Graph dictionary learning for the study of human motion

Marion Chauveau, Antoine Mazarguil and Laurent Oudre
Université Paris Saclay, Université Paris Cité, ENS Paris Saclay, CNRS, SSA, INSERM, Centre Borelli,
F-91190, Gif-sur-Yvette, France.
surname.name@ens-paris-saclay.fr

***Abstract*—In this article, we introduce a method inspired by Graph Signal Processing (GSP) for the analysis of human motion based on the 3D positions of skeletal joints. Our approach uses a graph dictionary learning technique, in which each velocity sample is decomposed into a linear combination of a limited set of atoms acquired directly from the data. The efficacy of this methodology is evaluated using a dataset focused on upper limb elevations. We present features and visualizations, and validate the robustness of the approach through the construction of inter- and intra-subject distances. The features are also used as inputs for Human Activity Recognition with competitive results. The interpretability of the features and visualizations obtained from this method make it suitable for applications such as inter-individual comparisons or longitudinal follow-up of patients.**

***Index Terms*—Dictionary learning, Graph Signal Processing (GSP), Human Motion Analysis**

I. Introduction

The exploration of human motion analysis is an intriguing area of research with a wide array of applications, spanning from video surveillance and human-machine interaction to diagnostic aid and medical rehabilitation. In recent times, the use of skeleton-based motion data has grown significantly, in particular the 3D positions of various skeletal joints tracked over time. This approach has shown significant potential for extracting useful information about human movements.

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 [1]–[4]. However, these models are often highly complex, time-consuming to train and require to work with large datasets [5]. Deep-learning approaches also tend to be task-specific, and features acquired in a supervised manner can be challenging to interpret. This becomes problematic when the focus extends beyond merely identifying actions to studying how they are executed and understanding the common and individual characteristics of human movement. In such investigations, it is crucial for features to be both discriminative and interpretable, facilitating the identification of similarities and differences in actions performed by distinct subjects.

Given these considerations, the methodology introduced in this article diverges from relying on deep learning techniques and instead adopts the Graph Signal Processing (GSP) framework [6]. Similar to GCNs, GSP methods operate under the assumption that skeletal structure is encoded as a graph, reflecting the proximity between body joints. However, distinct from utilizing "black boxes" like convolutional networks, tools derived from the GSP framework offer the intriguing property of being directly interpretable. This stems from the fact that most notions defined in this framework are extensions of standard signal processing tools to irregular domains, including filtering, sampling, Fourier transform, and sparsity. In the GSP framework, 3D skeleton data are simply regarded as "graph signals" lying on the graph. In this article, we introduce the first application (up to our knowledge) of GSP dictionary learning for human motion data. By leveraging the advantages of tools from the GSP framework and dictionary learning methods, we develop an analysis approach that combines simplicity, interpretability and versatility.

II. Background

A. The GSP framework

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 N 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.

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 $\mathbf{f} = [f_1, ..., f_N] \in \mathbb{R}^N$.

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 Graph Fourier Transform (GFT) as $\tilde{\mathbf{y}} = U^T \mathbf{y}$ where $\tilde{\mathbf{y}}$ contains the energies associated with each frequency.

B. GSP dictionary learning

Provided a collection of n graph signals $\mathbf{y_1}, \ldots, \mathbf{y_n}$, the aim of GSP dictionary learning techniques is to decompose these graph signals on a set of L vectors $(\mathbf{d_1}, ..., \mathbf{d_L})$ called *atoms*. The decomposition of a given graph signal $\mathbf{y} \in \mathbb{R}^N$ can be written as follows:

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

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, or using convex relaxation methods.

The atoms can either be fixed or learned from the data. For instance, Kao et al. [7], [8] use the graph Fourier basis, wherein atoms are predetermined, to investigate human motion and address the gesture recognition task. The use of an analytical dictionary has the advantage of being numerically fast, but it can also be poorly adapted to the data studied. Alternatively, atoms can be directly inferred from the data using training algorithms such as the method of optimal directions or the K-SVD algorithm. This approach is often numerically more expensive, but it allows to obtain dictionaries better adapted to the data. If we need a dictionary that combines the advantages of analytical and learned dictionaries, it is also possible to use hybrid approaches that search for atoms constructed as linear combinations of analytical atoms [9].

