Movement Analysis and Decomposition with the Continuous Wavelet Transform

Jules Françoise, Gabriel Meseguer-Brocal, Frédéric Bevilacqua

Movement Analysis and Decomposition with the Continuous Wavelet Transform

JULES FRANÇOISE, Université Paris-Saclay, CNRS, LISN, France
GABRIEL MESEGUER-BROCAL, STMS Lab, Ircam, CNRS, Sorbonne Université, France
FRÉDÉRIC BEVILACQUA, STMS Lab, Ircam, CNRS, Sorbonne Université, France

Human movements support communication, and can be used to imitate actions or physical phenomenons. Observing gestural imitations of short sounds, we found that such gestures can be categorized by their frequency content. To analyse such movements, we propose an analysis method based on wavelet analysis for clustering or recognizing movement characteristics. Our technique draws upon the continuous wavelet transform to derive a time-frequency representation of movement information. We propose several global descriptors based on statistical descriptors, frequency tracking, or non-negative matrix factorization, that can be used for recognition or clustering to highlight relevant movement qualities. Additionally, we propose a real-time implementation of the continuous wavelet transform based on a set of approximations, that enables its use in interactive applications. Our method is evaluated on a database of gestures co-executed with vocal imitations of recorded sounds.

CCS Concepts: • **Human-centered computing** → **Human computer interaction (HCI)**; **Gestural input**.

Additional Key Words and Phrases: Movement, Rhythmic Gesture, Wavelet, Continuous Wavelet Transform, Recognition, Vocalization

ACM Reference Format:
Jules Françoise, Gabriel Meseguer-brocal, and Frédéric Bevilacqua. 2022. Movement Analysis and Decomposition with the Continuous Wavelet Transform. In *8th International Conference on Movement and Computing (MOCO'22), June 22–24, 2022, Chicago, IL, USA*. ACM, New York, NY, USA, 24 pages. https://doi.org/10.1145/3537972.3537998

1 INTRODUCTION

The capture and description of human movements has an intertwined history between Sciences and Arts. The early endeavours of pioneers such as Étienne-Jules Marey, physiologist and physician, and Eadweard Muybridge, photographer, is exemplary of the different disciplines that were initially involved for studying human movements. Later, the possibility of quantifying human movement parameters opened the development of mathematical and computational modelling.

Movement analysis can be performed at different levels [6], from low-level features related to kinematics to high-level movement and gesture representations linked to semantic, affective or expressive qualities [5]. High-level qualitative representations have also been investigated in various fields, and in particular in non-verbal communication [22] and artistic practices such as dance and music. Nevertheless, important work remains to be performed to establish *quantitative* mid- and high-level movement representations. For example, while dance and somatic practitioners have developed powerful concepts to analyze how movements are performed, generally denoted as *movement qualities* [3, 35], it remain difficult to translate such concepts in general computational models [10].

In this paper, we investigate the use of time-frequency representations for computational movement analysis from wearable sensor data. Considering that movement qualities are often related to movement dynamics and rhythmic patterns, we hypothesize that such representations would be helpful for deriving features in a variety of tasks including

Manuscript submitted to ACM

the recognition and clustering of gestures or movement qualities. While the windowed Fourier transform has been applied to movement analysis [23, 32] and human activity recognition [13], we argue for the use of the Continuous Wavelet Transform (CWT), for it provides a more accurate time-frequency representation with optimal localization both in time and frequency. This means that the frequency resolution is higher in low frequencies, and that temporal resolution is higher in high frequencies. Despite its potential the CWT has been marginally applied to movement analysis, gesture recognition, and unsupervised learning of movement qualities [7, 36].

Our contributions are twofold. First, we propose a method for the analysis of human movement based on the CWT, illustrated with examples. We contribute an online approximation of the CWT for interactive applications, with various optimization schemes for performance. Second, we present applications of the method to the analysis of movement qualities on a dataset of gestural imitations of sounds. We propose three higher-level descriptions of the scalogram, based on simple statistics, target tracking and non-negative matrix factorization respectively. We report results for these methods on both clustering and recognition tasks, and we discuss how these methods help us analyse particular movement qualities and their relation to sounds in gestural imitations.

The remainder of this article is structured as follows. We review related work in Section 2. Then, we motivate and describe in Section 3 how the CWT can be used for movement analysis. In Section 4 we propose an online approximation for interactive applications. The method is then applied to movement analysis in Section 5, with both clustering and recognition tasks. The potential of the method for movement analysis is finally discussed in Section 6.

2 RELATED WORK

Computational movement analysis is a central research focus of the MOCO community. In this section, we review related research on movement representations, covering feature extraction frameworks and methods as well as time-frequency representations for movement analysis.

2.1 Computational Movement Analysis: from Trajectories to Qualities

Developing operational representations of movements is key to success in movement analysis and interaction design. Such representations build upon *features* that characterize some of the aspects of movement that researchers, engineers or designers aim to capture. These movement characteristics can cover low-level parameters such as trajectories, or higher-level aspects that relate to movement's expressive, affective or emotional qualities. Camurri et al. [6] proposed a multi-layered computational framework for movement analysis with a focus on extracting movement qualities. Their framework includes several successive layers of analysis, from signal-level to gesture-level and semantics-level descriptors.

Low-level movement representations build upon the raw data originating from either motion capture systems or wearable sensors. Common preprocessing operations include resampling, scaling, filtering, quantization, coordinate change, and numeric derivatives. In this paper, we focus on the use of Inertial Measurement Units (IMUs). For a single IMU, there are typically 6 or 9 dimensions from a 3D accelerometer, a 3D gyroscope and optionally a 3D magnetometer. Using several IMUs on the body scales the number of inputs. Cameras provide an even larger dimensionality considering pixels as raw data. Many mid-level features have been proposed within the field of movement computing to represent positional motion capture data, including curvature [12], Kinetic Energy [12], the contraction index [35], the silhouette [35], or the quantity of motion [35]. Other approaches focus on a geometrical description of postures, with features such as bounding space [16], center of mass [14], directness [35], or Müller's Full Body Relational Features [24].

Such mid-level features can be used to derive representations capturing higher-level concepts such as emotion and movement qualities [5, 9, 10]. Movement qualities are widely used in dance practice and choreography, and consider "how" a movement is performed (with what intention, expression or emotion). Laban Movement Analysis (LMA), developed by Rudolf Laban in the early twentieth century and pursued by theorists such as Bartenieff, is the most widespread theoretical model of movement that has been used for computational modeling. In particular, its "Effort" category is used to describe movement qualities through four dimensions of Space, Weight, Time and Flow [10]. In early research, affinities between such efforts and spatial representations have been proposed [35]. A number of methods based on machine learning have also been developed to recognize such movement qualities from sensor data [27, 33]. A second approach is to work with movement qualities defined by choreographers themselves, and modeled through physical modeling techniques, as proposed by Alaoui et al. [2].

2.2 Time-Frequency Representations for Movement Analysis

We are interested in exploring movement qualities through the perspective of movement dynamics. Gestures can be performed with different expression by altering their temporal dynamics to convey intent [9, 10]. We now review computational methods capturing temporal and/or spectral characteristics of human movement.

2.2.1 Fourier Transforms. The Fourier transform has been successfully used in the analysis of human movement and applied to gesture analysis and recognition. Usually, the Windowed Fourier Transform (WFT) is applied on segments of movement data, and computed using the Fast Fourier Transform (FFT). Shiqi Yu et al. [32] used the normalized power FFT on a set of points of the silhouette contour. The authors argue that the use of normalization allows for scale invariance, which improves the recognition of gait patterns. Makihara et al. [20] combined auto-correlation with spectral analysis of gait movements. They propose to evaluate the gait period from auto-correlation, and subsequently apply a WFT on each gait cycle to extract spectral features. The Fourier transform has also been applied to measure movement smoothness. Melendez-Calderon et al. [23] report a systematic evaluation of smoothness measures from IMUs relying on the spectral arc length. They show that thee method is valid when applied on velocity data or on rotational velocity data. Finally, the Fourier transform has been used in conjunction with convolutional neural networks for sensor-based activity recognition [13].

2.2.2 Wavelet Transforms. The Continuous Wavelet Transform (CWT) has been marginally applied to the analysis of human movement. Côté-Allard et al. [7] used the CWT together with Convolutional Neural Networks (CNNs) and transfer learning for gesture recognition from electromyography (EMG). However, EMG data has different statistical properties compared to IMUs. Recently, Nedorubova et al. [26] proposed to use the CWT for activity recognition. The CWT is used to extract time-frequency representations of activities captured with IMUs. The resulting images are fed to a CNN for the recognition of daily activities. The CWT has been previously applied to multimodal discourse analysis [36]. Xiong and Quek proposed to use the CWT with a Morlet Basis to compute a time-frequency representation of speech-related gestures. From the extraction of frequency ridges, the authors propose a method for analyzing the relationship between oscillatory gestures and speech content.

3 MOVEMENT ANALYSIS WITH THE CONTINUOUS WAVELET TRANSFORM

The Continuous Wavelet Transform (CWT) is a time-frequency analysis method allowing for optimal time-frequency localization [1, 21]. In this section, we consider how the wavelet transform can be used for characterizing dynamic

behaviors in hand gestures. For more extensive tutorials on wavelet analysis, we refer the reader to the dedicated literature [1, 21, 34].

3.1 Motivation

In Fourier analysis, the spectrum of a signal is estimated by convoluting the signal with a set of harmonic plane waves. The Windowed Fourier Transform (WFT) — or Short-term Fourier Transform (STFT) — computes a time-frequency representation of signals by performing a Discrete Fourier Transform on a sliding window along the signal. As all digital signals are finite, in practice the plane waves — or equivalently, the signals, — are multiplied by a window function to avoid artifacts due to border effects. The WFT therefore assumes a fixed window size, which determines the bandwidth of each frequency band in the time-frequency representation. One of the main limitation of the WFT is the inaccuracy resulting from the imposition of a scale or 'response interval' into the analysis [34]. Indeed, the WFT aliases all frequency components that do not fall within the frequency range of the window.

On the contrary, instead of assuming a fixed window size, the wavelet transform both translates and dilates a *wavelet* function with short-term influence. The dilation of the wavelet implies that the analysis window is expanded as the carrier frequency of the wavelet decreases. As a result of the Heisenberg's inequality, time-frequency resolution varies: the temporal window is short in high frequencies while the bandwidth is narrower in low frequencies.

