Physics Reports 985 (2022) 1–23

Contents lists available at ScienceDirect

Physics Reports

journal homepage: www.elsevier.com/locate/physrep

Wavelets behind the scenes: Practical aspects, insights, and perspectives

Rodrigo Capobianco Guido

Instituto de Biociências, Letras e Ciências Exatas, Unesp - Univ Estadual Paulista (São Paulo State University), Rua Cristóvão Colombo 2265, Jd Nazareth, 15054-000, São José do Rio Preto - SP, Brazil

ARTICLE INFO

Article history:
Received 6 June 2022
Accepted 1 August 2022
Available online 27 August 2022
Editor: Itamar Procaccia

Keywords:
Wavelets
Wavelet transforms
Continuous wavelet transform
Discrete wavelet transform
Discrete-time wavelet transform

ABSTRACT

Over the years, wavelet-based analyses have been responsible for remarkable achievements in physics and related sciences. Nevertheless, a deep inspection on wavelet-based strategies described in recent scientific papers, dissertations, and theses reveals that a significant number of authors, i.e., students and even researchers with a modest background on signal analysis, still misunderstand the fundamentals of wavelets. One classical source of confusion, for instance, involves two different but related approaches used to perform discrete-time wavelet transformations and their inverses: algebra- and filter-based. Although the latter is usually adopted in practice, the former reveals the beauty of multiresolution analysis over Heisenberg's uncertainty principle, showing what really happens behind the scenes. Thus, based on the solid and easy-to-follow explanations provided in this smoothly written tutorial-review article, interested readers will definitively comprehend the different types of wavelet transforms and their specific applications, getting hands-on experience and insights on how to extract the most of their research by using that powerful tool. Because its mission is clarification, example wavelet-related applications are provided in this document to stimulate state-of-the-art research in a diversity of branches in physics.

Contents

E-mail address: guido@ieee.org.
URL: http://www.sjrp.unesp.br/~guido/.

https://doi.org/10.1016/j.physrep.2022.08.001
0370-1573/

1. Introduction

Twenty years ago, when I was facing a challenge during my PhD research work on applied physics, aiming at the identification of patterns coming from a neurophysiological signal, a colleague who has passed away a few years back told me: "... *wavelets will solve your problem*". Since I had never heard anything about that tool, he sent me the article [1] and then I started to search for additional information. From that time, I could't stop working with the jealous I had discovered anymore and, currently, my colleague's voice frequently echoes in my mind forcing me to meditate on the distinguished role wavelets have being playing in physics and related sciences.

The principal barrier I found when I started working with wavelets was the amount of divergent information available. Distinguished scientific books and papers written by mathematicians, such as those in Refs. [2,3], usually explain wavelets based on the concepts of linear algebra. Conversely, relevant academic pieces of work authored by applied professionals, i.e., physicists, engineers, computer scientists, and so on, commonly use a signal processing-based approach to study fundamental wavelet concepts, as observed, for instance, in paper [4] and in the sixth chapter of book [5]. Although both treatments are perfectly valid and play a relevant role, students, fresh technical people, and even experienced researchers use to get lost in the middle of so many details which, sometimes, seem incongruent and even disconnected.

Thus, driven by a substantial level of questioning received over the years, I decided to write this review article focusing on one of the most appropriate tools for time–frequency signal analysis, coding, and treatment [6]. Because it is largely applicable in a diversity of fields, such as those documented in Refs. [7–11], and [12], I kindly invite readers to follow this piece of work, taking advantage of each detail. Ensuring an equilibrium between clarity and correctness, this paper is organized as follows. Basic concepts on wavelets are described in Section 2. Then, Section 3 explains different types of Wavelet Transforms (WTs) and what happens behind the scenes, i.e., clarifies how wavelet-derived entities are interrelated. Proceeding, illustrative experiments concerning the state-of-the-art in the field can be found in Section 4 and, lastly, the readers can find the conclusions followed by the references.

2. Why wavelets?

Wavelets, also known as small waves or *ondelettes* in French language, are non-periodic oscillations that exist over a short-time interval, such as those shown in Fig. 1. Just so simple! WTs, however, are mathematical conversions based on particular wavelets, where restrictions apply. Allowing for signal analysis, feature extraction, coding, filtering, compression, and so on, WTs are essentially used to convert an input signal from either the time or the space domain to the time–frequency or space–frequency joint domain, respectively. In other words, for input signals in time-domain, the transformation allows for the time-support of frequencies to be found, i.e., we can investigate whether specific frequencies exist or not in the input and, if so, in which portion of that signal they occur. Accordingly, for input signals in space-domain, i.e., images, the transformation reveals in which part of the bidimensional data specific frequencies take place, in case they are present.

Prior to the advent of WTs, spectral investigation used to be based on Fourier Transforms (FTs) [13], characterizing a straight conversion from the time, or space, to the frequency domain by using sine and cosine orthogonal functions as basis. Notably and opposed to WTs, FTs are unable to reveal the time-support of frequencies, as exemplified in Fig. 2. This is the main reason behind the advent of WTs decades ago.

