Graph dictionary learning for the study of human motion

Marion Chauveau, Laurent Oudre, Antoine Mazarguil

Université Paris Saclay, Université Paris Cité, ENS Paris Saclay, CNRS, SSA, INSERM, Centre Borelli, F-91190, Gif-sur-Yvette, France.

This is an extended version of our paper [*] published in EMBC 2024.

[*] M. Chauveau, A. Mazarguil, and L. Oudre. Graph dictionary learning for the study of human motion. In Proceedings of the International Conference of the IEEE Engineering in Medecine and Biology Society (EMBC), Orlando, Florida, USA, 2024.

Abstract

Objective. In this paper, we propose a method to analyze human motion from the 3D positions of skeletal joints. *Methods.* For this purpose, we use a dictionary learning method, where each velocity sample is decomposed into a linear combination of a few atoms that are learned directly from the data. The originality of the approach is that this procedure is combined with the a Graph Signal Processing framework which allows to add a graph structure to tools initially dedicated to time series. *Results.* This methodology is tested on a dataset of 16 healthy subjects performing upper limb elevations. Features and visualizations are provided, and the robustesness of the approach is validated by constructing inter/intra-subjects distances. The method also obtains state-of-the-art performance on two popular tasks: denoising and human activity recognition. *Conclusion.* In this paper, we show that by combining a spatial graph that incorporates the skeletal structure, and a carefully designed dictionary learning algorithm, it is possible to extract interesting and discriminative features for motion analysis. *Significance.* Because of the interpretability of the features and visualizations obtained from this methodology, this approach could be used for interindividual comparison or longitudinal follow-up of patients.

Keywords: Human motion analysis, Graph Signal Processing (GSP), dictionary learning, sparse representation

1 Introduction

Human motion analysis is a captivating field of research due to its diverse applications, ranging from video surveillance and human-machine interaction to diagnostic assistance and medical rehabilitation [1–3]. In recent years, the use of skeleton-based motion data [4, 5], i.e. 3D positions of multiple skeletal joints over time, has demonstrated great potential for extracting meaningful information about human movement.

Various methods have been developed to process this type of data, each with its own advantages and limitations. Historical approaches such as Principal Component Analysis (PCA) [6] or

histogram-based calculations of joint locations [7], have been employed to provide compact representations of postures. However, skeletal information is not incorporated into the data processing of these approaches. Recent work has shown that exploiting structural knowledge of the human body can significantly improve performance in several motion analysis tasks, such as the popular Human Action Recognition (HAR) task [8, 9] — a computer vision task focused on automatically identifying human actions from videos by analyzing motion patterns and classifying them into predefined categories [10].

To leverage the multivariate nature of skeleton data, researchers have proposed to encode the skeleton information as a graph structure. In recent years, deep learning methods, in particular Graph Convolutional Networks (GCN), have gained attention due to their remarkable results in Human Action Recognition [8, 11, 12]. However, these models are often highly complex, time-consuming to train and require to work with large datasets [13–15]. Deep-learning methods are also often task-specific, and the features obtained in a supervised manner are challenging to interpret. This is an issue if the focus is not only on identifying actions but rather on studying how they are performed and what are the common and individual characteristics of human movement. In fact, a fundamental question arises regarding the ability to identify a unique "motion style" or "signature" for each individual. Early experiments conducted by Gunnar Johansson in the 1960s [16], where lights attached to joints were sufficient to recognize walking individuals, inspired subsequent research on breaking down movements and studying variations between individuals [17–20]. For such studies, the features need to be both discriminative and interpretable to identify similarities and differences between actions performed by distinct subjects.

In light of these considerations, the approach introduced in this article is not based on deep learning techniques but rather on the Graph Signal Processing (GSP) framework [21, 22]. As for GCNs, GSP methods assume that skeletal structure is encoded in the form of a graph, that reflects the proximity between body joints. Yet, instead of using *black boxes* like convolutional networks, tools derived from the GSP framework offer the interesting property of being directly interpretable, as most of the notions defined in this framework are extensions of standard signal processing tools to irregular domains (filtering, sampling, Fourier transform, sparsity...). In the GSP framwork, 3D skeleton data are simply seen as *graph signals* that are lying on the graph. This makes it possible, for example, to use of the Graph Fourier Transform (GFT) to decompose and analyse motion [23, 24].

In this article, we introduce the first application (up to our knowledge) of graph signal dictionary learning for human motion data. By leveraging the advantages of tools from the GSP framework and dictionary learning methods, we develop an approach that combines simplicity, interpretability and versatility. The contributions are as follows:

- We propose a method to construct a spatial graph that reflects the structure of the human body as well as a Mexican Hat graph-wavelet basis [25] that is well adapted to motion analysis.
- We adapt a double sparsity dictionary learning method [26,27] to combine these wavelets and create compact, discriminative and interpretable representations of the input data.
- We demonstrate the versatility of the proposed features on three distinct popular motion analysis tasks. The interpretable features and visualizations obtained from our methodology are firstly used to gain insights into motion "signatures" on a database of upper limb elevations. Then, we show the effectiveness of the approach on two popular tasks : Signal denoising and Human Action Recognition.

This article is organized as follows. We introduce notations and we recall some useful concepts

about GSP and dictionary learning in Section 2. Then, we introduce the Arm-CODA database used for our study, along with our methodology to extract innovative features by applying a dictionary learning approach within the GSP framework (Section 3). In Section 4, we propose 3 experiments to assess the relevance of the method: 1) Analysis of upper limb elevations, 2) Signal Denoising and 3) Human Action Recognition. Finally, we present the results obtained on these 3 different tasks in Section 5 and we discuss the relevance of the proposed method for human motion analysis, especially inter-individual comparison and longitudinal follow-up of patients.

2 Technical background

Before going through the details of the proposed method we first introduce some useful concepts and related works about graph signal processing, and dictionary learning.

2.1 Graph signal processing

Formally, a graph is defined as a triplet $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, W\}$, $\mathcal{V} = \{v_1, v_2, ..., v_N\}$ being the set of nodes and $W \in \mathbb{R}_+^{N \times N}$ being the affinity matrix that contains the weights of the edges specified in the set $\mathcal{E} = \{(i,j), i,j \in \mathcal{V}\}$. From the affinity matrix, it is possible to compute the Laplacian of the graph $\mathcal{L} = D - W$ with D the diagonal degree matrix, i.e. $D_{ii} = \sum_{j \neq i} W_{ij}$. A graph is said to be connected if $\forall u, v \in \mathcal{V}$ there exist a finite sequence of edges connecting u and v. In the following, we will only deal with connected and undirected graphs so that the Laplacian is a symmetric matrix.

The use of this mathematical representation can be very helpful when it comes to dealing with signals that evolve on complex structures, such as biological, social or financial data [28, 29]. The GSP framework was recently developed to process these type of data by adapting signal processing tools to the study of graph signals. Formally, a graph signal is a function $f : \mathcal{V} \to \mathbb{R}$ that assigns a scalar value or a vector to each node of a graph. This function can be represented in a vector form $f \in \mathbb{R}^N$, which implies an implicit numbering of the vertices. In the case of skeleton-based motion data, each joint can be represented by a node to which the graph signal function will for example associate the velocity of the corresponding joint at a given instant. In this context, a graph signal is a velocity profile of the body that informs us about the limbs that are in motion, while carrying information about the structure of the skeleton.

Among the major tools developed in the GSP framework, the Graph Fourier Transform (GFT) makes it possible to study graph signals in the spatial frequency domain [22]. The eigendecomposition of the Laplacian provides us with a spectral basis corresponding to the eigenvectors denoted by $U = [\mathbf{u}_1, ..., \mathbf{u}_N]$, and eigenvalues interpreted as spatial frequencies denoted by $\sigma(\mathcal{G}) = \{\lambda_1, ..., \lambda_N\}$. For a given graph signal $\mathbf{y}$, it is thus possible to define its GFT as:

$$\tilde{\mathbf{y}} = U^T \mathbf{y} \tag{1}$$

where $\tilde{\mathbf{y}}$ contains the energies associated with each frequency. Analogously to the classical Fourier transform, the eigenvectors associated with small eigenvalues, i.e small frequencies, are smooth graph signals that do not exhibit strong variations across connected vertices. Some approaches leverage the properties of this decomposition to study human motion and tackle the gesture recognition task [23, 24].

2.2 Graph construction from data

The construction of the graph is a crucial step in the processing of data evolving on a complex structure. The challenge is to model the interactions between data entities in the form of pairwise relationships. In the case of skeleton-based motion data, the underlying structure that governs the interactions between the joints is the human skeleton. From this information, there are several ways to build a graph. In general, each joint is associated with a node of the graph and the weighted edges are determined from the physical dependencies between joints. Thus, many approaches construct a spatial graph by imposing a unitary weight on edges only if there exist a physical limb connecting the two nodes [23, 24, 30]. Other methods do not limit themselves to a spatial representation and construct a skeletal-temporal graph to model temporal dynamics [11, 24]. Finally, it is also possible to build the graph from the data. Solving an optimization problem on the Laplacian can for example allow to obtain a graph such that the processed data are smooth [31, 32], i.e. close points in the sense of the graph will tend to have similar values.