A rapid analysis of typical setups for capturing and representing movements further motivates the use of multi-resolution analysis. Typically, motion capture systems such as inertial sensors or low-cost motion capture devices have framerates of about 50 to 500 Hz. Human movements are typically in the range of a few hertz, from $0.2 - 0.5$ Hz to $10 - 15$ Hz for smooth and periodic movements, and up to 50 Hz for impacts. If one considers that the frequency resolution necessary to derive an accurate analysis around 1 Hz should be at most of order 0.1 Hz, then the minimal window size required with the WFT is 10 s. In this case, any transient high-frequency phenomenon (of duration typically inferior to a second) will be blurred by the size of the analysis window. On the contrary, imposing a window size of 1 s to guarantee an acceptable time localization restricts the resolution in the frequency domain to a bandwidth of 1 Hz, which is insufficient for capturing low-frequency phenomena. The multiresolution analysis solution provided by the wavelet transform allows us to derive an arbitrary high localization both in time and frequency [1].

3.2 Formulation of the Continuous Wavelet Transform

The Continuous Wavelet Transform (CWT) of a discrete sequence $\boldsymbol{x}_{1:N} = \{x_1 \cdots x_N\}$ sampled at period δt for a *wavelet function* Ψ_0 is defined as the convolution of the sequence with a scaled and dilated version of the base wavelet [34]:

$$W_n(s) = \sum_{n'=0}^{N} x_{n'} \Psi^* \left[\frac{(n'-n)\delta t}{s} \right] \quad , \quad \forall n = 1 \cdots N \tag{1}$$

where Ψ^* is the complex conjugate of the normalized wavelet Ψ:

$$\Psi \left(\frac{(n'-n)\delta t}{s} \right) = \left(\frac{\delta_t}{s} \right)^{1/2} \Psi_0 \left(\frac{(n'-n)\delta t}{s} \right) \tag{2}$$

and s is the scale parameter. The time-frequency representation can be constructed by *translating* the wavelet along the time axis (varying n) and *dilating* the wavelet with the scale parameter s. By analogy with the term 'spectrogram' for the WFT, the 'scalogram' can be computed by taking the power representation of spectral information in the scale domain $|W_n(s)|^2$.

For most wavelets, there exist an analytical formulation of their Fourier Transform, which reduces the computational cost to a single inverse FFT per frequency band that simultaneously estimates the scalogram at all time steps.

3.2.1 Wavelet Functions. While the WFT imposes windowed plane waves as set of base functions, wavelet analysis offers flexibility in the choice of the base functions for analysis. To be admissible as a wavelet, a function must meet three conditions: the must have zero-mean, finite energy, and their Fourier transform must be real and vanish in negative frequencies [1]. Several factors are to be considered in the choice of a wavelet basis, including orthogonality, real vs complex wavelets, width and shape [34].

In this research, we mostly experimented with the Complex Morlet wavelet — sometimes called Gabor Wavelet, — which has the property of optimal localization in time and frequency. It is defined as a plane wave modulated by a Gaussian window[1]:

$$\Psi_0(\eta) = \pi^{-1/4} \left(e^{i\omega_0\eta} - e^{-\omega_0^2/2} \right) e^{-\eta^2/2} \tag{3}$$

where ω_0 is the carrier frequency of the Wavelet. The spectrum of the Morlet wavelet is a unit Gaussian centered around its carrier frequency.

3.2.2 Choice of scales. In comparison with the WFT, the CWT offer a great flexibility in the choice of the analysis domain. In particular, while the WFT restricts the analysis domain to the set of harmonic bands, one can choose a subset of arbitrary scales for wavelet analysis. We follow a convention to distribute the scales as fractional powers of two:

$$s_j = s_0 2^{j/b} \quad , \quad j = J_{min}, J_{min}+1, \cdots, J_{max} \tag{4}$$

where b is the number of bands per octave and J_{min} and J_{max} can be specified from a target frequency range. The smallest resolvable scale s_0 can be estimated from the equivalent Fourier frequency according to the Nyquist frequency $1/2\delta t$. This representation is convenient for selecting the analysis domain, whose specification is therefore reduced to a frequency range and a number of bands per octave.

3.3 Example

In this section, we report an illustrative example comparing the proposed method with the Windowed Fourier Transform. We consider movements captured using inertial sensors. Our approach to multidimensional analysis using the wavelet transform is based on a late fusion of the scalograms. When analyzing dynamic movements with an IMU sensor fixed on the wrist, our goal is to derive a representation of the spectral behavior that is invariant to the movement's direction and amplitude. In this case, we compute the scalogram for each axis of the accelerometer independently, and we represent gestures by the sum of the scalograms on each axis. This approach provides a representation that is independent from the movement direction, and that can take into account different sensors (accelerometers and gyroscopes). However, it is possible to consider scalograms independently on each axis in order to preserve the spatial information.

Figure 1 shows a comparison of the CWT with the WFT on human movement captured with IMUs. We observe that oscillatory behaviors appear with a higher contrast in the scalogram (CWT) than in the spectrogram (WFT). Moreover, the low frequency component is also better defined in the scalogram.

[1]We report here the 'complete' for of the Morlet Wavelet. As discussed by Addison et al. [1], considering that the correction term is negligible for $\omega_0 > 5.$, most articles in the literature use a truncated form for the Morlet wavelet: $\Psi_0(\eta) = \pi^{-1/4} e^{i\omega_0\eta} e^{-\eta^2/2}$. However, in some cases it can be interesting to use smaller values of ω_0 in order to get a high temporal localization

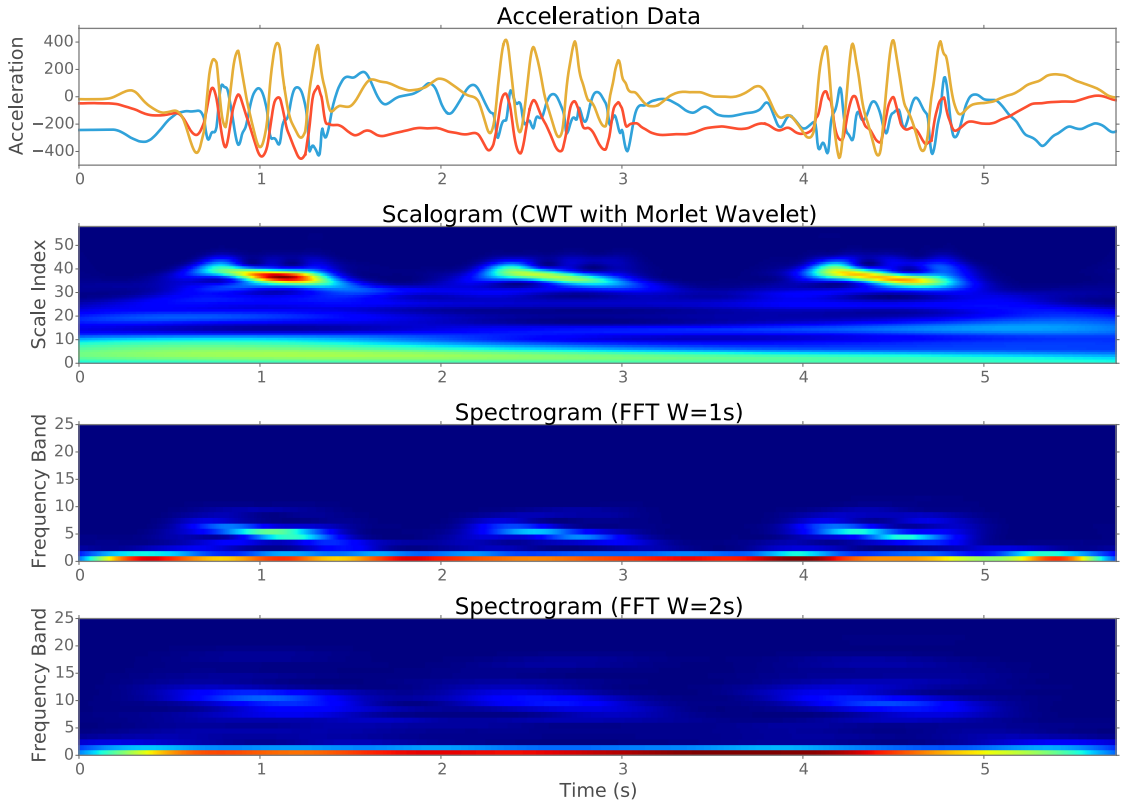

Fig. 1. Time-frequency power spectrum of an acceleration signal (i.e. raw signal from a 3D accelerometer in arbitrary units) using the CWT and WFT. We used a window size of 1s and 2s for the WFT, and 8 bands per octave on frequency range $[0.2; 50]$ Hz for the CWT.

4 IMPLEMENTATION

The computation of CWT can be performed either offline or online. Both are described below, with more attention to the online implementation which might necessitate different approximations. Before describing these two cases, we describe the important notion of *cone of influence*.

4.1 Cone of Influence

As for the WFT, the CWT is subject to edge effects resulting from the use of finite signals. While this problem can be partially resolved by padding with the edge values before performing the analysis, this process still introduces discontinuities at endpoints. The *cone of influence* is defined as the region where the edge effects become important, and can be defined in terms of *e-folding* time. According to Torrence and Compo [34] "This *e-folding* time is chosen so that the wavelet power for a discontinuity at the edge drops by a factor e^{-2} and ensures that the edge effects are negligible beyond this point".

The *e-folding* time for scale s with the Morlet wavelet is $\sqrt{2}s$, and the relationship between the scale and the equivalent Fourier frequency can be computed as

$$f = \frac{\omega_0 + \sqrt{2 + \omega_0^2}}{4\pi s} \tag{5}$$

In the remainder of this study, we consider a sufficient padding of the signals so that the scalogram can be estimated without edge effects at all scales. We also use the *e-folding* time as criterion for the online approximation of the CWT (section 4.3).

4.2 Offline Computation

For offline estimation, the CWT can be efficiently computed using the Fast Fourier Transform to evaluate the convolution as a product in the spectral domain:

$$\begin{aligned} &W_n(s) = \textstyle\sum_{k=0}^{N-1} \hat{x}_k \hat{\Psi}^*(s\omega_k) e^{i\omega_k n \delta t} \\ &\text{with angular frequencies} \quad \omega_k = \begin{cases} \frac{2\pi k}{N\delta t} & \text{if} \quad k \leq N/2 \\ -\frac{2\pi k}{N\delta t} & \text{if} \quad k > N/2 \end{cases} \end{aligned} \tag{6}$$

4.3 An Online Approximation of the CWT

The CWT is typically computed offline, when the entire signal is available. In order to use in interactive applications, we propose an online approximation of the CWT. Our implementation is based on a finite-length approximation of the wavelet depending on the cone of influence in each frequency band. We propose to use a computation window as small as possible at each scale to get a good approximation of the scalogram with a minimal delay in each frequency band. Additionally, we propose two optimization schemes based on a multi-rate representation of the wavelet and of the incoming signal.