III. Method

Our method is composed of three steps:

1) Graph and graph signals construction from the raw data
2) Creation of the analytical graph Mexican Hat wavelets basis [10]
3) Hybrid GSP dictionary learning using the double sparsity approach [9]

The method takes as input a dataset of 3D velocity time series and output the learned atoms and the time activations.

A. Data

We use the Arm-CODA [11] dataset which is composed of 143 motion sequences captured with 34 Cartesian Opto-electronic Dynamic Anthropometer (CODA) motion system 3D position markers, from a cohort of 16 healthy subjects. The 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 each subject.

B. Graph and graph signals construction

a) Graph construction: For this study, 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:

$$w_{i,j} \propto e^{-d_{max}^{i,j}} \tag{2}$$

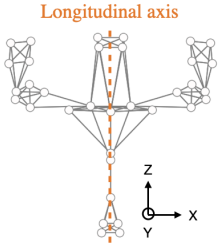

Fig. 1. Weighted and undirected k-nn Graph constructed from the Arm-CODA dataset with $k = 5$. Each node is represented with a white circle and the edges are the lines connecting these nodes.

Afterward, the generated graph undergoes simplification to form a k-nearest neighbors (k-nn) graph. Specifically, for each joint, only the edges connecting it to its k closest neighbors are retained. The value of k is set to the smallest required for maintaining a connected path between every pair of nodes, ensuring graph connectivity. Subsequently, we enforce symmetry with respect to the longitudinal axis by computing the mean of weights associated with symmetrical edges. Figure 1 provides a visual representation of the constructed graph using the Arm-CODA dataset with $k = 5$.

b) Graph signals construction: From the 3D positions of each skeletal joint, we calculate the first time derivative to obtain the velocity signals. Each multivariate time sample can be considered as 3 distinct graph signals (one per dimension). All these graph signals, represented as vectors, are then stored on the columns of three matrices $\mathbf{Y}^{(d)} \in \mathbb{R}^{N \times T}$, $d \in \{x, y, z\}$.

C. Graph Mexican hat wavelets basis

We first construct a set of analytical atoms derived from Mexican hat wavelets. The coefficients of the wavelet $\psi_{\beta,i} \in \mathbb{R}^N$, at scale β and centered on node i, are obtained with the following formula:

$$\psi_{\beta,i}(m) = \sum_{l=0}^{N-1} g(\lambda_l) \hat{\delta}_i(l) \mathbf{u}_l(m) \tag{3}$$

with $\mathbf{u}_l(m)$ the m^{th} coefficient of the eigenvector associated to the eigenvalue λ_l of the Laplacian, and $\hat{\delta}_i$ the GFT of the Dirac graph signal located on node i.

In the following, we will use two different kernels:

- A band-pass filter $g_1(\beta x) = \beta x \times e^{-\beta x}$ with scale parameter β.
- A low-pass filter $g_2(x) = \gamma e^{-\left(\frac{20}{0.4\lambda_{max}} x\right)^4}$, with $\gamma = 1.2 \times e^{-1}$, that captures low-frequency phenomena.

In the final atom set, we consider $N = 34$ impulse signals, which are Diracs located on each of the N nodes of the graph, and 5 different kernels (one low-pass and four band-pass with different scales), leading to a total of $K = 5 \times 34 = 170$ wavelets. These wavelets, which have different scales and are

centered on different graph nodes, enable us to account for phenomena with different ranges and locations.

D. GSP dictionary learning

In this paper, we use a double sparsity approach [9], [12] that will allow us to learn a set of *super-atoms* using Mexicat hat analytic atom combinations. The final dictionary is defined as a product $\mathbf{D} = \mathbf{\Phi A}$, where $\mathbf{\Phi} \in \mathbb{R}^{N \times K}$ is the fixed dictionary containing the Mexican hat *atoms* in its columns, and $\mathbf{A} \in \mathbb{R}^{K \times L}$ is a learned sparse matrix.

Our Double Sparsity Mexican Hat (DSMH) dictionary learning algorithm learns a set of *super-atoms* $(\mathbf{d_1}, \ldots, \mathbf{d_L})$, defined as linear combinations of Mexican hat *atoms* in $\mathbf{\Phi} = (\boldsymbol{\phi_1}, \ldots, \boldsymbol{\phi_K})$:

$$\mathbf{d_l} \approx \sum_{k=1}^{K} a_k \boldsymbol{\phi_k} \tag{4}$$

where $\mathbf{a} = (a_1, ..., a_K)$ is a sparse vector containing the weights of the linear combination.