2.3 Dictionary learning

To describe human motion effectively from skeleton-based motion data, the typical approach is to extract compact and informative features from this data. Dictionary learning methods are for example widely used when it comes to finding a sparse approximation of a signal. Within the GSP framework, we are dealing with a collection of graph signals that can be written in the following vector form $\mathbf{y} = [y_1, ..., y_N]$, with y_i the value associated to the node i of the graph. The principle is to decompose these graph signals on a set of vectors $(\mathbf{d}^1, ..., \mathbf{d}^L)$ called *atoms* and stored on the columns of a dictionary matrix $D \in \mathbb{R}^{N \times L}$, L being the size of the dictionary. The decomposition of a given graph signal $\mathbf{y} \in \mathbb{R}^N$ can be written as follows:

$$\mathbf{y} \approx \sum_{l=1}^{L} x_l \mathbf{d}^l \tag{2}$$

where $\mathbf{x} = (x_1, ..., x_L)$ is the activation vector giving the contribution of each *atom* in the approximation of the signal $\mathbf{y}$. The sparsity of this activation vector can be imposed using greedy algorithms such as the Matching Pursuit [33], or using convex relaxation methods [34].

Regarding the dictionary, it can be constructed analytically but it can also be learned from the processed data. In the first case, the dictionary structure is generally based on a mathematical model. This is for example the case of any Wavelet basis [25, 35, 36], but also of the Fourier basis [23, 24, 37] which can be considered as a fixed dictionary. The use of an analytical dictionary has the advantage of being numerically fast, but it can also be poorly adapted to the data studied. Concerning approaches based on a learned dictionary, the *atoms* are directly inferred from the data using training algorithms such as the method of optimal directions [38] or the K-SVD algorithm [39] to name a few. This method is often more expensive numerically but it allows to obtain dictionaries more adapted to the data. If we need a dictionary that combines the advantages of analytical and learned dictionaries, it is also possible to impose a structure and learn parameters for this structure [40–42].

Subject	Sexe	Age	Size (cm)	Weight (kg)
1 (A)	F	47	170	65
2 (B)	M	57	173	75
3 (C)	F	52	156	64
4 (D)	M	28	179	73
5	M	30	175	77
6	M	27	188	78
7	F	50	172	68
8	F	62	158	51
9	F	65	168	74
10	M	42	175	84.5
11	M	40	181	95
12	M	47	180	62
13	M	62	171	84
14	M	50	178	75
15	M	23	183	93
16	M	25	172	61

Table 1: Characteristics of the 16 subjects from the Arm-CODA database. The grey ones are the subjects A, B, C and D that we have analyzed in the following sections.

3 Methodology

In this section, we introduce the Arm-CODA database used for our study, along with our strategy to extract understandable and discriminative features from skeleton-based motion data. The key steps of the proposed method are the following:

1. Constructing a spatial graph from the database in such a way that it reflects the human body
2. Deriving velocity profiles that indicate how fast the different body joints are moving at each time
3. Creating a basis of Mexican Hat wavelets [25] well adapted to the study of human motion
4. Employing this basis within the double sparsity approach [27] to learn a dictionary that efficiently approximate the velocity profiles of our database
5. Using the contribution of each component of the learned dictionary in the signal reconstruction to build interpretable features

In the following, we will call Double Sparse Mexican Hat (DSMH) the dictionary obtained thanks to the double sparsity method.

3.1 Arm-CODA dataset

The Arm-CODA dataset [43] used in this paper was obtained from a cohort of 16 healthy subjects, whose characteristics are given in Table 1. These subjects were asked to perform several types of movements including elevation movements of the right arm, the left arm, and both arms simultaneously, in a standing position. For these movements, the instruction was to reach maximal natural elevation following the scapular plane, pause for about 1 sec, and finally lower the arm back to the

initial position. The study protocol was conducted in compliance with the Good Clinical Practices protocol and Declaration of Helsinki principles. All participants provided informed written consent. STROBE and GRRAS guidelines were used for reporting.

The subjects were equipped with 46 Cartesian Optoelectronic Dynamic Anthropometer (CODA) motion system 3D position markers. These sensors provide 3D positions data in millimeters measured at a frequency of 100 H_z thanks to a system of 6 depth cameras[1]. Formally, we will note $p_{t,i} = [x_i^{(t)}, y_i^{(t)}, z_i^{(t)}]$ the position of the joint i at time t, with $i \in \{1, ..., N\}$, N being the total number of joints and $t \in \{1, ..., T\}$, T being the number of samples available. In the end, we have access to the 3D position of N skeleton joints of a subject at T given instants. Some sensors often hidden from the cameras during the movements have been excluded from the dataset we used in this article. The locations of the $N = 34$ retained markers are detailed in Fig. 1. As illustrated in the figure, the x-axis corresponds to the sagittal axis, the y-axis to the frontal axis and the z-axis is aligned with the longitudinal axis, oriented upward.

Each movement is repeated 2 or 3 times by each subject during a single acquisition. In this work, we have only used the motion sequences of 16 subjects for which the different repetitions of each movement were segmented by hand. In the end, the processed database contains 3 different elevation movements of the right arm, the left arm, and both arms in the scapular plane, performed 2 or 3 times by 16 subjects, which makes a total of 143 motion sequences.

The latter are then processed as follows. We start by subsampling the signals by a factor 10 in order to limit the computation time. Then, we compute the velocity signals defined as the time-discrete derivative of the positions : $v_{i,t} = p_{i,t+1} - p_{i,t} \quad \forall \quad t = 1, ..., T-1$. To eliminate the outliers, we finally apply a median filter of order 3 on the velocity time series.

3.2 Graph construction

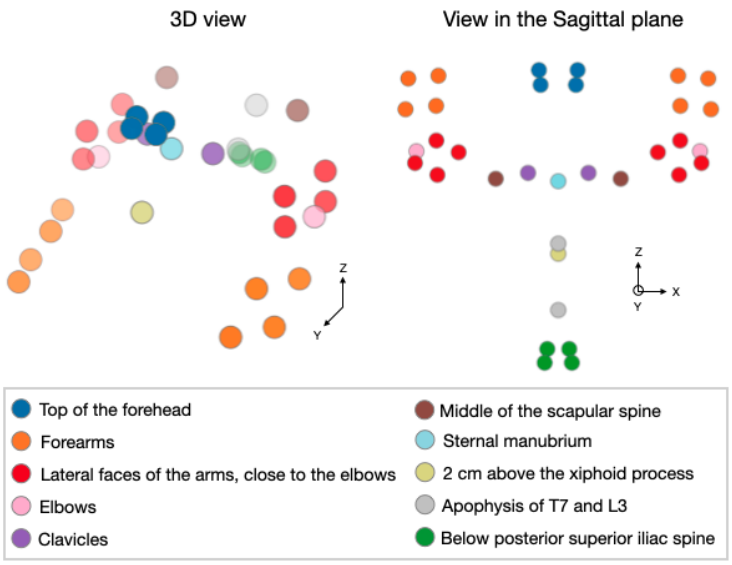
The constructed graph is an undirected and weighted spatial graph representative of the human body. Each node is associated with a joint (or a sensor) and the weighted edges are determined from the database. The weight $w_{i,j}$ is computed by taking the maximum distance $d_{max}^{i,j}$ between the joints i and j on the whole database :

$$w_{i,j} \propto e^{-d_{max}^{i,j}} \tag{3}$$

Then, the resulting graph is simplified to construct a k-nn graph, i.e. for each joint we only keep the edges linking it to its k nearest neighbors. k is the smallest possible value to keep a path connecting each pair of nodes so that we have a connected graph. Finally, we impose that our graph is symmetrical with respect to the longitudinal axis by computing the mean of the weights associated with two symmetrical edges. In this constructed graph, the spatial proximity between 2 sensors is given by the weight of the edge linking the two nodes associated to these sensors. With this procedure we thus obtain a graph where two close nodes in the sense of the graph correspond to two close joints along the body.

Fig 2a illustrates an example of the constructed graph for the Arm-CODA dataset with $k = 5$. We can first note that the preserved edges make it possible to recognize the shape of the human body. Moreover, it should be added that this graph will be the same for all the subjects, so that the elements used to model the movement are common to the different subjects.

[1]Resolutions: 1 in 70,000 within its field of view
Nominal operating range: between 2.0m and 4.5m from the unit



(a)

(b)

Figure 1: (a) Detail of the N = 34 markers locations for the Arm-CODA dataset. (b) Picture taken after the installation of the 3D position sensors.

3.3 Dictionary learning

We are dealing with velocity signals evolving on a graph representative of the human skeleton. Thus, the processed data are a set of graph signals interpreted as velocity profiles. At a given time, a graph signal is a function which associates to each node of the graph the velocity of a body joint.