4.3.1 Formulation. We consider a finite-length windowing method where the window size is specified for each frequency band depending on the wavelet's energy decrease. The CWT is implemented as a filter bank with minimal delay per frequency band, and the computations are done in the temporal domain. At each new frame of the signal, we estimate only the central value of the scalogram on a sliding window with minimal size with regards to the e-folding time. This process is illustrated in Figure 2.

For each scale s, the value of the scalogram with delay $N_s/2$ is estimated from a new observation value as

$$W_s[t - N_s/2] = \sum_{k=0}^{N_s} x[t - N_s + k] \cdot \Psi^* \left[\frac{(k - N_s/2)\delta t}{s} \right] \tag{7}$$

where the window size N_s can be estimated from the wavelet's e-folding time τ_s as $N_s = \lambda \tau_s / \delta t$ with λ a constant determined experimentally we call the *windowing factor*. For a Morlet wavelet with carrier frequency ω_0, the window size can be estimated at a given scale from the equivalent Fourier frequency f as

$$N_s = \lambda\sqrt{2} \cdot \frac{\omega_0 + \sqrt{2 + \omega_0^2}}{4\pi f \delta t} \tag{8}$$

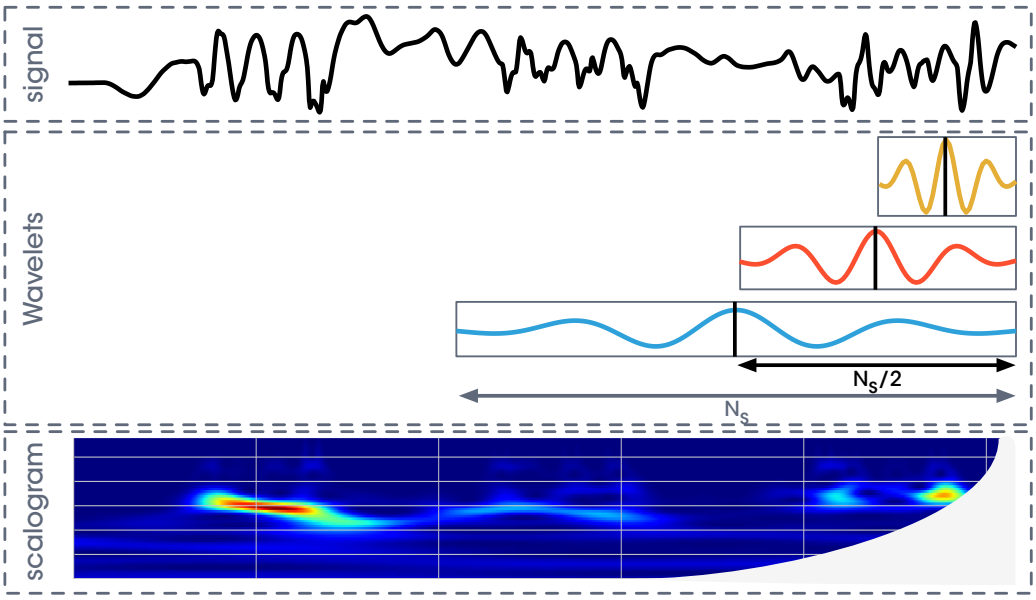

Fig. 2. Illustration of the online implementation of the Continuous Wavelet Transform.

The complexity per frequency band is therefore N_s multiplications and $N_s - 1$ additions. For the Morlet wavelet with scales distributed in powers of two, the total complexity is exponential in the number of bands and can be estimated as

$$C = 2 \sum_{j=J_{min}}^{J_{max}} \lambda\sqrt{2} s_0 2^{j/b} - 1 = 2\sqrt{2}\lambda s_0 \left(\frac{(2^{1/b})^{J_{max}} - (2^{1/b})^{J_{min}}}{2^{1/b} - 1} \right) \tag{9}$$

where b is the number of bands per octave and $s_0 = \frac{\omega_0 + \sqrt{2+\omega_0^2}}{4\pi \cdot 2\delta t}$ is the highest resolvable scale.

4.3.2 Optimization by Multi-rate Approximation. Computing the online CWT can be intensive at high framerates as the number of bands per octave increases, and as the analysis requires low-frequency components involving large window sizes. To alleviate this issue, we propose two optimization schemes of the online transform based on a multi-rate representation of the wavelets and of the signal.

Standard Optimization: The number of computations per frequency band can be reduced by considering that low-frequency components can be approximated by a downsampled version of the wavelet. We propose to downsample the wavelets by an integer factor depending on the ratio of their equivalent Fourier frequency with the Nyquist frequency. As a result, we guarantee that the wavelet's samplerate is sufficient to avoid aliasing the corresponding frequencies, while reducing the number of computation by an integer factor. Note that in this case, the incoming signal is not decimated and we still evaluate the CWT at each frame for all frequency bands. To avoid aliasing in low frequencies, the signal is passed to a bank of low-pass filters for each decimation factor.

Aggressive Optimization Further optimization can be achieved by decimating not only the wavelet, but also the incoming signal. In this case, instead of evaluating the CWT at each frame for all frequency bands, we apply a

Table 1. Normalized Mean Squared Error of the online approximation of the scalogram compared with the standard estimate. Results are averaged over all imitations from all participants on the *Abstract* sound family of the ANON dataset (see Section 5. The results are displayed according to the size of the approximation window relative to the wavelet's e-folding time in each frequency band.

Win. Factor	**NMSE (%)**	**Comp. Time (10^{-3} ms/loop)**
1	54.676228	1.867
2	21.404965	3.687
3	4.632857	5.502
4	1.293430	7.362
6	0.008430	11.035

similar decimation of the signal. The wavelet power spectrum is therefore evaluated at a lower framerate for low-frequency components, which provides a sparser representation of the scalogram.

As the standard optimization scheme keeps a frame-based computation of the scalogram, the estimation of the wavelet power spectrum remains smooth, which is particularly suited to interactive applications. On the contrary, aggressive optimization involves a severe downsampling of the incoming signal and the wavelet spectrum for low-frequency components is evaluated at larger time steps. This latter optimization scheme can be interesting when a sparse representation of the scalogram is desired.

In our experiments, *standard* and *aggressive* optimization techniques lead to a gain of a factor 20 and 80 in computation time, respectively. Note that the gain with standard optimization is made at the cost of an increased memory footprint, as it is required to store several versions of the incoming signal with different lowpass filtering. In both cases, the major drawback of the optimization methods is the introduction of an additional delay due to the low-pass filtering of the signal.

Figure 3 shows an example of the online approximation of the continuous wavelet transform computed over an acceleration signal. The same plot with the realigned scalograms is showed in Figure 4, where the delay introduced by the online approximation is compensated in each frequency band, for comparison with the online estimate. While the online approximation without optimization introduces a delay, that increases as the scale increases, it can be seen that the distortion of the scalogram is relatively small compared with the offline estimate.

We evaluated the quality of the approximation of the scalogram using the online implementation of the CWT. Table 1 reports the Normalized Mean Squared Error (NMSE) or the online approximation of the scalogram as a function of the windowing factor. Each error is estimated with respect to the offline estimation of the scalogram, and is averaged over the gestures from all participants from the dataset presented in Section 5. For comparison with the true estimate, the estimation delay of the power spectrum was compensated in each frequency band.

A relative window size of 3 — with respect to the e-folding of the wavelet in each frequency band, — provides a NMSE inferior to 5%. This approximation is sufficient in most interactive applications, as the approximation of the scalogram is not distorted.

4.3.3 Software. We implemented the online CWT as a freely available external for Cycling'74 Max within the MuBu package[2] [31]. It is implemented as PiPo module that can used for real-time processing or off-line processing of data buffers. An experimental JavaScript implementation is also available within the CODA live-coding environment[3] [11]. Two modes are possible for the real-time analysis of movement data within Max.

[2]https://forum.ircam.fr/projects/detail/mubu/
[3]https://github.com/julesFrancoise/coda

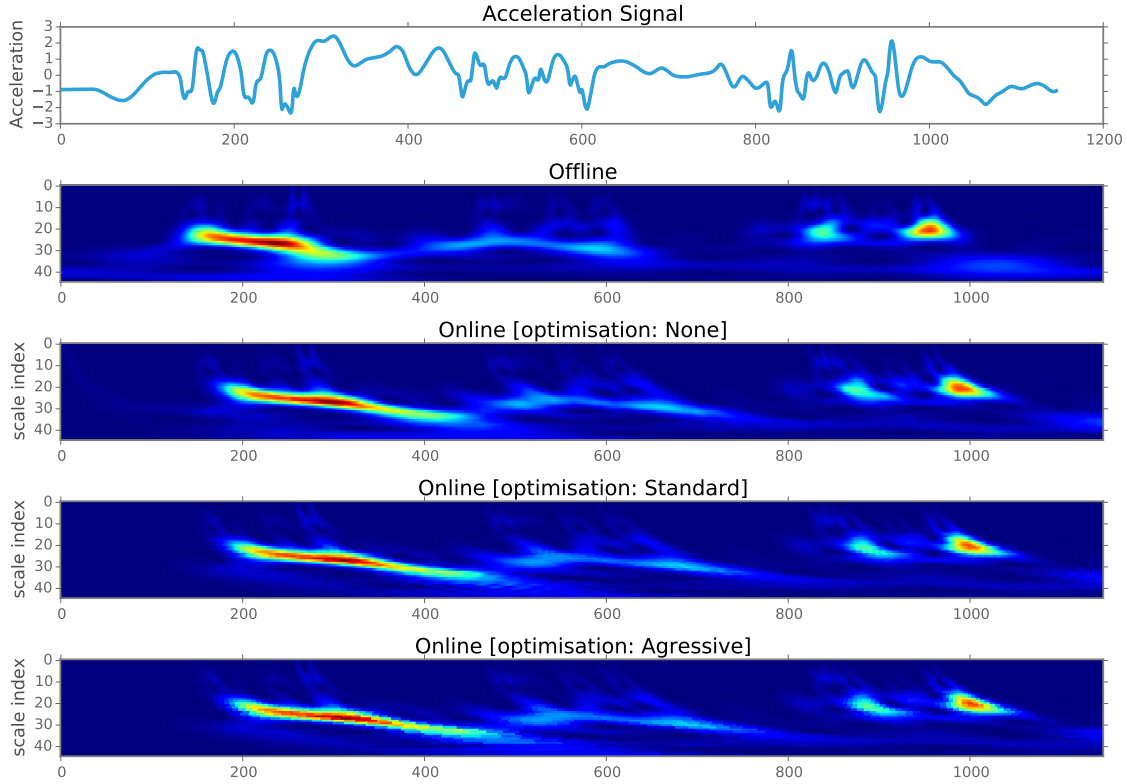

Fig. 3. Example of online approximation of the continuous wavelet transform computed over an acceleration signal (i.e. from an accelerometer) from a dataset of gestural imitations of sounds. The approximation was computed with carrier frequency $\omega_0 = 5$ and *windowing factor* $\lambda = 3$. The x-axis unit the sample number, using sampling rate of 200 Hz. The y-axis unit of the acceleration signal is in g

In the first mode, the values of the wavelet power spectrum are outputted with a minimal delay in each frequency band. In this case, each band of the filterbank is evaluated with a different delay. While this approach does not guarantee a correct alignment of the different frequency bands, it provides a high reactivity in high frequencies, which can be interesting for interactive applications where impulsive gestures should be identified with low-latency while low-frequency components are longer to establish. In the second mode, the wavelet spectrum is outputted with the same delay in all frequency bands, guaranteeing a correct alignment of the various components. However, this implies that the delay must be aligned on the largest delay corresponding to the lowest frequency component.