These weights are learned from the graph signals $\mathbf{Y}$ by solving the optimization problem:

$$\begin{aligned} \underset{\mathbf{A},\mathbf{X}}{\operatorname{argmin}} \quad & ||\mathbf{Y} - \mathbf{\Phi A X}||_F^2 \\ \text{s.t.} \quad & ||\mathbf{x}_i||_0 \leq s_1 \qquad \forall i, \\ & ||\mathbf{a}_j||_0 \leq s_2 \qquad \forall j, \qquad ||\mathbf{\Phi a}_j||_2 = 1 \qquad \forall j. \end{aligned} \tag{5}$$

where s_2 is the maximum number of *atoms* combined to create a *super-atom* and s_1 is the maximum number of *super-atom* used to approximate each graph signal. $\mathbf{X} \in \mathbb{R}^{L \times T}$ is the activation matrix containing the contribution of the *super-atoms* to the signal reconstruction. This problem is addressed using a greedy procedure described in [12], which is based on the Matching Pursuit algorithm.

IV. Results

Several outcomes from our DSMH approach can be extracted for the study of human locomotion:

- The learned *super-atoms* $\mathbf{D}$ are displayed on Figure 2. The parameters chosen are the following : $s_1 = 3$, $s_2 = 5$ and $L = 10$.
- The activations $\mathbf{X}$ provide an indication of the use of each *super-atom* over time. For sake of conciseness, we only display, for each movement, timelines that represents the most activated *super-atom* and the second most activated *super-atom*, on Figure 3.
- Two motion sequences can be compared using the histograms of the activations of each *super-atom*. For each motion sequence, we compute the contribution of each *super-atom* (in percentage) to construct an histogram defined as a probability vector. Next, we use the Jensen-Shannon divergence to calculate the distance between two histograms, in order to obtain a distance between two motion sequences.
- Similarly to [8] that use a fixed graph Fourier dictionary, the activations can be used as features for Human Action Recognition (HAR) through a temporal pyramid procedure (TPM) [13].

A. Interpretation of the super-atoms

In Figure 2, the learned *super-atoms* are ordered by decreasing percentage of activation, so that the first *super-atoms* are those that contribute most to signal reconstruction over the whole database.

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 (for x-, y- and z-axes), 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. In the end, the most used *super-atoms* are consistent with the analyzed movements which include right arm, left arm, and both arm elevations.

Some similarities between *super-atoms* can also be observed. 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 show differences in the intensity of the velocity on the nodes located on the forearms. These subtle differences are worth noting, because with such a small dictionary, the *super-atoms* obtained are constructed to approximate the signal as closely as possible, and must therefore characterize significant phenomena. Among the other *super-atoms*, 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-axis, 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.

B. Interpretation of the activations

Figure 3 shows the activations of the two *super-atoms* that contribute the most to the signal reconstruction over time for 4 subjects and 3 movements (elevation movement of both arms, right arm and left arm).

The 1^{st} activation timelines (top) are fairly similar from one subject to the next, and easily distinguishable for two different movements. By carefully studying these timelines we can also notice some important variations between subjects. 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

