, B.; Kilmartin, L.; Glavin, M.; Jones, E. Compressed Sensing for Bioelectric Signals: A Review. *IEEE J. Biomed. Health Inform.* **2015**, *19*, 529–540. [[CrossRef](#)]
23. Cao, D.; Yu, K.; Zhuo, S.; Hu, Y.; Wang, Z. On the Implementation of Compressive Sensing on Wireless Sensor Network. In Proceedings of the 2016 IEEE First International Conference on Internet-of-Things Design and Implementation (IoTDI), Berlin, Germany, 4–8 April 2016; pp. 229–234. [[CrossRef](#)]
24. Ren, F.; Marković, D. A Configurable 12–237 kS/s 12.8 mW Sparse-Approximation Engine for Mobile Data Aggregation of Compressively Sampled Physiological Signals. *IEEE J. Solid-State Circuits* **2016**, *51*, 68–78.
25. Kanoun, K.; Mamaghanian, H.; Khaled, N.; Atienza, D. A real-time compressed sensing-based personal electrocardiogram monitoring system. In Proceedings of the 2011 Design, Automation Test in Europe, Grenoble, France, 14–18 March 2011; pp. 1–6.
26. Chen, F.; Chandrakasan, A.P.; Stojanovic, V.M. Design and Analysis of a Hardware-Efficient Compressed Sensing Architecture for Data Compression in Wireless Sensors. *IEEE J. Solid-State Circuits* **2012**, *47*, 744–756. [[CrossRef](#)]
27. Mangia, M.; Paleari, M.; Ariano, P.; Rovatti, R.; Setti, G. Compressed sensing based on rakeness for surface ElectroMyoGraphy. In Proceedings of the 2014 IEEE Biomedical Circuits and Systems Conference (BioCAS) Proceedings, Cleveland, OH, USA, 17–19 October 2014; pp. 204–207.
28. Marchioni, A.; Mangia, M.; Pareschil, F.; Rovatti, R.; Setti, G. Rakeness-based Compressed Sensing of Surface ElectroMyoGraphy for Improved Hand Movement Recognition in the Compressed Domain. In Proceedings of the 2018 IEEE Biomedical Circuits and Systems Conference (BioCAS), Cleveland, OH, USA, 17–19 October 2018; pp. 1–4.
29. Donoho, D.L. Compressed sensing. *IEEE Trans. Inf. Theory* **2006**, *52*, 1289–1306. [[CrossRef](#)]
30. Candes, E.J.; Tao, T. Near-Optimal Signal Recovery From Random Projections: Universal Encoding Strategies? *IEEE Trans. Inf. Theory* **2006**, *52*, 5406–5425. [[CrossRef](#)]
31. Donoho, D.L.; Stark, P.B. Uncertainty Principles and Signal Recovery. *SIAM J. Appl. Math.* **1989**, *49*, 906–931. [[CrossRef](#)]
32. Candes, E.J.; Tao, T. Decoding by linear programming. *IEEE Trans. Inf. Theory* **2005**, *51*, 4203–4215. [[CrossRef](#)]
33. Candes, E.J.; Romberg, J.; Tao, T. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Trans. Inf. Theory* **2006**, *52*, 489–509. [[CrossRef](#)]
34. Candes, E.J.; Wakin, M.B. An Introduction To Compressive Sampling. *IEEE Signal Process. Mag.* **2008**, *25*, 21–30. [[CrossRef](#)]
35. Qaisar, S.; Bilal, R.M.; Iqbal, W.; Naureen, M.; Lee, S. Compressive sensing: From theory to applications, a survey. *J. Commun. Netw.* **2013**, *15*, 443–456. [[CrossRef](#)]
36. Tropp, J.A.; Wright, S.J. Computational Methods for Sparse Solution of Linear Inverse Problems. *Proc. IEEE* **2010**, *98*, 948–958. [[CrossRef](#)]
37. Kim, S.; Koh, K.; Lustig, M.; Boyd, S.; Gorinevsky, D. An Interior-Point Method for Large-Scale ℓ_1 -Regularized Least Squares. *IEEE J. Sel. Top. Signal Process.* **2007**, *1*, 606–617. [[CrossRef](#)]