5 APPLICATION TO MOVEMENT QUALITIES MODELING

In this section, we present the application of the method to movement analysis. We used the CWT for both recognizing or clustering movement qualities on a dataset of gestural imitations of sounds.

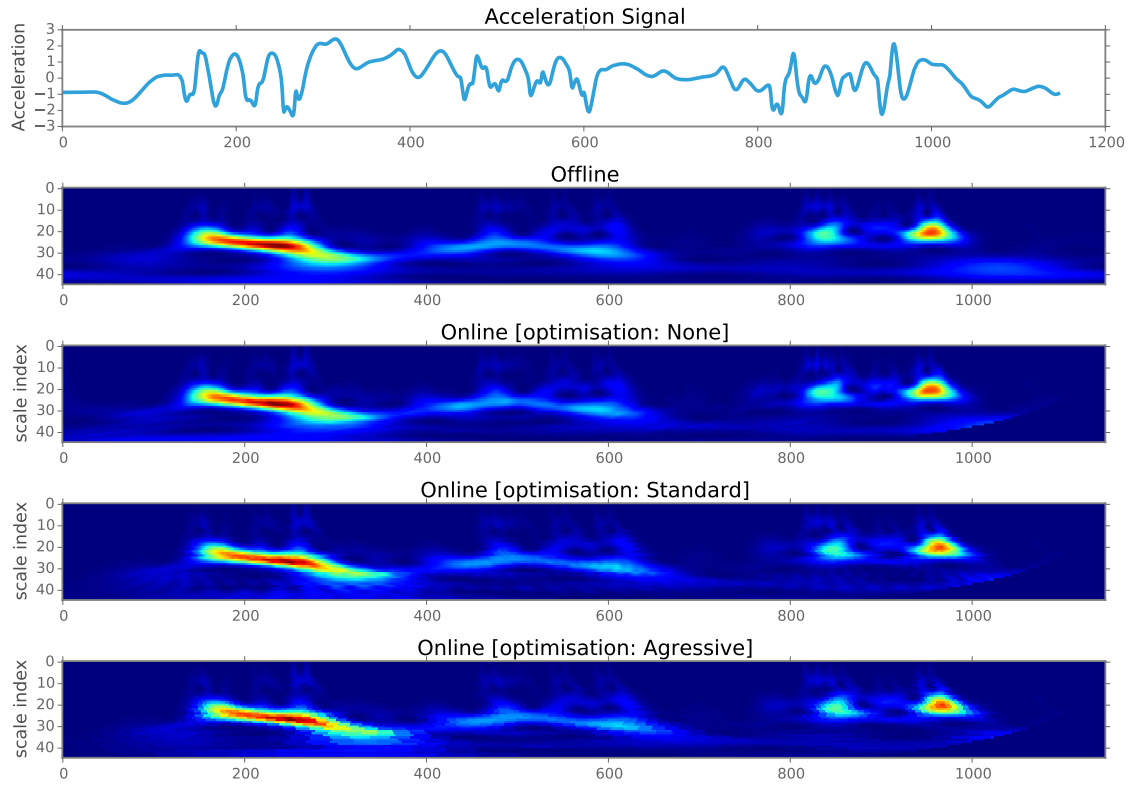

Fig. 4. Example of online approximation of the continuous wavelet transform computed over an acceleration signal from from a dataset of gestural imitations of sounds. The delays in each frequency band are compensated for comparison with the offline estimate. The approximation was computed with carrier frequency $\omega_0 = 5$ and *windowing factor* $\lambda = 3$. The x-axis unit the sample number, using sampling rate of 200 Hz. The y-axis unit of the acceleration signal is in g

5.1 Data and Method

5.1.1 The Imitation Dataset. The SkAT-VG Imitation Dataset[4] is a publicly available dataset of vocal and gestural imitation of sounds [18]. The dataset contains audio, video, and motion capture recordings of participants performing vocal imitations and gestures of recorded sounds belonging to three broad families: *Abstract* sounds, *Interaction* sounds and *Machine* sounds. The protocol used to constitute the dataset is summarized hereafter, and it is fully described in [18]. The experiment was divided in two blocks where participants were instructed to jointly perform a vocal and gestural imitation of the referent sound. We only used the second block where participants' hand movements are recorded, using two inertial measurements units attached to wrists, measuring 3D accelerometer data at a sampling rate of 200 Hz.

The choice of the wavelet movement analysis method on the SkAT-VG imitation dataset was driven by the insights gathered through qualitative studies on participants' strategies [18]. In particular, considering the case of abstract sound where no identifiable pitch could be perceived, participants did not favor any specific direction in the movement. The

[4]See https://www.ircam.fr/projects/blog/multimodal-database-of-vocal-and-gestural-imitations-elicited-by-sounds/. This dataset was built during the Skat-VG project (https://cordis.europa.eu/project/rcn/110562/factsheet/en)

use of specific rhythmic behaviors for different sounds proved to be the most salient feature in the sense that it is shared across most participants.

5.1.2 Manual Annotation of the Dataset. While the dataset was initally annotated with sound categories, observations of the participants' gestures highlighted the importance of temporal and rhythmic patterns. Therefore, we reframed the analysis problem as the extraction of "movement qualities", independent from spatial directions but characterized by their frequency content, instead of the recognition of sound categories from gestures. Importantly, we define these "movement qualities" for our specific evaluation, and their validity is restricted to the specific dataset we use.

First, gestures were segmented into three phases of *Preparation*, *Stroke* and *Recovery* — inspired by similar representations proposed in the literature [15]:

Preparation refers to the phase that leads from the relaxed position to the stroke.
Stroke is where the actual expression of the gesture is accomplished.
Recovery covers the phase from the stroke to the final relaxed state.

Then, we defined six categories to characterize the *Stroke*:

Steady gestures are postures that remain constant over time. They include completely still postures and gestures that evolve very slowly.
Smooth movements are fluid and include gradual changes in posture.
Dynamic gestures involve abrupt, energetic and rapid actions.
Impulse describe single and sudden gestures.
Periodic gestures refer to movements exhibiting periodicity
Shaky is a specific class for the dataset that involves the hand shaking.

The dataset was manually annotated by two researchers, using a dedicated interface created with Max/MSP, where video recordings and signals were displayed to facilitate annotation. We evaluated the inter-rater agreement between annotators using Cohen's kappa coefficient. We obtained a value of 0.67 for this coefficient, that indicates a substantial conformity. To understand where disagreement occur, we analyzed a confusion matrix between the two annotators. It revealed that most disagreements occur between the dynamic and smooth categories. This is due to the fact that the threshold to consider a gesture as fluid or energetic depends on the annotator's judgment.

5.2 Statistical Gesture Descriptors for Recognition

In this section, we describe two approaches to the representation of gestures based on gesture-level descriptors derived from wavelet analysis. The first approach is based on a description of the scalogram by its normalized *moments* (centroid, variance, skewness, kurtosis) in the wavelet spectrum domain and in the temporal domain. The second approach draws upon the multi-target tracking of the frequency ridges.

5.2.1 Spectral and Temporal Moments. We propose to describe the scalogram at a high level from the distribution of the wavelet power spectrum and of the energy envelope. The global wavelet power spectrum for an entire gesture can be obtained through the time-averaged scalogram (Figure 5, right):

$$\bar{P}(s) = \sum_{n=0}^{N-1} |W_n(s)|^2 \quad , \quad \forall s \in \{s_{min}, \cdots, s_{max}\} \tag{10}$$

Similarly, the energy envelope of the gesture can be obtained from the scale-averaged scalogram (Figure 5, bottom):

$$\bar{E}_n = \sum_{s=s_{min}}^{s_{max}} |W_n(s)|^2 \quad , \quad \forall n \in \{0, \cdots, N-1\} \tag{11}$$

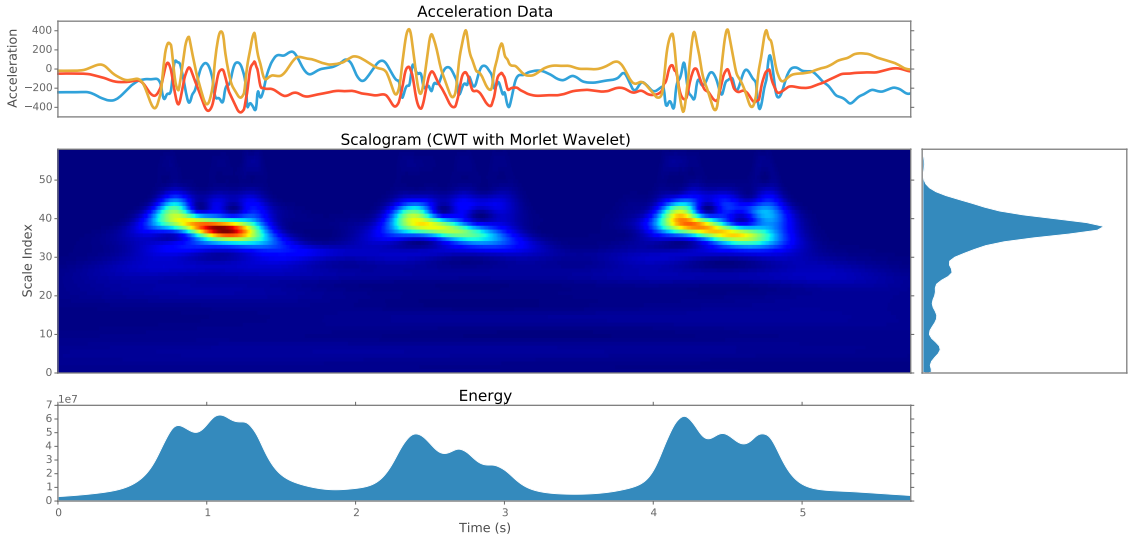

Fig. 5. Description of a gesture through its scalogram's energy envelope and power spectrum. Global descriptors can be extracted using the moments of these distributions (centroid, variance, skewness, kurtosis).