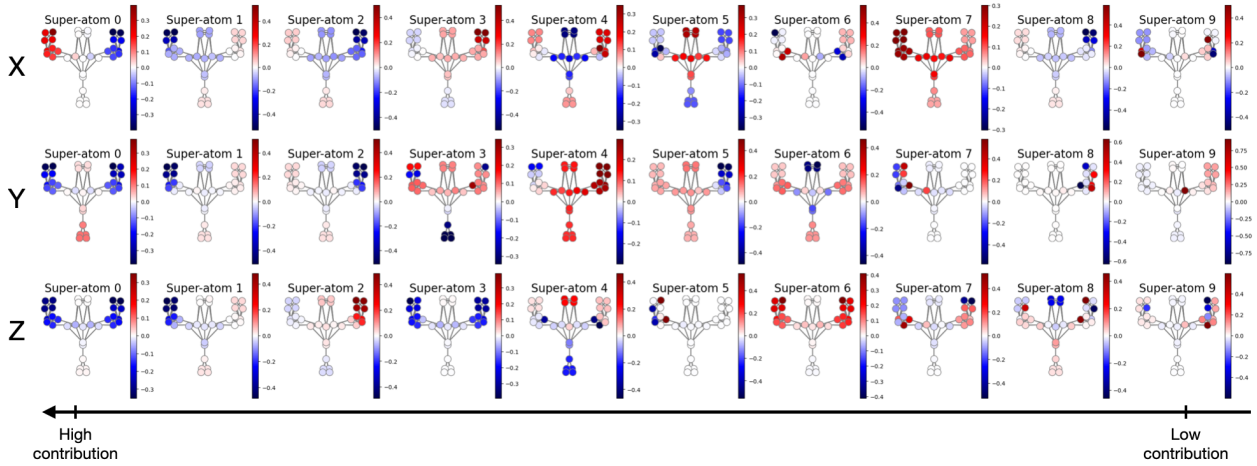

Fig. 2. DSMH dictionaries of $L = 10$ *super-atomes* obtained to reconstruct the matrices $Y^{(x)}, Y^{(y)}, Y^{(z)}$ in the case of elevation movements of the right arm, left arm and both arms in the scapular plane. Each line corresponds to a space dimension and the *super-atoms* are ordered by decreasing total contribution to the reconstruction.

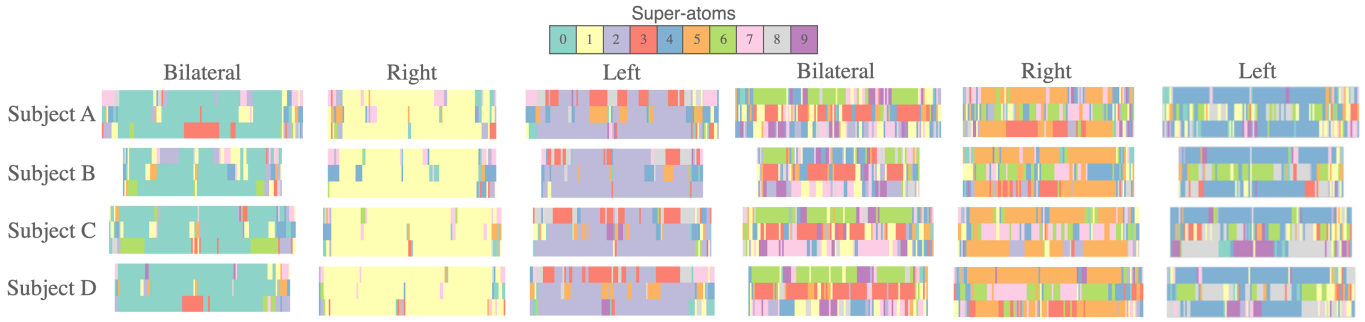

Fig. 3. Timelines indicating the *super-atoms* that contribute the most to the signal reconstruction over time for subject A, B, C and D (selected from the 16 subjects of the CODA database). These 4 subjects have the following characteristics [Sexe, Age, Size (cm), Weight (kg)]: A[F;47;170;65], B[M;57;173;75], C[F;52;156;64], D[M;28;179;73]. Each color in the legend corresponds to a given *super-atom*. (Top) Timelines for the elevation movement of both arms, right arm and left arm, indicating the most used *super-atom* over time. (Bottom) Timelines for the elevation movement of both arms, right arm and left arm, indicating the 2^{nd} most used *super-atoms* over time.

of the forearm. Thus, the use of the *super-atom* 6 could be associated with a rotation of the forearm.

For the 2^{nd} activation timelines (bottom), we can enter into a much finer analysis of movement. 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 is still capable of capturing movement-specific phenomena, and therefore merits further analysis.

In the end, the timelines have the advantage of being compact while being very informative. The temporal information provided by this representation is crucial to analyze the human motion. It can be used to identify patterns that are present in the motion sequences of different subjects.

C. 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 4 the inter/intra-subject distances (Jensen- Shannon divergence) between the activation histograms.

For bilateral elevations, the intra-subject distances are all at least below the first quartile of the inter-subject distribution. This means that the characteristics obtained for a subject repeating the same movement twice are closer than the characteristics of two movements performed by different subjects. We also note that subject D has very good repeatability whatever the movement, and that subject B seems to be the one who differs most from the other subjects. These results can be supplemented by the analysis of the timelines of Figure 3. For instance, Subject B uses *super-atoms* 1 and 2 in the first activation and *super-atom* 4 in the second activation along the x-axis for the bilateral elevation.