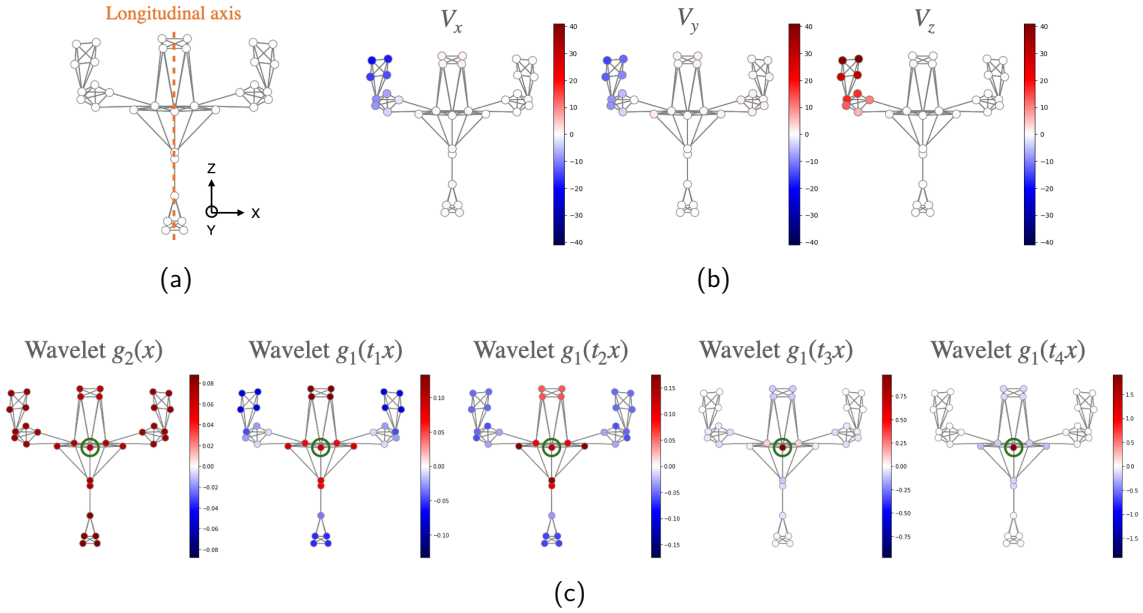


Figure 2: (a) Weighted and undirected k-nn Graph constructed from the Arm-CODA dataset with $k = 5$. Each node is represented with a white circle and the edges are the lines connecting these nodes. The graph is symmetrical with respect to the longitudinal axis. The human body represented in this graph faces us so that the left arm is located on the right, and the right arm is located on the left. (b) Example of a graph signal at a given time during a right arm elevation in the sagittal plane. We represent this signal by separating the 3 dimensions of space. For a given time, we have 3 graph signals, each corresponding to the velocity signal along the x, y, and z-axis. The velocity associated with each node is indicated by a color ranging from darkest blue for negative values to darkest red for positive values. (c) Example of wavelets centered on the node circled in green for the Arm-CODA graph. The first wavelet is generated with the low-pass filter $g_2(x)$. The 4 other wavelets are obtained by using the kernels $g_1(tx)$ with $t \in \{t_1 = 5.17, t_2 = 1.51, t_3 = 0.44, t_4 = 0.13\}$. The scale t_1 corresponds to the most extended wavelet and t_4 to the narrowest one.

As illustrated on Fig 2b, we represent the graph signals by separating the 3 dimensions of space. For a given instant, we have 3 graph signals each corresponding to the velocity profile along the x, y, and z-axis. All these signals represented as vectors are then stored on the columns of three matrices $Y^{(d)} \in \mathbb{R}^{N\times T}$, $d \in \{x, y, z\}$. The objective will be to approximate each of these velocity profiles using a linear combination of a few graph signals that will be stored in a common dictionary.

As explained earlier, we have chosen the double sparsity approach [26, 27] which is halfway between the analytic and the learned dictionary. The dictionary is defined as a product $D = \Phi A$, where $\Phi \in \mathbb{R}^{N\times K}$ is a fixed dictionary containing *atoms* in its columns, and $A \in \mathbb{R}^{K\times L}$ is a learned sparse matrix.

The DSMH dictionary is a set of *super-atoms* $(\mathbf{d}^1, ..., \mathbf{d}^L)$, defined as linear combinations of

atoms stored in the fixed dictionary $\Phi = (\phi^1, ..., \phi^K)$:

$$\mathbf{d}^l \approx \sum_{k=1}^{K} a_k \phi^k \tag{4}$$

where $\mathbf{a} = (a_1, ..., a_K)$ is a sparse vector containing the weights of the linear combination. The challenge is to learn these weights from the data so that the *super-atoms* can be used to approximate as well as possible the graph signals stored in $Y^{(d)}$. Formally, the learning problem is given by:

$$\begin{aligned} \underset{A^{(d)}, X^{(d)}}{\text{argmin}} \quad & ||Y^{(d)} - \Phi A^{(d)} X^{(d)}||_F^2 \\ \text{s.t.} \quad & ||x_i||_0 \leq s_1 \qquad \forall i, \\ & ||a_i||_0 \leq s_2 \qquad \forall j, \qquad ||\Phi a_j||_2 = 1 \qquad \forall j. \end{aligned} \tag{5}$$

$Y^{(d)} \in \mathbb{R}^{N \times T}$ is the data matrix containing the graph signals in its columns. $\Phi \in \mathbb{R}^{N \times K}$ is the fixed dictionary and A is a sparse matrix, having s_2 non-zeros per column such that each *super-atom* is a combination of at most s_2 *atoms*. $X^{(d)} \in \mathbb{R}^{L \times T}$ is the activation matrix, having s_1 non-zeros per column such that the decomposition of each graph signal is done with the combination of at most s_1 *super-atoms*. Finally, we impose that the *super-atoms* of the DSMH dictionary are unit vectors.

In the end, the method rely on a set of L *super-atoms* that are optimized to approximate as well as possible the graph signals we are processing. These *super-atoms* are defined as linear combinations of only a few *atoms* that we have to choose beforehand.

3.4 Mexican Hat Wavelets

This section is dedicated to the construction of the *atoms* stored in the fixed dictionary. In this work, we have opted for Mexican Hat wavelets. The velocity profiles of the database can correspond to the global motion of several body joints but it can also be related to a very localised movement. Thus, we aim to construct wavelets with various ranges and locations to be able to describe these different behaviours. The construction is done using an operator $T_g^\beta = g(\beta\mathcal{L})$, with β the scaling of the wavelet, g the kernel and $\mathcal{L}$ the Laplacian of the graph [25]. Then, the wavelets are generated by applying this operator to an impulse $\delta_n \in \mathbb{R}^N$, which is equal to zero except at node n where it takes the value 1.

Formally, the coefficients of the wavelet $\psi_{\beta,n} \in \mathbb{R}^N$, at scale β and centered on node n, are obtained with the following formula:

$$\psi_{\beta,n}(m) = \sum_{l=0}^{N-1} g(\lambda_l) \hat{\delta_n}(l) \mathbf{u}_l(m) \tag{6}$$

with $\mathbf{u}_l(n)$ the n^{th} coefficient of the eigenvector associated to the eigenvalue λ_l of the Laplacian, and $\hat{\delta_n}$ the GFT of the impulse.

In the following, we will use two different kernels:

- A band-pass filter $g_1(\beta x) = \beta x \times e^{-\beta x}$, which will be used to create Mexican hat wavelets at scale β.

- A low-pass filter $g_2(x) = \gamma e^{-(\frac{20}{0.4\lambda_{max}}x)^4}$, with $\gamma = 1.2 \times e^{-1}$, that will allow to capture low-frequency phenomena.

A given atom of the fixed dictionary is defined by a wavelet operator and the impulse on which it is applied, i.e the node on which the wavelet is centered. We note N_w the number of wavelets with different scales that we consider, including the wavelet constructed with the low-pass filter. An example of some Mexican Hat wavelets is illustrated in Figure 2c on the graph constructed from the Arm-CODA database. These wavelets which have different scales and are centered on different nodes of the graph allow us to account for phenomena with different ranges and localities. The combination of these atoms thanks to the double sparsity method will then provide interpretable *super-atoms* that are adapted to the data.

4 Experiments

To assess the relevance of the DSMH approach we propose to use it in the scope of 3 different experiments. The first one consists in designing features that are used to analyze upper limb elevations from the Arm-CODA database. Then, we aim to apply the DSMH method on two "real world" problems: the denoising task and the HAR task.

4.1 Experiment 1: Analysis of upper limb elevations

In this experiment we apply the DSMH method on upper limb elevations to show that it can be used to obtain interpretable and discriminative features.

4.1.1 Database

We use the 143 motion sequences from the Arm-CODA database presented in Section 3.1. This corresponds to elevation movements of the right arm, the left arm and both arms in the scapular plane performed 2 or 3 times by 16 different subjects.