We further abstract the temporal and spectral information by taking the normalized moments of each distribution, raising 8 parameters for the description of a single gesture: *Spectral Centroid, Spectral Variance, Spectral Skewness, Spectral Kurtosis, Temporal Centroid, Temporal Variance, Temporal Skewness, Temporal Kurtosis*. While this description remains simple, it is invariant to the scale of the movement data. This invariance is essential to the consistency of the analysis, considering that participants' energy in the gesture performance is extremely variable in the database.

5.2.2 Average Ridge Descriptors. A similar description can be derived from features obtained through the tracking of frequency ridges. The wavelet spectrum representation of movement data provides rich information about the frequential content of dynamic gestures. In particular, periodic and impulsive gestures result in clear ridges in the scalogram centered around the fundamental frequency of the movement. We developed a method for tracking frequency ridges in the scalogram, drawing upon methods from multi-target tracking in computer vision and radar applications. Our method is described in detail in Appendix A.

As result of the tracking process, several target ridges are identified and their *amplitude* $\bar{A}$, *frequency* $\bar{F}$, and *variance* $\bar{W}$ are continuously estimated along the gesture. This allows to separate more clearly the contribution of concurrent frequential modes in the gesture. As before, we can consider time-averaged ridge descriptors as the basis for high-level

description of gestures. In particular, for a set of K ridges, we can compute $3K$ parameters per ridge as:

$$\begin{aligned} \bar{F}^{(k)} &= \sum_{n=0}^{N_1} F_n^{(k)} \quad \forall k \in \{1, \cdots, K\} \\ \bar{A}^{(k)} &= \sum_{n=0}^{N_1} A_n^{(k)} \quad \forall k \in \{1, \cdots, K\} \\ \bar{W}^{(k)} &= \sum_{n=0}^{N_1} W_n^{(k)} \quad \forall k \in \{1, \cdots, K\} \end{aligned} \tag{12}$$

The description of the temporal envelope can be obtained independently for each ridge from the estimated ridge amplitude $A_{1:N-1}^{(k)}$.

5.2.3 Recognition of Movement Qualities. We evaluated whether our method could be used as a mean to automatically recognize the movement qualities defined manually. For this we followed a standard 5-fold cross-validation method where the 35 participants used in the experiment are randomly assigned to 5 groups. This guarantees that the gestures of a same participant cannot be found both in the training and test sets. The classification was performed using Gaussian Mixture Models (GMMs) [25]. We used 5 full-covariance Gaussian components per GMM, with variance regularization.

As baseline, we computed the accuracy score between the annotation of the two human annotators. Once again, the inter-annotator accuracy of 0.67 emphasizes a relative agreement of the annotators, with disagreements regarding the *smooth* and *dynamic* categories.

We compare several gesture-level features, derived either from the spectral and temporal moments of the scalogram, or from the ridge tracking algorithm. For the moments representation, we use the first two moments of the scalogram averaged either in time or scale, raising four descriptors per gesture: the temporal and wavelet spectral centroid and variance. Gesture-level descriptors derived from the ridge tracking algorithm have a similar form, with simplified temporal modeling. The best scores were obtained by using only the duration information of each gesture with the average frequency of each ridge — amplitude and spread did not significantly improve the classification accuracy. For consistency, in multi-target tracking, the ridges are ordered in descending intensity.

Table 2 reports the classification accuracy with regards to each annotator's reference labels. With a 2s window size, the WFT gives significantly lower accuracy scores than the description based on the continuous wavelet transform. The results obtained using continuous ridge tracking are superior to the accuracy obtained using the moment-based description. This might indicate that ridge tracking is more robust to noise than the moments. However, no significant difference in accuracy is found between 1, 2 and 3 ridges for the recognition of gesture primitives on the Abstract categories, which might indicate that all gestures are sufficiently represented using a single frequential mode.

Descriptor	Annotator 1		Annotator 2	
	CWT	WFT	CWT	WFT
moments	0,63	0,58	0,57	0,54
1 ridge	**0,68**	0,42	**0,62**	0,43
2 ridges	0,67	0,25	0,59	0,24
3 ridges	0,66		0,59	

Table 2. Classification accuracy of movement qualities from gesture-level descriptors. The classification was performed using GMMs with 5 full-covariance Gaussian components. Each accuracy score was computed on 5-fold cross-validation on the imitations of *Abstract* sounds from 35 participants, where participants from the test set are absent from the training set.

While the recognition might appear low, the confusion matrix provide us with a more general view of the results. Figure 6 depicts the confusion matrix associated with ridge-based classification. First, it shows that the some movement qualities are better recognised than others (impulse is recognized at approximately 90%), and that the relatively low

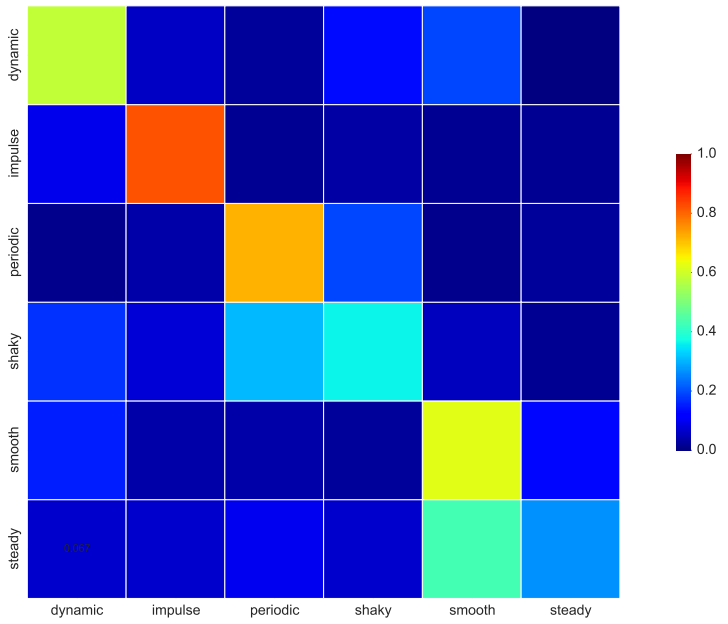

Fig. 6. Confusion Matrix of the recognition of gesture primitives. The classification was performed using GMMs with 5 full-covariance Gaussian components from the gesture-level descriptors derived from ridge tracking with 2 target frequencies.

recognition rate is due to specific confusions. Specifically, we found that most of the recognition errors occur between the *periodic* and *shaky* categories. This suggest that both categories could be merged, which is supported by further qualitative analysis: some gestures are visually associated to the *shaky* categories, due to apparent 'random' motion of the hand and finger movements, but the wrist movements, which are actually the ones measured, remain globally periodic. Merging the periodic and shaky categories led to an improvement of the accuracy using all descriptors, that reaches about 80% using ridge tracking descriptors. The other sources of inaccuracy arise from the definition of the boundary between smooth and dynamic gestures. Once again, smooth and dynamic categories are defined using a threshold that is subjective to the annotator, while their union forms a continuous space with varying frequency and intensity.

5.3 Representation Learning using Non-negative Matrix Factorization

The task of recognition of human-defined movement qualities is known to be difficult, as seen in the previous section. Another approach is to use data to derive "movement qualities" using a non-supervised task. We refer this as representation learning. Thus, we investigated whether scalograms can represent first basic gestures primitives, then leading to define movement qualities by using clustering techniques . As previously reported, one possibility is to simply apply statistics to the scalogram to extract a global representation. In order to better characterize specific patterns appearing in the scalogram, we introduce an unsupervised method based on Non-negative Matrix Factorization (NMF) [17]. NMF is similar to Principal Component Analysis (PCA), but working strictly with non-negative matrices. The idea is to find a limited set of components (i.e. primitives) from a dataset of scalograms stored in a non-negative matrix V (the scalogram is non-negative by definition). The components, represented by a non-negative matrix H, allow for expressing any

scalogram from the dataset (and by extension other scalogram) as a weighted sum of the components. The weights are also represented as a non-negative matrix. The mathematical description of the method is available in Appendix B.

The amplitude of each scalogram was normalized between 0 and 255 (treated as an image of float values, with a fixed size of 64 scale bands × 500 normalized time steps. Since both time is resampled to 500 points and amplitude is normalized to 255, both the average time and amplitude information are ignored (for this step, but these parameters will be reintroduced in the clustering operation, see next section) Then, the V matrix is obtained by concatenating scalogram, flattened to vectors of size 32000 (64×500). Finally, using equations 19 and 20, the first $k = 25$ components are computed. They are reported in Figure 7.

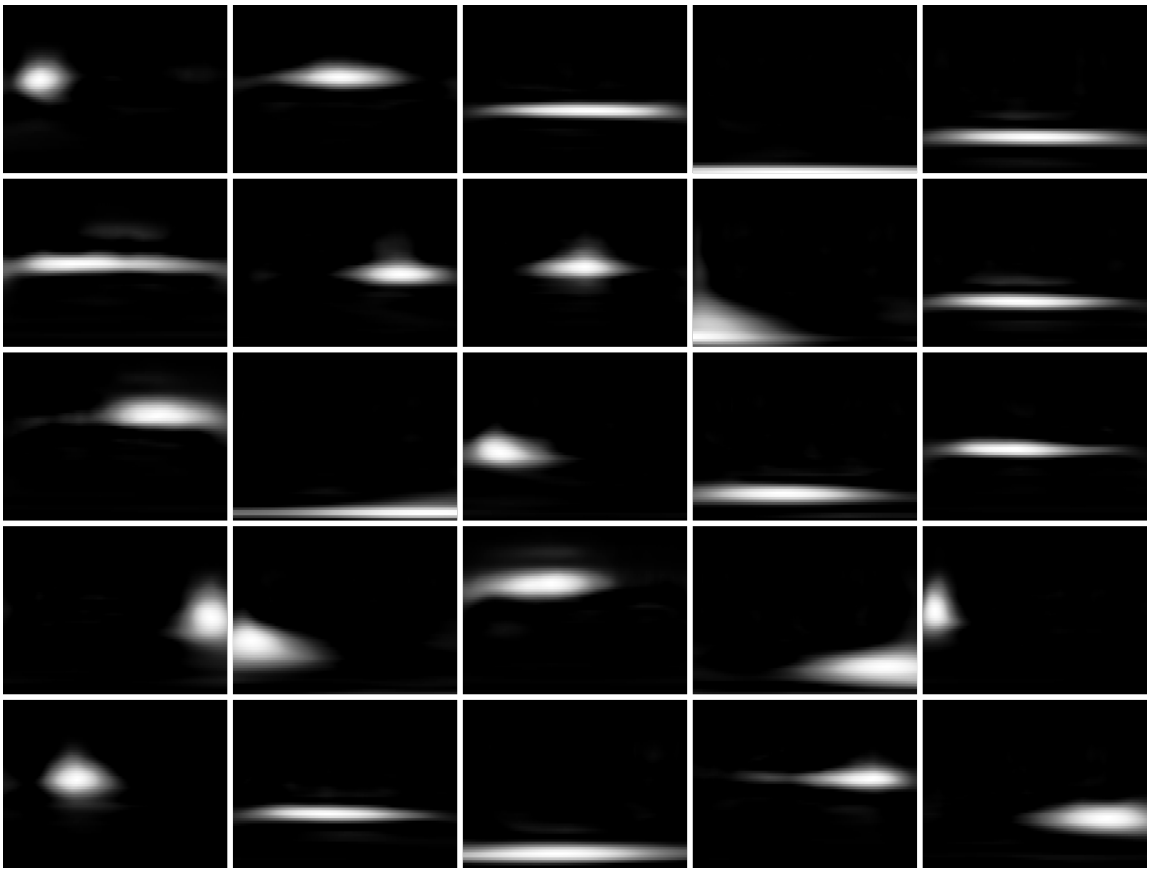

Fig. 7. The 25 basic components of the dataset. Each one represents a gesture primitive spatially localized (time and amplitude) in our gesture scalogram dataset. The components are represented as images of 64 scale bands × 500 normalized time steps.