D. Application to Human Activity Recognition (HAR)

In this section, we evaluate the performance of the proposed features to discriminate between different motions on four different Action 3D datasets. Table I compares our method to one method based on GSP [8] and four methods using Graph Neural Networks (GNN). It should be noted that, unlike the two GSP methods, the latter are specifically dedicated to this task and rely on several million parameters.

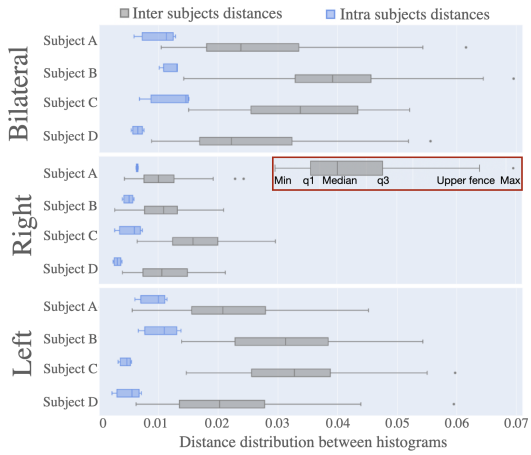

Fig. 4. For each movement, we plotted boxes indicating the distribution of distances between the histograms of the different movement sequences in order to measure inter/intra-subject variability. For a given movement, we have two boxes per subject: a blue one for the distances between repetitions of the same movement for that subject, and a grey one for the distances between movements performed by that subject and those performed by the other 15 subjects.

Recognition Method	**UTK** [14]	**MSR** [15]	**F3D** [16]	**ntu_cs_mini** [5] [17]
Kao et al. [8]	95.00	71.45	82.63	-
ST-GCN [1], [5], [18]	-	27.64 (CS)	-	71.53 (CS)
GR-GCN [3]	98.5*	-	98.4*	-
Deep STGC$_K$ [2]	-	-	99.1*	-
shift-GCN [4], [5]	-	-	-	60.00 (CS)
Our method	96.00	71.94	82.38	66.25

TABLE I
ACCURACY OF THE DIFFERENT RECOGNITION METHODS ON THE HAR TASK. 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. *RESULTS OBTAINED WITH DATA AUGMENTATION PROCEDURES.

For most datasets, the scores obtained by our method are consistent with state-of-the-art results. Although the main objective of the DSMH method is not to tackle the HAR task, it nevertheless captures motion features that are sufficiently generic to effectively discriminate between different actions. In addition, our approach performs slightly better than the alternative GSP method on UTK and MSR.

Deep learning techniques outperform GSP methods, but this comes at the cost of data augmentation procedures and/or a complex architecture with millions of parameters [5] compared to around a hundred for the DSMH method. What's more, without data augmentation, these methods can perform significantly less well. For example, the ST-GCN method has an accuracy of 27.64% on the MSR dataset with between-subjects validation, while the DSMH method has an accuracy of 71.94% with a LOOCV validation scheme.

V. CONCLUSION

The methodology presented in this article makes it possible to extract meaningful and useful visual representations and features that provide insights for the analysis of human movement. This approach could be used, for example, for longitudinal follow-up and inter-individual comparison.

ACKNOWLEDGMENT

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] 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.

[2] 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.

[3] 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.

[4] 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.

[5] 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.

[6] A. Ortega, P. Frossard, J. Kovačević, J. M. Moura, and P. Vandergheynst, "Graph signal processing: Overview, challenges, and applications," *Proceedings of the IEEE*, vol. 106, no. 5, pp. 808–828, 2018.

[7] 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)*. IEEE, 2014, pp. 2061–2065.

[8] 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)*. IEEE, 2019, pp. 2025–2029.

[9] 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, 2009.

[10] D. K. Hammond, P. Vandergheynst, and R. Gribonval, "Wavelets on graphs via spectral graph theory," *Applied and Computational Harmonic Analysis*, vol. 30, no. 2, pp. 129–150, 2011.

[11] S. W. 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 Data Set of Upper-limb Human Movement During Routine Examination," *Image Processing On Line*, vol. 14, pp. 1–13, 2024, https://doi.org/10.5201/ipol.2024.494.

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

[13] 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, 2016.

[14] 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*. IEEE, 2012, pp. 20–27.

[15] 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*. IEEE, 2010, pp. 9–14.

[16] L. Seidenari, V. Varano, S. Berretti, A. Bimbo, and P. Pala, "Recognizing actions from depth cameras as weakly aligned multi-part bag-of-poses," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2013, pp. 479–485.

[17] 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.

[18] 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.