4.1.2 Protocol

The graph is constructed from the whole database using the method proposed in Section 3.2 and the data are processed as explained in Section 3.1. The double sparsity method is then applied to obtain the DSMH dictionary and the sparse activation matrices. The latter are finally used to construct 2 types of features that are presented below.

Timelines: A column of the matrix $X^{(d)}$ gives the contribution of each *super-atom* to reconstruct the velocity profile along the d-axis at a given time. For a each motion sequence, a timeline is constructed to indicate the *super-atoms* that contribute the most to the signal reconstruction over time for the 3 dimensions of space. An example of a timeline is illustrated in Figure 3 for a left arm elevation. As expected the most used *atoms* over time are left unilateral graph signals.

The same procedure can be done to construct a timeline that indicate the second most used *super-atoms* over time.

Histograms: For each motion sequence, we compute the total percentage contribution of each *super-atom* to construct an histogram defined as a probability vector. Then, we use the Jensen-Shannon divergence to compute the distance between two histograms [44], so that we get a distance between 2 motion sequences.

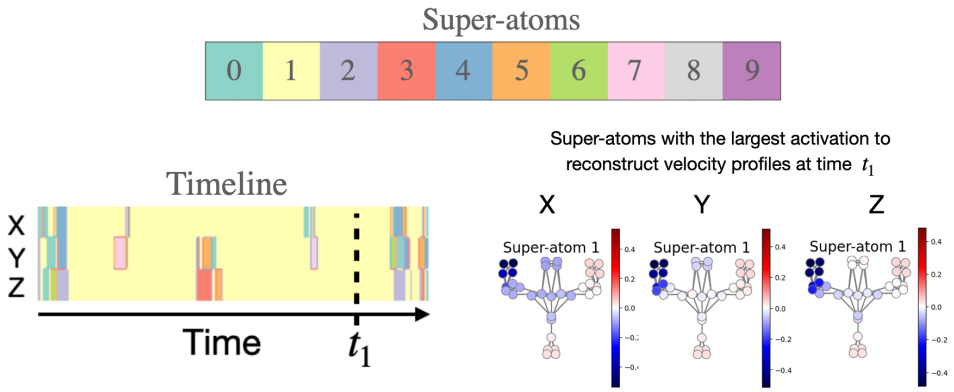


Figure 3: Example of a timeline for a right arm elevation. The timeline indicates which *super-atoms* contribute the most to the signal reconstruction over time for each dimension of space. Each color corresponds to a *super-atom* (Note that *super-atom* i for dimension X is different from *super-atom* i for dimension Y). At the time t_1 indicated on the figure, the most used *super-atoms* for the dimensions X, Y, and Z are respectively *super-atoms* 1, 1, and 1.

4.1.3 Parameters

In the following, we will use $N_w = 5$ wavelets with different scales, ranging from a very wide wavelet to a very narrow one. Fig 2c illustrates these 5 wavelets centered on a node of the graph built from the Arm-CODA database. Each of these wavelets is centered on each of the N nodes of the graph to build the $5N$ atoms of the fixed dictionary $\Phi \in \mathbb{R}^{N \times 5N}$.

We choose to limit to $L = 10$ the size of the DSMH dictionary, i.e. the number of super-atoms built with the double sparsity method. Then, the parameters s_1 and s_2 are chosen to ensure a signal reconstruction greater than 80%. For a given signal matrix Y the reconstruction is defined as follows:

$$Rec(Y) = 100 \times \left(1 - \frac{||Y - \Phi AX||_F}{||Y||_F}\right) \tag{7}$$

In the end, we apply the DSMH method with $s_1 = 3$ and $s_2 = 5$, so that each *super-atom* is a combination of at most 5 wavelets and each velocity profile is reconstructed thanks to the combination of 3 *super-atoms*.

4.2 Experiment 2: Denoising

In this experiment we aim to evaluate the performance of the DSMH method to denoise skeleton-based motion data and we compare with state-of-art approaches.

4.2.1 Dataset

We use motion sequences from a single subject of the Arm-CODA dataset, i.e. right arm, left arm and both arm elevations performed by a given subject, and these sequences are processed as explained in Section 3.1.

4.2.2 Method

We select a certain percentage of data points to which we add a Gaussian noise with zero mean and standard deviation equal to 5. We use the double sparsity method to obtain a denoised signal. The performance is finally evaluated by calculating the decibel version of the normalized root squared error (nRSE-db):

$$\text{nRSE-db} = -\log_{10}\left(\frac{||Y - Y_{denoised}||_F}{||Y||_F}\right) \tag{8}$$

The higher the nRSE-db, the better the quality of the signal reconstruction.

4.2.3 Comparison with state-of-the-art approaches

We compare the DSMH method to 2 different approaches:

- A sparse Fourier transform, where each graph signal is reconstructed with at most s_1 eigenvectors thanks to an Orthogonal matching pursuit algorithm [45].
- A graph filtering method [46] which consists of an optimization problem. It provides a smooth approximation of the original noisy signal, the strength of the smoothing being controlled by a tuning parameter α.

4.2.4 Calibration & parameters

For all methods, we compute a grid search to find the best performances. The parameter $s_1 = 5$ is set to obtain the highest nRSE-db score with the sparse Fourier method for 20% and 50% of corrupted data. For the DSMH method, we keep $s_1 = 5$ and the parameter $s_2 = 7$ is optimized to obtain the highest score. Concerning the graph filtering method, the tuning parameter providing the best results is $\alpha = 0.39$.

4.3 Experiment 3: Human Action Recognition (HAR)

In this experiment, we evaluate the ability of the proposed method to provide effective features for Human Action Recognition and we compare with state-of-the-art approaches.

4.3.1 Datasets

We present here the 3 databases used for this experiment. All these datasets are widely used for the HAR task.

UTKinect-Action3D (UTK) [7] : 199 data samples captured with a single stationary Kinect camera. The dataset contains the 3D positions of 20 skeleton joints for 10 different actions (*carry, clap hands, pick up, pull, push, sit down, stand up, throw, walk, wave hands*) performed twice by 10 subjects.

MSR-Action3D (MSR) [47] : most common database for 3D action recognition, containing 557 motion sequences captured by a depth-camera [4]. We have access to the positions of 20 skeleton joints for 20 actions (*bend, draw circle, draw tick, draw x, forward kick, forward punch, golf swing, hand catch, hand clap, hammer, high arm wave, high throw, horizontal arm wave, jogging, pick up and throw, side boxing, side kick, tennis serve, tennis swing, two-handwave*) repeated 2 or 3 times by 10 subjects.

Florence-Action3D (F3D) [48] : 215 data samples captured with a stationary Kinect sensor. Unlike the two previous databases, it uses the positions of only 15 skeleton joints. The 9 activities (*wave, drink from a bottle, answer phone, clap, tight lace, sit down, stand up, read watch, bow*) are performed 2 or 3 times by 10 subjects.

4.3.2 Method

As in [24] we use the positions of the body joints to tackle the HAR task[2]. By applying the DSMH method on this position signal we obtain activation matrices that are used to extract feature vectors. The construction of the features relies on a temporal pyramid procedure (TPM) [50][3]. Then, these features are fed to a linear SVM classifier trained with a leave one-subject-out validation scheme.

4.3.3 Comparison with state-of-the-art approaches

We evaluate performance in gesture recognition with 2 different graphs as well as two different methods to construct the features. The first graph is the weighted graph proposed in Section 3.2. The second one is a spatio-temporal graph constructed as in [24]. Concerning the 2 approaches to obtain motion representations, we aim to compare the DSMH method with a Graph Fourier Transform method that uses the Fourier basis as an analytical dictionary to approximate graph signals. In the end, the different combinations of graphs and methods to obtain features lead to 4 different configurations:

- **Spatio-temporal Graph + GFT:** this approach is similar to the one presented in [24].
- **Spatio-temporal Graph + DSMH:** this approach is a mix between the DSMH method and the graph proposed in [24].
- **Weighted Graph + GFT:** it corresponds to the use of the GFT with the weighted spatial graph proposed in this article.
- **Weighted Graph + DSMH:** it corresponds to the approach presented in this article, i.e. double sparsity method with a weighted spatial graph.

4.3.4 Calibration & parameters

The DSMH method is applied with $N_w = 5$ wavelets and parameters $s_1 = 3, s_2 = 5$, and $L = 10$. The spatio-temporal graph is constructed with a temporal line composed of 2 nodes. Concerning the construction of the features, the value of the maximum pyramid level for the TPM procedure is set to $M = 2$ whatever the Recognition method and the regularization parameter of the SVM classifier is left equal to 1.

[2]One of the body joints is taken as a reference [49], so that the processed graph signals are vectors containing the distance of each joint from the fixed one.

[3]In addition we separate the positive and negative activations for each *super-atom* so that we double the number of *super atoms* in our dictionary.