5.3.1 Clustering. Each gesture is represented by a vector containing the weights of the primitives, along with the *average time* and *energy* (total energy of the original scalogram). In order to discover significant gestures primitives, the 27 dimensions[5] feature space previously described is clustered. In order to apply the clustering algorithm, we transform

[5]the 25 basic vectors of the NMF + average time and energy.

the initial descriptors distribution, which are close to an exponential distribution, to a distribution that is close to a normal distribution, by applying the function $ln(x + 1)$. Finally, the dataset is standardized by centering and scaling each feature independently (zero mean and unit variance distribution).

The clustering is performed using a K-Means algorithm on all gestures in the dataset. The cluster number was set to six to compare the automatic clustering to the six primitives defined manually. The six primitives are reported in Fig. 8 as reconstructed scalograms. It appears that most clusters occupy specific time-frequency regions of the scalograms. While clusters 1 and 3 (left column of Figure 8 occupy similar time-frequency regions, they are distinct in term of average time and energy.

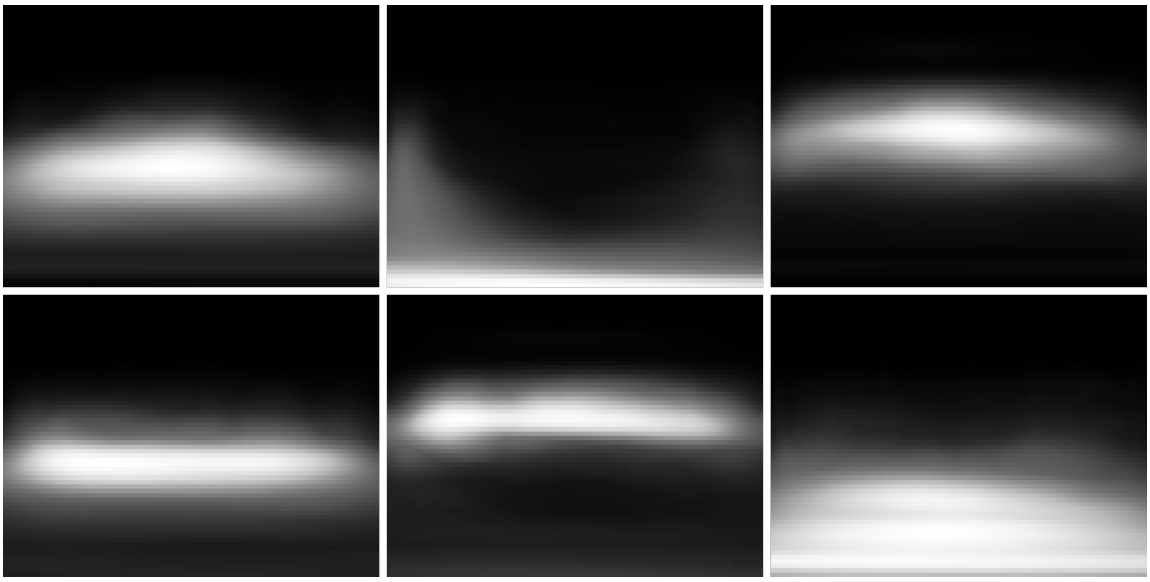

Fig. 8. Each primitive can be represented as a scalograms reconstructed from the NMF descriptors, and average Time and Energy descriptors. Top: primitives 1, 2, and 3 Bottom: primitives 4, 5 and 6. Average times and energy are (respectively for primitives 1 to 6): Time 0.86, 5.87, 0.67,4.70, 4.09,2.93 [s], and Energy & $1.86e+09$, $1.18e+09$, $1.00e+09$, $8.33e+09$, $6.01e+09$, $1.73e+09$

.

These primitives extracted using unsupervised learning can be compared to the movement qualities defined manually using a confusion matrix, as displayed in Figure 9. Overall, all the manual movement qualities can be expressed as a combination of our manually defined primitives. Specifically, we found that the primitive 1 (see Figure 7) can be considered mainly as the combination of *impulse* and *dynamic* gestures, primitive 2 as *steady* gestures, primitive 3 being equivalent to *impulse*, primitive 4 is the combination of *periodic* and *dynamic*, primitive 5 is *periodic* and *shaky*, primitive 6 is a combination of *smooth* and *dynamic*. This confirms that this unsupervised approach seems to be a valuable method to discover and quantify movement qualities from data. In particular, the association of data-driven primitives and manually defined movement qualities is consistent with the confusion matrix obtained on a recognition task in the previous section.

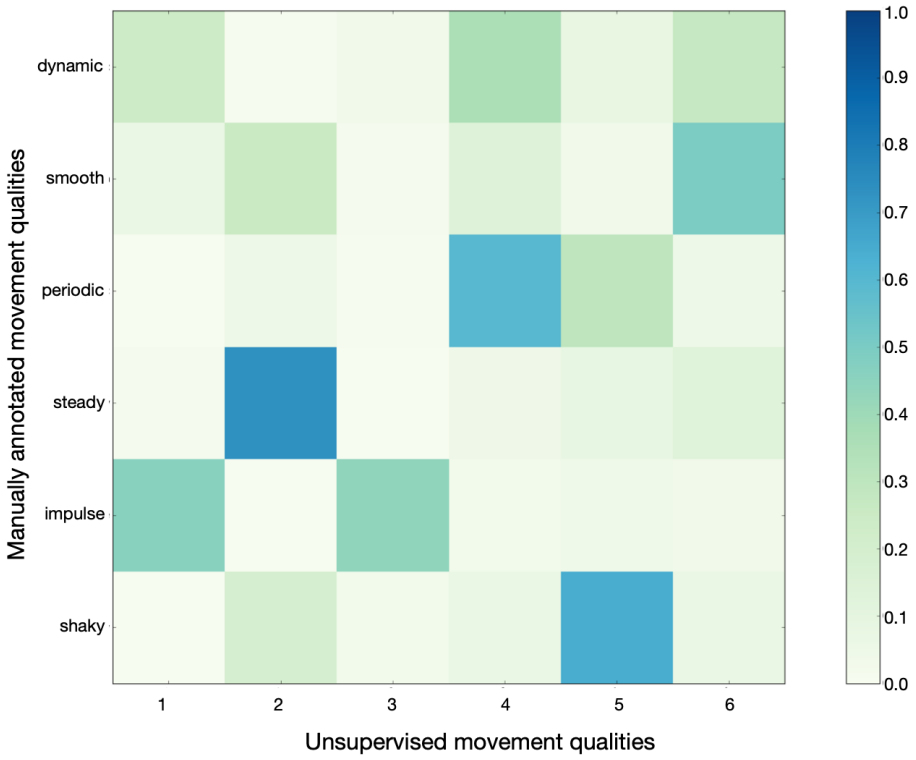

Fig. 9. Comparison of the unsupervised movement qualities using NMF with the manual qualities.

6 DISCUSSION AND CONCLUSION

We presented in this paper an approach to movement analysis based on the Continuous Wavelet Transform. While using such a well established signal processing technique is not novel *per se*, we believe that we provided researchers and artists in the MOCO community with several key elements to facilitate experimentation with this approach. We summarize below the main contributions and provide elements of discussion.

First, in contrast to fields such as audio processing that have well established several low level analysis methods for sound, the movement and computing community cannot currently rely on widely shared methods. Towards such a goal, we first recall why the Continuous Wavelet Transform presents several advantages in comparison with the Windowed Fourier Transform when working with movement and gesture data. The CWT provides an elegant solution to guarantee workable resolution from low to high frequencies. An important difference of movement data compared with audio signals relies in the importance of low frequency components (< 1 Hz) that make the WFT difficult to use in practice. The CWT indeed offers a better compromise to characterize movement data that includes smooth movements with important low frequency content, and more impulsive gestures presenting time-localized high frequency content. Our results on the recognition of manually annotated movement qualities confirm that the CWT outperforms the WFT.

Overall, we believe that the CWT offers a promising low-level representation that forms the base for descriptors of particular temporal and spectral movement patterns. We proposed several approaches using either a simple statistical

description of the scalogram, frequency tracking, or unsupervised learning of particular patterns through Non-negative Matrix Factorization. We believe that the prospect of using the CWT to derive high-level movement properties using non-supervised approaches is particularly promising. Interestingly, the various representations can be computed in a way that fuses information from the sensor's different dimensions, effectively making the representation invariant to orientation or sensor placement. The proposed approach still has a number of limitations. First, the fusion of scalograms across sensor channels makes for a representation that is somehow invariant to orientation. However, many applications involve gestures where spatial content is meaningful, in this case the CWT would need to be combined with other descriptors characterizing trajectories and spatial aspects of movement. Second, even with the proposed online approximation, the computational complexity of the continuous wavelet transform remains high, and its application to embedded systems with low computing power might be difficult. Finally, while the proposed methods represent a first step in exploiting the potential of the CWT, more elaborate approaches building could be developed, in particular by including phase information to complement the scalogram. Yet, the CWT seems particularly appropriate for movement presenting some level of periodicity, and further research would be necessary to understand its potential for more transient gestures.