38. Tropp, J.A. Greed is good: Algorithmic results for sparse approximation. *IEEE Trans. Inf. Theory* **2004**, *50*, 2231–2242. [[CrossRef](#)]
39. Tropp, J.A.; Gilbert, A.C. Signal Recovery From Random Measurements Via Orthogonal Matching Pursuit. *IEEE Trans. Inf. Theory* **2007**, *53*, 4655–4666. [[CrossRef](#)]

40. Cai, X.; Zhou, Z.; Yang, Y.; Wang, Y. Improved Sufficient Conditions for Support Recovery of Sparse Signals Via Orthogonal Matching Pursuit. *IEEE Access* **2018**, *6*, 30437–30443. [[CrossRef](#)]
41. Davis, G.; Mallat, S.; Avellaneda, M. Adaptive greedy approximations. *Constr. Approx.* **1997**, *13*, 57–98. [[CrossRef](#)]
42. Pati, Y.C.; Rezaiifar, R.; Krishnaprasad, P.S. Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition. In Proceedings of the 27th Asilomar Conference on Signals, Systems and Computers, Pacific Grove, CA, USA, 1–3 November 1993; pp. 40–44.
43. Needell, D.; Tropp, J. CoSaMP: Iterative signal recovery from incomplete and inaccurate samples. *Appl. Comput. Harmon. Anal.* **2009**, *26*, 301–321. [[CrossRef](#)]
44. Dai, W.; Milenkovic, O. Subspace Pursuit for Compressive Sensing Signal Reconstruction. *IEEE Trans. Inf. Theory* **2009**, *55*, 2230–2249. [[CrossRef](#)]
45. Blumensath, T.; Davies, M.E. Normalized Iterative Hard Thresholding: Guaranteed Stability and Performance. *IEEE J. Sel. Top. Signal Process.* **2010**, *4*, 298–309. [[CrossRef](#)]
46. Ravelomanantsoa, A.; Rabah, H.; Rouane, A. Compressed Sensing: A Simple Deterministic Measurement Matrix and a Fast Recovery Algorithm. *IEEE Trans. Instrum. Meas.* **2015**, *64*, 3405–3413. [[CrossRef](#)]
47. Ravelomanantsoa, A.; Rouane, A.; Rabah, H.; Ferveur, N.; Collet, L. Design and Implementation of a Compressed Sensing Encoder: Application to EMG and ECG Wireless Biosensors. *Circuits Syst. Signal Process.* **2017**, *36*, 2875–2892. [[CrossRef](#)]
48. Shukla, K.K.; Tiwari, A.K. *Efficient Algorithms for Discrete Wavelet Transform: With Applications to Denoising and Fuzzy Inference Systems*; Springer Publishing Company, Incorporated: Berlin/Heidelberg, Germany, 2013.
49. Dixon, A.M.R.; Allstot, E.G.; Gangopadhyay, D.; Allstot, D.J. Compressed Sensing System Considerations for ECG and EMG Wireless Biosensors. *IEEE Trans. Biomed. Circuits Syst.* **2012**, *6*, 156–166. [[CrossRef](#)]
50. Biagetti, G.; Crippa, P.; Falaschetti, L.; Orcioni, S.; Turchetti, C. A portable wireless sEMG and inertial acquisition system for human activity monitoring. *Lect. Notes Comput. Sci.* **2017**, *10209 LNCS*, 608–620.
51. Biagetti, G.; Crippa, P.; Falaschetti, L.; Orcioni, S.; Turchetti, C. Human Activity Monitoring System Based on Wearable sEMG and Accelerometer Wireless Sensor Nodes. *BioMed. Eng. OnLine* **2018**, *17* (Suppl. 1), 132. [[CrossRef](#)]
52. PhysioBank. Available online: <https://physionet.org/physiobank/> (accessed on 19 March 2019).
53. Neuroelectric and Myoelectric Databases—Examples of Electromyograms. Available online: <https://physionet.org/physiobank/database/emgdb/> (accessed on 19 March 2019).



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).