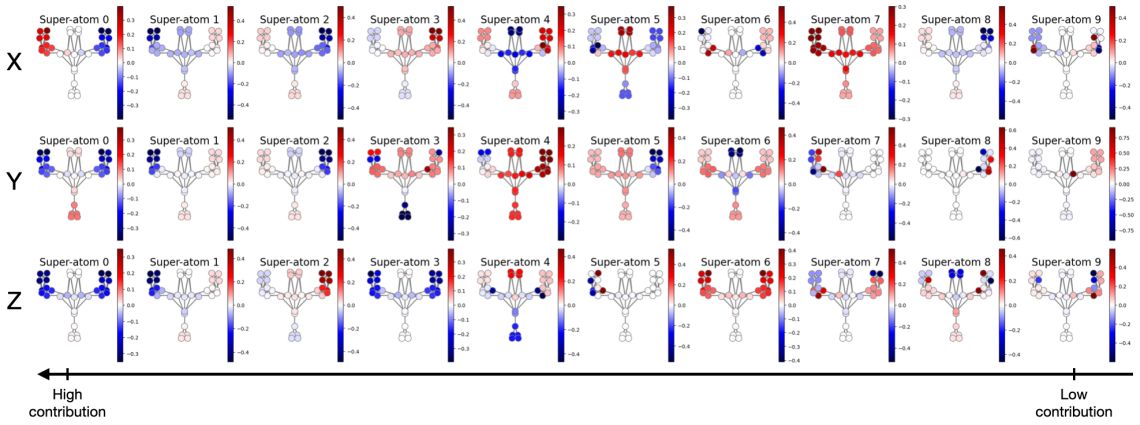


Figure 4: DSMH dictionaries obtained to reconstruct the matrices Y_x, Y_y, Y_z in the case of elevation movements of the right arm, left arm and both arms in the scapular plane. The parameters chosen are the following : $s_1 = 3$, $s_2 = 5$ and $L = 10$. Each line corresponds to a space dimension and the *super-atoms* are ordered by decreasing total contribution to the reconstruction.

5 Results and discussion

We present here the results obtained for the 3 experiments detailed in the previous section. The first part is dedicated to the analysis of upper limb elevations from the Arm-CODA dataset. In particular we display the DSMH dictionary and the timelines obtained on this database, and we assess the robustness of the method with Inter/Intra-subjects distances. Then, we evaluate the performance of the DSMH method on the denoising task and the HAR task.

5.1 Experiment 1: Analysis of upper limb elevations

5.1.1 DSMH dictionary

Figure 4 shows the *super-atoms* obtained by applying the DSMH method on elevation movements of the right arm, the left arm and both arms in the scapular plane. We have a DSMH dictionary of size 10 for each dimension of space, and each *super-atom* is a graph signal with colors indicating the velocity of the different body joints. We recall that the z-axis is oriented upward so that the elevation movement is mostly performed along this axis. At each instant, a combination of these *super-atoms* allows to approximate the velocity profile along a given direction of the space. For this reason, we will also use the term *behavioral atoms* to designate the *super-atoms* of the DSMH dictionary.

In Figure 4, the *super-atoms* are ordered by decreasing activation percentage, so that the first *super-atoms* are the ones that contribute the most to the signal reconstruction on the whole database. To know the value of the activation percentage for each *super-atom*, we can refer to the Figure 5, where an activation histogram is plotted for each dimension of space.

We start by looking at the *super-atoms* that are the most used on the whole database. Along the z-axis, we notice that the *super-atoms* 0, 1 and 2 have a particularly important contribution compared to the rest of the dictionary (Fig 5). This remark is also valid for the *super-atoms*

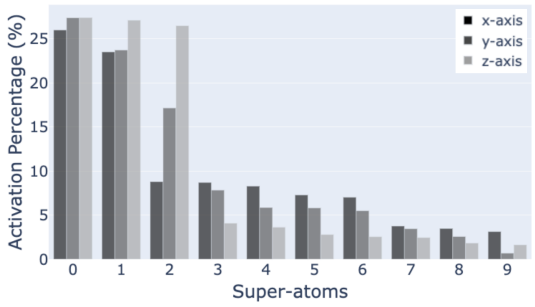


Figure 5: Activation percentage of each *super-atom* along the x, y and z-axis.

corresponding to the y-axis. For the x-axis it is especially *super-atoms* 0 and 1 that stand out from the rest of the dictionary.

We can look at these *super-atoms* in more detail by studying the corresponding graph signals in Figure 4. Along the z-axis, the *super-atom* 0 is a symmetrical atom with the two arms colored in blue, which correspond to a negative velocity signal. This bilateral *super-atom* can typically be used to reconstruct an elevation of both arms, with a negative activation during the ascending phase and a positive activation during the descending phase. Concerning the *super-atoms* 1 and 2, we can describe them as unilateral *super-atoms* as there is a higher velocity signal on one of the two arm. Thus, the *super-atom* 1 could for example be used to reconstruct a right arm movement, while the *super-atom* 2 could be used to approximate a left arm elevation. Similar observations can be made for the most used *super-atoms* along the x and y-axis. This first result is consistent with the analyzed movements which include right arm, left arm, and both arm elevations.

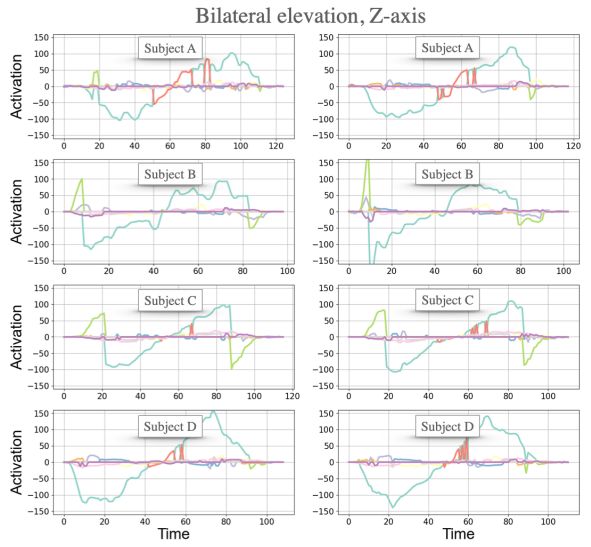
Some *super-atoms* are similar to the most used *super-atoms* described above. For example, the *super-atoms* 3 and 6 for the z-axis are bilateral graph signals just like the *super-atom* 0. Nevertheless, these 3 graph signals are not identical and present differences in the velocity intensity on the nodes located on the forearms. These subtle differences should be noted because with such a small dictionary size the obtained *super-atoms* are constructed in order to approximate the signal as well as possible, and must therefore characterize significant phenomena.

Among the other *behavioral atoms* in the DSMH dictionary, some have very localized velocity signals on the graph, i.e. only a few nodes carry a significant velocity. We can for example mention *super-atoms* 6 and 9 for the x-axis, *super-atoms* 8 and 9 for the y-axis as well as *super-atoms* 5, 8 and 9 for the z-axis. Some of them, such as *super-atom* 9 for the y-direction, have a localized velocity signal on a single node located on the left clavicle. These *super-atoms* are linked to outliers in the dataset because it corresponds to sensors that are sometimes hidden from the cameras. In the Figure 5, we can see that the *super-atom* 9 along the y-axis has a much lower total contribution than the others because it is only used occasionally.

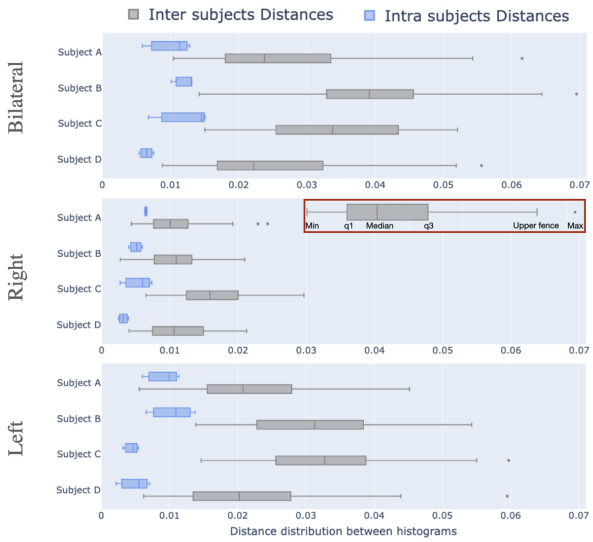
5.1.2 Timelines

In Figure 7, we have plotted timelines for subjects A, B, C and D whose characteristics are given in Table 1. Figure 7a indicates the most used *super-atoms* and Figure 7b indicates the 2^{nd} most used *super-atoms* over time.