We also elaborated on practical issues regarding the implementation of the CWT either offline or online for movement analysis. We provide open-source code as well as a Max external to facilitate its use in interactive application. Our online implementation can be used in real-time through a set of approximations and optimization methods. It has been used in several interactive applications, notably for tracking the fundamental frequency of walking or breathing [28], and in an interactive system for music conducting dedicated to a young public[6], where the average tempo is estimated from gestures, and used to control the speed of an orchestra recording. One characteristics of our implementation is that we use a minimal delay in each frequency band. While this distorts the spectral representation because the frequency bands are not aligned temporally, in our experience this characteristics can be interesting in interactive scenarios. Indeed, while latency is not essential in low-frequency movements, having low latency in higher-frequency movement enables the design of interactions that are reactive with impulsive gestures.

To conclude, we believe that the CWT holds promise as a low-level representation that could be used to build higher-level movement descriptors, in particular for modeling movement qualities. We hope that the work reported in this article can foster further investigations of the method in the movement and computing community.

ACKNOWLEDGMENTS

This research was supported by the project SkAT-VG with financial support of the Future and Emerging Technologies (FET) programme within the Seventh Framework Programme for Research of the European Commission, under FET-Open grant number: 618067. This research was partially supported by the ELEMENT project (ANR-18-CE33-0002) from the French National Research Agency.

REFERENCES

[1] Paul S Addison. 2005. Wavelet transforms and the ECG: a review. *Physiological Measurement* 26, 5 (2005), R155. http://stacks.iop.org/0967-3334/26/i=5/a=R01

[2] Sarah Fdili Alaoui, Frederic Bevilacqua, and Christian Jacquemin. 2015. Interactive Visuals as Metaphors for Dance Movement Qualities. *ACM Transactions on Interactive Intelligent Systems (TiiS)* 5, 3 (2015), 13.

[6] see https://ircamamplify.com/en/achievements/maestra-maestro-with-la-philharmonie-des-enfants/, accessed May 05 2022

[3] Sarah Fdili Alaoui, Baptiste Caramiaux, Marcos Serrano, and Frédéric Bevilacqua. 2012. Movement Qualities as Interaction Modality. In *Proceedings of the Designing Interactive Systems Conference* (Newcastle Upon Tyne, United Kingdom) *(DIS '12)*. Association for Computing Machinery, New York, NY, USA, 761–769. https://doi.org/10.1145/2317956.2318071

[4] Yvo Boers and JN Driessen. 2004. Multitarget particle filter track before detect application. *IEE Proceedings-Radar, Sonar and Navigation* 151, 6 (2004), 351–357.

[5] Antonio Camurri, Gualtiero Volpe, Giovanni De Poli, and Marc Leman. 2005. Communicating expressiveness and affect in multimodal interactive systems. *Ieee Multimedia* 12, 1 (2005), 43–53.

[6] Antonio Camurri, Gualtiero Volpe, Stefano Piana, Maurizio Mancini, Radoslaw Niewiadomski, Nicola Ferrari, and Corrado Canepa. 2016. The Dancer in the Eye: Towards a Multi-Layered Computational Framework of Qualities in Movement. In *Proceedings of the 3rd International Symposium on Movement and Computing* (Thessaloniki, GA, Greece) *(MOCO '16)*. Association for Computing Machinery, New York, NY, USA, Article 6, 7 pages. https://doi.org/10.1145/2948910.2948927

[7] Ulysse Côté-Allard, Cheikh Latyr Fall, Alexandre Drouin, Alexandre Campeau-Lecours, Clément Gosselin, Kyrre Glette, François Laviolette, and Benoit Gosselin. 2019. Deep learning for electromyographic hand gesture signal classification using transfer learning. *IEEE transactions on neural systems and rehabilitation engineering* 27, 4 (2019), 760–771.

[8] Samuel P Ebenezer and Antonia Papandreou-Suppappola. 2014. Multiple transition mode multiple target track-before-detect with partitioned sampling. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 8008–8012.

[9] Sarah Fdili Alaoui, Baptiste Caramiaux, M. Serrano, and Frédéric Bevilacqua. 2012. Movement Qualities as Interaction Modality. In *ACM Designing Interactive Systems (DIS'12)*. Newcastle, UK.

[10] Sarah Fdili Alaoui, Jules Françoise, Thecla Schiphorst, Karen Studd, and Frederic Bevilacqua. 2017. Seeing, Sensing and Recognizing Laban Movement Qualities. In *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI'17)*. ACM, Denver, CO, USA. https://doi.org/10.1145/3025453.3025530

[11] Jules Françoise, Sarah Fdili Alaoui, and Yves Candau. 2022. CO/DA: Live-Coding Movement-Sound Interactions for Dance Improvisation. In *CHI Conference on Human Factors in Computing Systems* (New Orleans, LA, USA) *(CHI '22)*. Association for Computing Machinery, New York, NY, USA, Article 482, 13 pages. https://doi.org/10.1145/3491102.3501916

[12] Donald Glowinski, Nele Dael, Antonio Camurri, Gualtiero Volpe, Marcello Mortillaro, and Klaus Scherer. 2011. Toward a minimal representation of affective gestures. *Affective Computing, IEEE Transactions on* 2, 2 (2011), 106–118.

[13] Wenchao Jiang and Zhaozheng Yin. 2015. Human Activity Recognition Using Wearable Sensors by Deep Convolutional Neural Networks. In *Proceedings of the 23rd ACM International Conference on Multimedia* (Brisbane, Australia) *(MM '15)*. Association for Computing Machinery, New York, NY, USA, 1307–1310. https://doi.org/10.1145/2733373.2806333

[14] Mubbasir Kapadia, I-kao Chiang, Tiju Thomas, Norman I Badler, and Joseph T Kider Jr. 2013. Efficient motion retrieval in large motion databases. In *Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*. ACM, 19–28.

[15] Adam Kendon. 2004. *Gesture: Visible action as utterance*. Cambridge University Press.

[16] Caroline Larboulette and Sylvie Gibet. 2015. A review of computable expressive descriptors of human motion. In *Proceedings of the 2nd International Workshop on Movement and Computing*. ACM, 21–28.

[17] Daniel D Lee and H Sebastian Seung. 1999. Learning the parts of objects by non-negative matrix factorization. *Nature* 401, 6755 (1999), 788–791.

[18] Guillaume Lemaitre, Hugo Scurto, Jules Françoise, Frédéric Bevilacqua, Olivier Houix, and Patrick Susini. 2017. Rising tones and rustling noises: Metaphors in gestural depictions of sounds. *PLOS ONE* 12, 7 (07 2017), 1–30. https://doi.org/10.1371/journal.pone.0181786

[19] Chih-Jen Lin. 2007. Projected gradient methods for nonnegative matrix factorization. *Neural computation* 19, 10 (2007), 2756–2779.

[20] Yasushi Makihara, Ryusuke Sagawa, Yasuhiro Mukaigawa, Tomio Echigo, and Yasushi Yagi. 2006. Gait Recognition Using a View Transformation Model in the Frequency Domain. In *Computer Vision – ECCV 2006*, Aleš Leonardis, Horst Bischof, and Axel Pinz (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 151–163.

[21] Stéphane Mallat. 2009. *A Wavelet Tour of Signal Processing (Third Edition)*. Academic Press, Boston. https://doi.org/10.1016/B978-0-12-374370-1.X0001-8

[22] David McNeill. 2008. *Gesture and thought*. University of Chicago Press.

[23] Alejandro Melendez-Calderon, Camila Shirota, and Sivakumar Balasubramanian. 2021. Estimating Movement Smoothness From Inertial Measurement Units. *Frontiers in Bioengineering and Biotechnology* 8 (2021). https://doi.org/10.3389/fbioe.2020.558771

[24] Meinard Müller, Andreas Baak, and Hans-Peter Seidel. 2009. Efficient and robust annotation of motion capture data. In *Proceedings of the 2009 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. ACM, 17–26.

[25] Kevin P Murphy. 2012. *Machine learning: a probabilistic perspective*. MIT press.

[26] Anna Nedorubova, Alena Kadyrova, and Aleksey Khlyupin. 2021. Human Activity Recognition using Continuous Wavelet Transform and Convolutional Neural Networks. *arXiv preprint arXiv:2106.12666* (2021).

[27] Radoslaw Niewiadomski, Maurizio Mancini, Stefano Piana, Paolo Alborno, Gualtiero Volpe, and Antonio Camurri. 2017. Low-Intrusive Recognition of Expressive Movement Qualities. In *Proceedings of the 19th ACM International Conference on Multimodal Interaction*. 230–237.

[28] Mirjana Prpa, Kıvanç Tatar, Jules Françoise, Bernhard Riecke, Thecla Schiphorst, and Philippe Pasquier. 2018. Attending to Breath: Exploring How the Cues in a Virtual Environment Guide the Attention to Breath and Shape the Quality of Experience to Support Mindfulness. In *Proceedings of the 2018 Designing Interactive Systems Conference* (Hong Kong, China) *(DIS '18)*. Association for Computing Machinery, New York, NY, USA, 71–84.

https://doi.org/10.1145/3196709.3196765

[29] Changzhen Qiu, Zhiyong Zhang, Huanzhang Lu, and Huiwu Luo. 2015. A Survey of Motion-Based Multitarget Tracking Methods. *Progress In Electromagnetics Research B* 62 (2015), 195–223. https://doi.org/10.2528/PIERB15010503

[30] D.J. Salmond and H. Birch. 2001. A particle filter for track-before-detect. In *Proceedings of the 2001 American Control Conference. (Cat. No.01CH37148)*, Vol. 5. 3755–3760 vol.5. https://doi.org/10.1109/ACC.2001.946220

[31] Norbert Schnell, Axel Röbel, Diemo Schwarz, Geoffroy Peeters, Riccardo Borghesi, et al. 2009. MuBu and friends–assembling tools for content based real-time interactive audio processing in Max/MSP. In *ICMC*.

[32] Shiqi Yu, Liang Wang, Weiming Hu, and Tieniu Tan. 2004. Gait Analysis for Human Identification in Frequency Domain. In *Third International Conference on Image and Graphics (ICIG'04)*. IEEE, 282–285. https://doi.org/10.1109/ICIG.2004.72

[33] Diego Silang Maranan, Sarah Fdili Alaoui, Thecla Schiphorst, Philippe Pasquier, Pattarawut Subyen, and Lyn Bartram. 2014. Designing for Movement: Evaluating Computational Models Using LMA Effort Qualities. In *Proceedings of the 32Nd Annual ACM Conference on Human Factors in Computing Systems (CHI '14)*. Toronto, Ontario, Canada, 991–1000. https://doi.org/10.1145/2556288.2557251

[34] Christopher Torrence and Gilbert P Compo. 1998. A practical guide to wavelet analysis. *Bulletin of the American Meteorological society* 79, 1 (1998), 61–78.