First, we can note that the 1^{st} activation timelines (Fig. 7a) are quite similar from one subject

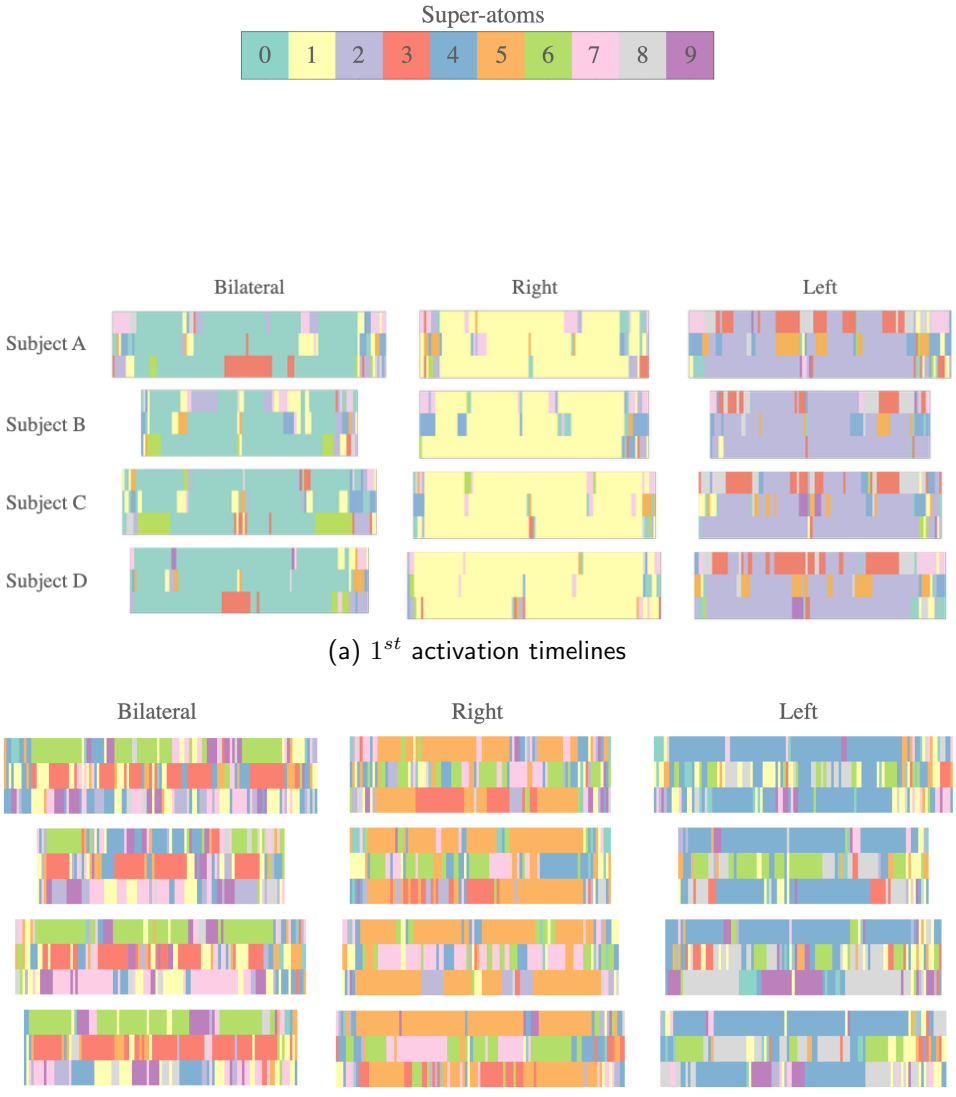


(a)



(b)

Figure 6: (a) Activation of each *super-atom* along the z-axis for a bilateral elevation. The 2 columns correspond to different repetitions of the same movement. As detailed on the legend above, each color is associated with a given *atom* of the DSMH dictionary (b) For each movement we have plotted boxes indicating the distribution of distances between the histograms of different motion sequences to measure the inter/intra subject variability. We have 3 figures corresponding to the 3 movements: bilateral elevation, right arm elevation and left arm elevation. For a given movement we have two boxes per subject: a blue one for the distances between the repetitions of the same movement for this subject, and an orange one for the distances between the movements performed by this subject and those performed by the 15 other subjects.



(a) 1^{st} activation timelines

(b) 2^{nd} activation timelines

Figure 7: Timelines indicating the *super-atoms* that contribute the most to the signal reconstruction over time for subject A, B, C and D. As explained on figure 3, each color in the legend corresponds to a given *super-atom*. (a) Timelines for the elevation movement of both arms, right arm and left arm, indicating the most used *super-atom* over time. (b) Timelines for the elevation movement of both arms, right arm and left arm, indicating the 2^{nd} most used *super-atoms* over time.

to another and easily distinguishable for two different movements. For the 2^{nd} activation timelines (Fig. 7b), we may enter into a much finer analysis of the movement. We can notice that there are many more differences between subjects but the timelines associated with different movements can still be identified. This suggests that this second level of activation still allows us to capture phenomena specific to movements, and therefore deserved to be analyzed.

We can study in more detail the timelines and identify what seems to be common to all subjects. For the bilateral elevation, we can for example note that the most used *super-atoms* along the x, y, and z-axis are the *super-atoms* 0, which correspond to bilateral graph signals. The same kind of observation can be done for the right and the left arm elevations : the most used *super-atoms* are unilateral graph signals with velocity mostly on the left side for the left arm elevation and signal on the right side for the right arm elevation.

Then, by carefully studying these timelines we can also notice some important variations between subjects. Still regarding the bilateral elevation, subjects A and D use the *super-atom* 3 in red along the z-axis in the middle of the movement, i.e. when the arms are about to reach the maximal elevation and when it starts to go down. Subject B, and more particularly subject C, use *super-atom* 6 along the z-axis at the beginning and at the end of their movement. This *super-atom* is a bilateral graph signal similar to the *super-atom* 0, but the intensity of the velocity is different for the two sensors located at the end of the forearm. Thus, the use of the *super-atom* 6 could be associated with a rotation of the forearm.

To consider these differences as specific to each subject, we must verify that these patterns are found for several repetitions of the same movement. In Figure 6a, we have plotted the activations along the z-axis for 2 repetitions of the bilateral elevation performed by subjects A, B, C and D. Before looking at the differences between subjects it is important to note that the activations corresponding to the most used *super-atoms* are much larger than the other activations. Moreover, we can observe that whatever the type of movement, the most used *super-atom* remains globally the same over time especially when the arms are in motion. These two remarks suggest that the 1^{st} activation could be a kind of continuous component in the movement. Concerning the patterns specific to each subject, Figure 6a also clearly highlights the switches between the bilateral *super-atom* 0 and *super-atom* 3 in red or *super-atom* 6 in green. As on the 1^{st} activation timelines, subjects A and D switch to *super-atom* 3 in the middle of their motion while subject B continues to use *super-atom* 0. In the end, the patterns visible on the 1^{st} activation timelines are well noticeable on the different repetitions of the same movement in Figure 6a. This is encouraging regarding the robustness of the method and this suggests that we have identified significant differences between subjects.

5.1.3 Inter/Intra-subject distances

In order to better quantify the differences between subjects and to validate the robustness of our motion representations we have plotted on figure 6b the inter/intra-subject distances between the activation histograms presented in Section 4.1.2.

Concerning bilateral elevations, the intra-subject distances are all at least lower than the first quartile of the inter-subject distribution. It means that the features obtained for a subject who repeats the same movement twice are closer than the features of two movements performed by different subjects. This result suggests that the method to build our motion representations is robust. On the other hand, these box plots allow us to better quantify the differences between subjects. For example, subject B, for which we had noted some differences with the other subjects

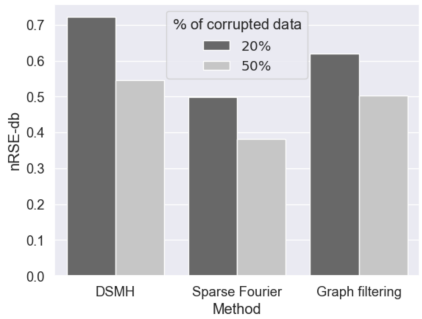


Figure 8: nRSE-db obtained for 20% and 50% of corrupted samples with the following 3 methods: the double sparsity method ($s_1 = 5$, $s_2 = 7$, $L = 10$), the graph filtering method ($\alpha = 0.39$), and the sparse Fourier method ($s_1 = 5$).

(use of *super-atoms* 1 and 2 in the first activation and use of *super-atom* 4 in the second activation along the x-axis), seems to be the one that differs the most from the other subjects. Indeed, the upper fence and the median of the distance distribution between this subject and the other subjects are higher than for subjects A, C, and D. Finally, we can also note that subject D seems to have very good repeatability whatever the movement.

5.2 Experiment 2: Denoising

In this Section, we evaluate the performance of the double sparsity method for denoising skeleton-based motion data. Figure 8 shows the nRSE-db obtained on signals with a percentage of noisy data equal to 20% and 50%.

The DSMH method performs slightly better than the graph filtering method [46] regardless of the noisy data percentage. As for the sparse Fourier method [45], its performance is lower than the other two methods.