[35] Gualtiero Volpe. 2003. Computational models of expressive gesture in multimedia systems. *InfoMus Lab, DIST–University of Genova* 12 (2003). ftp://ftp.infomus.org/pub/Publications/2003/VolpeDissertation.pdf

[36] Yingen Xiong and Francis Quek. 2006. Hand Motion Gesture Frequency Properties and Multimodal Discourse Analysis. *International Journal of Computer Vision* 69, 3 (2006), 353–371. https://doi.org/10.1007/s11263-006-8112-5

A FREQUENCY RIDGE EXTRACTION USING TARGET TRACKING

In this section, we describe the implementation of a multi-target tracker for characterizing multiple frequency modes in dynamic gestures.

A.1 Multi-target Track-before-detect in the Wavelet Domain

Our approach follows recent developments in multi-target tracking in Radar tracking and computer vision systems. In particular, we consider track-before-detect (TBD) approaches to multiple object tracking where the target detection and tracking is performed jointly [4, 8, 29, 30]. Such approaches differ from standard detect-before-track methods where a set of potential targets are first detected — e.g. from image segmentation in the case of object tracking, — and then filtered using dynamic system modeling.

In TBD, the decision is made at the end of the processing chain, and therefore integrates all information over time. As a result, the tracking is less sensitive to errors in the detection step. Moreover, recursive approaches to TBD avoid the classical problem of data association between tracked targets and detected candidates: no thresholding is necessary, which removes the need for explicit association. Finally, it makes it possible to estimate additional parameters about the target, such as its intensity or size.

A.1.1 Problem Definition. We aim at simultaneously tracking a maximum number K of frequency components of the movement — thereafter called 'targets', — from measurements of the wavelet power spectrum computed from one or several sensors. We assume that each target $k \in \{1, \cdots, K\}$, if present, is specified by a time-varying position $x^{(k)}$ in the wavelet scale domain — which is associated with the movement's fundamental frequency, — and a time-varying amplitude $a^{(k)}$. We assume that each frequential component is projected over the wavelet spectrum as a Gaussian, centered around $x^{(k)}$ and spread on the wavelet space with width $\Sigma^{(k)}$. Our goal is to estimate, from measurements of wavelet power spectrum, the number of components present at the current time step as well as their characteristic parameters $\left\{f^{(k)}, a^{(k)}, \Sigma^{(k)}\right\}_{k=1}^{K}$.

A.1.2 System Dynamics. We assume that the position of each target evolves according to linear dynamics. We further assume a constant velocity model, meaning that, on a short time scale, each ridge evolves linearly in the wavelet

domain.[7] For each target $k \in \{1, \cdots, K\}$, we assume the following time-invariant state equation:

$$\boldsymbol{s}_{t+1} = \boldsymbol{f}(\boldsymbol{s}_t) + \boldsymbol{g}(\boldsymbol{s}_t)\boldsymbol{w}_t \tag{13}$$

where the process noise $\boldsymbol{w}_t$ is assumed to be standard Gaussian white noise. The hidden state $\boldsymbol{s}_t$ at time step t for a given target is composed by the target's position, velocity, amplitude, and spread:

$$\boldsymbol{s}_t = \begin{bmatrix} x_t \\ v_t \\ a_t \\ \Sigma_t \end{bmatrix} \tag{14}$$

The system dynamics function $\boldsymbol{f}$ and process noise input model $\boldsymbol{g}$ can be defined under a time-invariance assumption as

$$f(\boldsymbol{s}_t) = \begin{pmatrix} 1 & \delta t & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \boldsymbol{s}_t \quad , \quad g(\boldsymbol{s}_t) = \begin{pmatrix} \sigma_x \delta t^3/3 & \sigma_x \delta t^2/2 & 0 & 0 \\ \sigma_x \delta t^2/2 & \sigma_x \delta t & 0 & 0 \\ 0 & 0 & \sigma_a \delta t & 0 \\ 0 & 0 & 0 & \sigma_\Sigma \delta t \end{pmatrix} \tag{15}$$

where δt is the sampling period and σ_x, σ_a and σ_Σ are the expected maximum acceleration of the target, maximum change in amplitude, and maximum change in spread.

A.1.3 Measurement Model. By analogy with radar sensors or optical system, we consider the measured power as reflected on a unidimensional observation space in the wavelet domain. A measurement $\boldsymbol{z}_t$ therefore consists of N power measurements z_t^i in the scale domain, where N is the number of bands of the CWT. In a multi-target setting, we hypothesize that the power measurements in the wavelet spectrum result from the superposition of the power emitted by each target, if present, yielding the following form:

$$\boldsymbol{z}_t = \sum_{k=1}^{K} \delta_k \boldsymbol{h}\left(\boldsymbol{s}_t^{(k)}\right) + \boldsymbol{n}_t \tag{16}$$

where δ_k is a binary indicator variable specifying if target k is present, and $\boldsymbol{h}\left(\boldsymbol{s}_t^{(k)}\right)$ is the reflected power of the kth target in the wavelet space, assumed to be Gaussian:

$$h^i\left(\boldsymbol{s}_t^{(k)}\right) = \frac{a_t^{(k)}}{\sqrt{2\pi\Sigma_t^{(k)}}} \exp\left[-\frac{\left(x_t^{(k)} - i\right)^2}{2\Sigma_t^{(k)}}\right] \quad , \quad \forall i \in \{0, \cdots, N-1\} \tag{17}$$

A.1.4 Track Birth and Death. We propose to model the possibility of birth and death of the frequency ridges using a jump Markov process, as often proposed in multi-target tracking systems [4, 8]. Instead of using a single *mode* state variable, as proposed by Boers et al. [4], we keep a vector of binary indicator variable $\boldsymbol{\delta} = [\delta_1, \cdots, \delta_K]^\top$. We assume that track birth and death are ruled a first-order Markov process where the probability of switching between different *modes* — i.e. configurations of the targets present at a given time steps — only depends on the mode at the previous time step. Constraints on track birth and death are then specified through a transition matrix that allows the birth of a new track with probability p_{birth} and the death of a new track with probability p_{death}. Additionally, it prevents the

[7] We assume that the scales are distributed as fractional powers of 2, meaning that the associated distribution in the Fourier domain is logarithmic in frequency.

simultaneous birth and death of two tracks as well as the simultaneous birth of several tracks, in order to avoid target switching.

A.2 Particle Filter Implementation

We derived a a Bayesian solution to the estimation problem introduced in the previous section. While the state dynamics are linear and could be optimally solved analytically, the non-linear nature of the observation model makes exact inference intractable. The track-before-detect can be solved using Sequential Monte Carlo estimation — also called particle filtering. Our particle filter implementation follows a standard Particle Filter implementation, as described in [4]. An example of ridge tracking from the scalogram is shown in Figure 10.

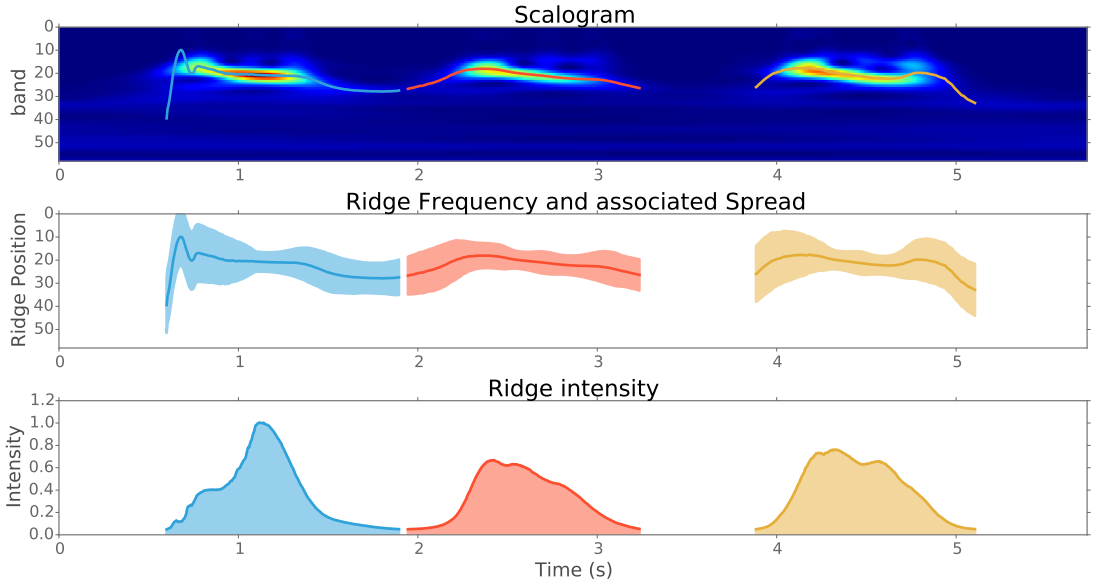

Fig. 10. Example of Ridge tracking from the scalogram. The top plot represents the scalogram computed from the 3D acceleration of an example gesture from a dataset of gestural imitations of sounds. The middle and bottom plots respectively represent the tracked ridge frequency and intensity. We used a single target with 100 particles.

B NON-NEGATIVE MATRIX FACTORIZATION

Mathematically, NMF allows for decomposing the non-negative matrix V as a product of two non-negative matrices W and H:

$$V_{f,t} \approx (W \times H)_{f,t} = \sum_{i=1}^{k} W_{f,k} H_{k,t}. \tag{18}$$

where k is a number of basis components. $V \in \mathbb{R}^{f \times t}$ is the original non-negative data, $W \in \mathbb{R}^{f \times k}$ the basis vector decomposition and $H \in \mathbb{R}^{k \times t}$ the weight matrix. Each column of V represents the whole dataset. Here, each matrix row represents one scalogram. In W, each column is referred to as a basis vector, representing the k components. Rows of H

stand for the gain of corresponding basis vector. k is chosen so that $(f + t) \times k < (f \times t)$. The product $W \times H$ presents a compressed form of the original data V.

Technically, W and H are calculated resolving the equation:

$$\min_{W,H} f(W,H) \equiv \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{m} (V_{i,j} - (WH)_{i,j})^2 \quad \text{subject to } W_{i,a} \geq 0, H_{b,j} \geq 0, \forall i, a, b, j. \tag{19}$$

There are several approaches to solve (19). In this paper, we used the Alternating Nonnegative Least Squares Matrix Factorization Using Projected Gradient or LSNMF as described in [19]. It converges fast and provides small approximation errors.

Then, each gesture scalogram b (new or already known) is described as a linear combination of the k basis vectors, where x is the weight of each component to be calculated by resolving:

$$\arg\min_{x} = \|Ax - b\|^2 \text{ for } x >= 0 \tag{20}$$

where A the vector of components W. Only additive combinations are allowed. Any given scalogram does not have to use all the available components.