These results show that the adaptive dictionary that is built is suitable to model the dominant phenomena in the motion sequences. In fact, it is specific enough to adapt to the motion sequences and reconstruct well the signal.

5.3 Experiment 3: Human Action Recognition

In this section, we evaluate the performance of the proposed method to discriminate between different motions on 3 different Action 3D datasets. Table 2 compares the 4 handcrafted Recognition methods presented in Section 4.3.3.

On the UTK database, the DSMH method performs better whatever the graph used. In particular, the DSMH method with a weighted graph achieves the best performance with 96% accuracy against 95% for the method inspired by the article [24]. Thus, even if the main purpose of the DSMH method is not to tackle the HAR task, we show that it still allows to capture motion characteristics that are generic enough to effectively discriminate between different actions. Appart from that, we can also notice that the double sparsity method gives better results with a weighted graph than with a spatio-temporal graph, while it is the opposite for the GFT method.

Recognition Method	UTK	MSR	F3D
Spatio-temporal Graph + GFT	95,00 %	71,45 %	**82,63** %
Spatio-temporal Graph + DSMH	95,50 %	70,14 %	78,12 %
Weighted Graph + GFT	94,97 %	70,78 %	79,76 %
Weighted Graph + DSMH	**96,00** %	**71,94** %	82,38 %

Table 2: Accuracy of the different Recognition methods on the HAR task

Concerning the MSR database, the performance of the DSMH method with a weighted graph is also slightly better with 71.94% accuracy against 71.45% for the GFT method with a spatio-temporal graph.

Finally, the performances on the F3D database are slightly higher by 0.25 % for the method proposed in [24] compared to the method proposed in this paper. As with the 2 other databases, the performance of the DSMH method is better using a weighted spatial graph, while the performance of the GFT is better with a spatio-temporal graph.

In the end, we have shown that the DSMH method provides comparable performance to other handcrafted recognition methods, although it was not specifically designed for the HAR task.

We also include a comparison with 4 deep-learning methods based on GCNs: ST-GCN (2018) [11], Deep STGC$_K$ (2018) [51], GR-GCN (2019) [52] and shift-GCN (2020) [53]. Each of these methods has its own characteristics, but they all rely on a spatio-temporal graph and a GCN to model dependencies and classify actions. The comparison is firstly done on the three datasets used previously. The results presented in Table 3 are reported from the literature and it should be noted that validation scheme is not always the same as the one we used in our study. As deep learning methods require more training data to achieve better performance, they are often employed with data augmentation procedures. This is the case for the methods GR-GCN and Deep STGC$_K$, which outperform the DSMH approach on the UTK and F3D datasets. However, without data augmentation, such methods may perform significantly worse. This is demontrated by the ST-GCN's accuracy of 27.64% on the MSR dataset with cross-subject validation, while the DSMH method has an accuracy of 71,94 % with a LOOCV validation scheme.

Additionally, we tested our DSMH approach on the ntu_cs_mini dataset introduced in [15]. The latter is a small database extracted from the NTU RGB+D [54], that contains 4 subjects performing 6 daily actions. The ST-GCN and Shift-GCN methods achieve higher performance than the DSMH approach (with parameters $s_1 = 10,\ s_2 = 12,\ L = 15$). However, comparing these results remains challenging due to the significant difference in parameter count. In fact, these two GCNs methods use 220k and 3M parameters, while our approach only uses around 100 parameters.

6 Conclusion

The main results of this article can be summarized as follows. We have proposed the use of the double sparsity method within a Graph signal processing framework for skeleton-based motion data analysis. We have shown that this approach can be used on real data of upper limb elevation from the Arm-CODA database. The DSMH dictionary and the timelines obtained showed the interest of the method to analyse the human motion. In addition, the inter/intra subject distances computed from the activation vectors have highlighted the robustness of the proposed method, and have suggested that the movement could be composed of a "core" and a "style". Finally, we have

Recognition Method	**UTK**	**MSR**	**F3D**	ntu_cs_mini	# Param.
ST-GCN [11, 14, 15]	-	27.64 (CS)	-	71,53 (CS)	3.08M
GR-GCN [52]	98,5	-	98,4	-	
Deep STGC$_K$ [51]	-	-	99,1	-	
shift-GCN [15, 53]	-	-	-	60.00 (CS)	0.22M
DSMH	96,5	71,94	82.38	56,25 (CS)	$\sim 10^2$

Table 3: Reported accuracies of GCN Deep-learning methods on the HAR task. Results in grey were obtained with data augmentation procedures. All the accuracies correspond to a LOOCV validation scheme, except for the ntu_cs_mini dataset and for the ST-GCN method on the MSR dataset for which a Cross-Subject (CS) validation was used.

illustrated two possible applications of the method on "real-world" problems: The denoising task for which we have obtained better results than a state-of-the-art graph filtering method, and the human action recognition task for which we have obtained results comparable to those of state-of-the-art methods.

The motion representation proposed in this article can be dedicated to different types of analysis. First, histograms can be used to quantitatively compare movements. The computation of inter/intra subject distances was important to evaluate the robustness of the method but this metric can also be used to perform inter-individual comparisons. Concerning clinical applications, we can think about the early detection of pathological movements, as well as longitudinal studies and patient follow-up [2, 55, 56]. Moreover, the timelines can be used to make qualitative comparisons between subjects. These features have the advantage of being compact while being very informative. The temporal information provided by the timeline is crucial to analyze the human motion. It can be used to identify patterns that are present in the motion sequences of different subjects. The fact that a given pattern is noticeable on the different repetitions of the same elevation for a given subject and not for others suggests that we have identified different movement strategies. In the end, both the histograms and the timelines suggest that there are similarities between movements performed by different subjects. Thus, it would be interesting to go further in the study of these motion representations, in order to answer more fundamental questions about human motion, in particular concerning the "style" [17–20].

Acknowledgments

The authors would like to thank P.P. Vidal, D. Wang, A. Roren, D. Vaquero-Ramos, and M.-M. Lefèvre-Colau, as well as the BioMedTech Facilities INSERM US36 — CNRS UMS2009 — Université de Paris for the data collection.

References

[1] Q. Wang and Y. Rao, "Visual analysis of human motion: A survey on recent advances and applications," in *2018 IEEE Visual Communications and Image Processing (VCIP)*, 2018, pp. 1–4.

[2] Z. Kertesz and I. Lovanyi, "3d motion capture methods for pathological and non-pathological human motion analysis," in *2006 2nd International Conference on Information & Communication Technologies*, vol. 1, 2006, pp. 1062–1067.

[3] S. Tanaka, T. Wada, and K. Kawakita, "Determination of human motion for rehabilitation based on time-scale transformation," in *2007 International Conference on Mechatronics and Automation*, 2007, pp. 2160–2165.

[4] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake, "Real-time human pose recognition in parts from single depth images," in *CVPR 2011*, 2011, pp. 1297–1304.

[5] E. Hegarini, A. B. Mutiara, A. Suhendra, M. Iqbal, and B. A. Wardijono, "Similarity analysis of motion based on motion capture technology," in *2016 International Conference on Informatics and Computing (ICIC)*, 2016, pp. 389–393.

[6] P. Glardon, R. Boulic, and D. Thalmann, "Pca-based walking engine using motion capture data," in *Proceedings Computer Graphics International, 2004.*, 2004, pp. 292–298.

[7] L. Xia, C.-C. Chen, and J. K. Aggarwal, "View invariant human action recognition using histograms of 3d joints," in *2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, 2012, pp. 20–27.

[8] L. Shi, Y. Zhang, J. Cheng, and H. Lu, "Adasgn: Adapting joint number and model size for efficient skeleton-based action recognition," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 13 413–13 422.

[9] C. Dai, Y. Wei, Z. Xu, M. Chen, Y. Liu, and J. Fan, "An investigation of gcn-based human action recognition using skeletal features," in *2022 27th International Conference on Automation and Computing (ICAC)*. IEEE, 2022, pp. 1–10.

[10] Z. Sun, Q. Ke, H. Rahmani, M. Bennamoun, G. Wang, and J. Liu, "Human action recognition from various data modalities: A review," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–20, 2022.

[11] S. Yan, Y. Xiong, and D. Lin, "Spatial temporal graph convolutional networks for skeleton-based action recognition," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 32, no. 1, 2018.

[12] L. Shi, Y. Zhang, J. Cheng, and H. Lu, "Two-stream adaptive graph convolutional networks for skeleton-based action recognition," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 12 026–12 035.

[13] Y.-F. Song, Z. Zhang, C. Shan, and L. Wang, "Constructing stronger and faster baselines for skeleton-based action recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 45, no. 2, pp. 1474–1488, 2022.

[14] L. Wang, D. Q. Huynh, and P. Koniusz, "A comparative review of recent kinect-based action recognition algorithms," *IEEE Transactions on Image Processing*, vol. 29, pp. 15–28, 2019.

[15] L. Feng, Y. Zhao, W. Zhao, and J. Tang, "A comparative review of graph convolutional networks for human skeleton-based action recognition," *Artificial Intelligence Review*, pp. 1–31, 2022.

[16] G. Johansson, "Visual perception of biological motion and a model for its analysis," *Perception & psychophysics*, vol. 14, no. 2, pp. 201–211, 1973.

[17] J. E. Cutting and L. T. Kozlowski, "Recognizing friends by their walk: Gait perception without familiarity cues," *Bulletin of the psychonomic society*, vol. 9, no. 5, pp. 353–356, 1977.

[18] S. V. Stevenage, M. S. Nixon, and K. Vince, "Visual analysis of gait as a cue to identity," *Applied Cognitive Psychology: The Official Journal of the Society for Applied Research in Memory and Cognition*, vol. 13, no. 6, pp. 513–526, 1999.

[19] S. Xia, "Modeling style and variation in human motion," in *2010 4th International Universal Communication Symposium*, 2010, pp. 207–207.

[20] M. Vasilescu, "Human motion signatures: analysis, synthesis, recognition," in *2002 International Conference on Pattern Recognition*, vol. 3, 2002, pp. 456–460 vol.3.

[21] X. Dong, D. Thanou, L. Toni, M. Bronstein, and P. Frossard, "Graph signal processing for machine learning: A review and new perspectives," *IEEE Signal Processing Magazine*, vol. 37, no. 6, pp. 117–127, nov 2020. [Online]. Available: https://doi.org/10.1109%2Fmsp.2020.3014591

[22] A. Ortega, P. Frossard, J. Kovačević, J. M. F. Moura, and P. Vandergheynst, "Graph signal processing: Overview, challenges and applications," 2017. [Online]. Available: https://arxiv.org/abs/1712.00468

[23] J.-Y. Kao, A. Ortega, and S. S. Narayanan, "Graph-based approach for motion capture data representation and analysis," in *2014 IEEE International Conference on Image Processing (ICIP)*, 2014, pp. 2061–2065.

[24] J.-Y. Kao, A. Ortega, D. Tian, H. Mansour, and A. Vetro, "Graph based skeleton modeling for human activity analysis," in *2019 IEEE International Conference on Image Processing (ICIP)*, 2019, pp. 2025–2029.

[25] D. K. Hammond, P. Vandergheynst, and R. Gribonval, "Wavelets on graphs via spectral graph theory," 2009. [Online]. Available: https://arxiv.org/abs/0912.3848

[26] Y. Yankelevsky and M. Elad, "Dictionary learning for high dimensional graph signals," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 4669–4673.

[27] R. Rubinstein, M. Zibulevsky, and M. Elad, "Double sparsity: Learning sparse dictionaries for sparse signal approximation," *IEEE Transactions on Signal Processing*, vol. 58, no. 3, pp. 1553–1564, 2010.

[28] E. Bullmore and O. Sporns, "The economy of brain network organization," *Nature reviews neuroscience*, vol. 13, no. 5, pp. 336–349, 2012.

[29] N. Tremblay and P. Borgnat, "Graph wavelets for multiscale community mining," *IEEE Transactions on Signal Processing*, vol. 62, no. 20, pp. 5227–5239, 2014.

[30] C. Wu, X.-J. Wu, and J. Kittler, "Spatial residual layer and dense connection block enhanced spatial temporal graph convolutional network for skeleton-based action recognition," in *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, 2019, pp. 1740–1748.

[31] P. Humbert, B. L. Bars, L. Oudre, A. Kalogeratos, and N. Vayatis, "Learning laplacian matrix from graph signals with sparse spectral representation," *Journal of Machine Learning Research*, vol. 22, no. 195, pp. 1–47, 2021. [Online]. Available: http://jmlr.org/papers/v22/19-944.html

[32] X. Dong, D. Thanou, P. Frossard, and P. Vandergheynst, "Learning laplacian matrix in smooth graph signal representations," *IEEE Transactions on Signal Processing*, vol. 64, no. 23, pp. 6160–6173, 2016.

[33] S. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Transactions on Signal Processing*, vol. 41, no. 12, pp. 3397–3415, 1993.

[34] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the royal statistical society series b-methodological*, vol. 58, pp. 267–288, 1996.

[35] S. Mallat, *A wavelet tour of signal processing (2. ed.).* Academic Press, 1999.

[36] M. Gavish, B. Nadler, and R. R. Coifman, "Multiscale wavelets on trees, graphs and high dimensional data: Theory and applications to semi supervised learning," in *Proceedings of the 27th International Conference on International Conference on Machine Learning*, ser. ICML'10. Madison, WI, USA: Omnipress, 2010, p. 367–374.

[37] A. Sandryhaila and J. M. F. Moura, "Discrete signal processing on graphs: Graph fourier transform," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013, pp. 6167–6170.

[38] K. Engan, S. Aase, and J. Hakon Husoy, "Method of optimal directions for frame design," in *1999 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings. ICASSP99 (Cat. No.99CH36258)*, vol. 5, 1999, pp. 2443–2446 vol.5.

[39] M. Aharon, M. Elad, and A. Bruckstein, "K-svd: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4311–4322, 2006.

[40] D. Thanou, D. I. Shuman, and P. Frossard, "Learning parametric dictionaries for signals on graphs," *IEEE Transactions on Signal Processing*, vol. 62, no. 15, pp. 3849–3862, 2014.

[41] X. Zhang, X. Dong, and P. Frossard, "Learning of structured graph dictionaries," in *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2012, pp. 3373–3376.

[42] R. Coifman and M. Wickerhauser, "Entropy-based algorithms for best basis selection," *IEEE Transactions on Information Theory*, vol. 38, no. 2, pp. 713–718, 1992.

[43] S. Combettes, P. Boniol, A. Mazarguil, D. Wang, D. Vaquero-Ramos, M. Chauveau, L. Oudre, N. Vayatis, P.-P. Vidal, A. Roren, and M.-M. Lefèvre-Colau, "Arm-coda: A dataset of upper-limb human movement during routine examination," *submitted to Image Processing On Line (IPOL)*, 2023.

[44] B. Fuglede and F. Topsoe, "Jensen-shannon divergence and hilbert space embedding," in *International Symposium onInformation Theory, 2004. ISIT 2004. Proceedings.*, 2004, pp. 31–.

[45] Y. Pati, R. Rezaiifar, and P. Krishnaprasad, "Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition," in *Proceedings of 27th Asilomar Conference on Signals, Systems and Computers*, 1993, pp. 40–44 vol.1.

[46] S. Chen, A. Sandryhaila, J. M. F. Moura, and J. Kovacevic, "Signal denoising on graphs via graph filtering," in *2014 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, 2014, pp. 872–876.

[47] W. Li, Z. Zhang, and Z. Liu, "Action recognition based on a bag of 3d points," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops*, 2010, pp. 9–14.

[48] L. Seidenari, V. Varano, S. Berretti, A. Del Bimbo, and P. Pala, "Recognizing actions from depth cameras as weakly aligned multi-part bag-of-poses," in *2013 IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2013, pp. 479–485.

[49] G. Paoletti, J. Cavazza, C. Beyan, and A. Del Bue, "Subspace clustering for action recognition with covariance representations and temporal pruning," in *2020 25th International Conference on Pattern Recognition (ICPR)*, 2021, pp. 6035–6042.

[50] P. Wang, Y. Cao, C. Shen, L. Liu, and H. T. Shen, "Temporal pyramid pooling-based convolutional neural network for action recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, no. 12, pp. 2613–2622, 2017.

[51] L. Chaolong, C. Zhen, Z. Wenming, X. Chunyan, and Y. Jian, "Spatio-temporal graph convolution for skeleton based action recognition," in *Thirty-Second AAAI Conference on Artificial Intelligence*, vol. 2, 2018.

[52] X. Gao, W. Hu, J. Tang, J. Liu, and Z. Guo, "Optimized skeleton-based action recognition via sparsified graph regression," in *Proceedings of the 27th ACM international conference on multimedia*, 2019, pp. 601–610.

[53] K. Cheng, Y. Zhang, X. He, W. Chen, J. Cheng, and H. Lu, "Skeleton-based action recognition with shift graph convolutional network," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 183–192.

[54] A. Shahroudy, J. Liu, T.-T. Ng, and G. Wang, "Ntu rgb+ d: A large scale dataset for 3d human activity analysis," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1010–1019.

[55] A. Franklyn-Miller, C. Richter, E. King, S. Gore, K. Moran, S. Strike, and E. Falvey, "Athletic groin pain (part 2): a prospective cohort study on the biomechanical evaluation of change of direction identifies three clusters of movement patterns," *British journal of sports medicine*, vol. 51, no. 5, pp. 460–468, 2017.

[56] P. E. Taylor, G. J. Almeida, J. K. Hodgins, and T. Kanade, "Multi-label classification for the analysis of human motion quality," in *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2012, pp. 2214–2218.