Notably, if we adapt a sliding window [13] to FTs, permitting only specific portions of the input signal to be analyzed, i.e., transformed to the frequency domain, we may find the time-support of frequencies. This is just the role Short-Time Fourier Transform (STFT) [14] plays, however, there is one fundamental drawback: the permanent window length, regardless of the frequency being analyzed. Narrow windows may mischaracterize low frequency components because they need sufficient time to complete at least one cycle. Conversely, wide windows may capture more than one frequency component occurring separately in the corresponding time interval, forbidding the intended localization, i.e., the condition of stationarity may be violated. In other words and as stated by Heisenberg's uncertainty principle [3], high time-resolution implies low frequency-resolution and vice-versa. This time–frequency tradeoff is illustrated in Fig. 3.

Since this tutorial is ***definitively not*** dedicated to FTs, further comments on that subject will ***not*** be included hereafter. Nevertheless, the humble intuitive considerations on Fourier-based spectral investigation made up to this point suffice for us. Interested readers may consult, for instance, Refs. [15,16] for a profound explanation. Noteworthy and contrary to FTs, WTs intelligently explore different frequency bands based on adaptive window lengths and positions.

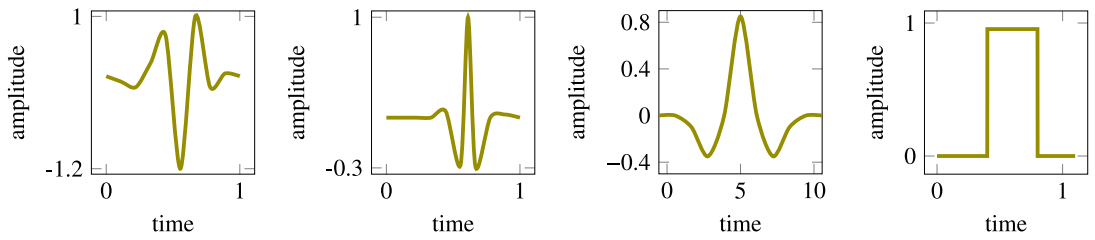
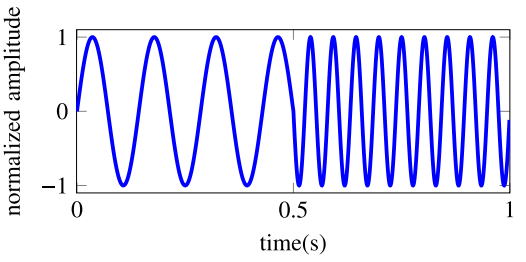
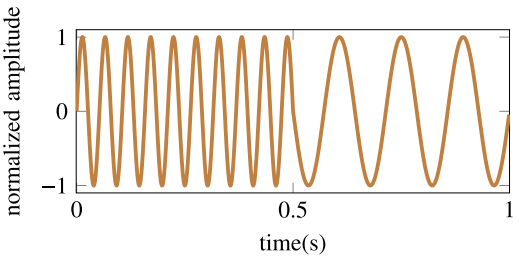


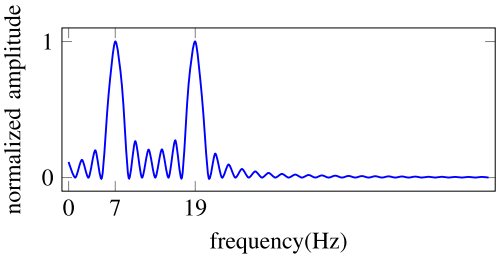
Fig. 1. Example wavelets. Hypothetical scales for both axes, i.e., time and amplitude, are used.



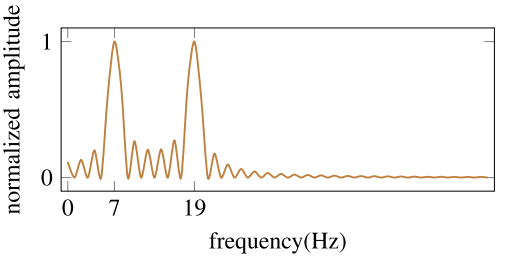
(a) time-domain signal where the 7Hz sinusoidal component appears from 0 to 0.5 seconds and the 19 Hz component exists in the interval from 0.5 to 1.0 seconds



(b) time-domain signal where the 19Hz sinusoidal component appears from 0 to 0.5 seconds and the 7 Hz component exists in the interval from 0.5 to 1.0 seconds



(c) in blue we can see the FT module from the signal in (a)



(d) in brown we can see the FT module from the signal in (b)

Fig. 2. Two time-domain signals in (a) and (b): the former with a 7 Hz sine wave preceding a 19 Hz sine wave, and the latter with a 19 Hz sine wave preceding a 7 Hz sine wave. Additionally, the module of FTs for the signals in (a) and (b) appear in (c) and (d), respectively. Clearly, (c) and (d) are indistinguishable, i.e., ordinary Fourier analysis is ***unable*** to reveal ***where*** the 7 Hz and 19 Hz sine waves are localized in the respective time-domain signals.

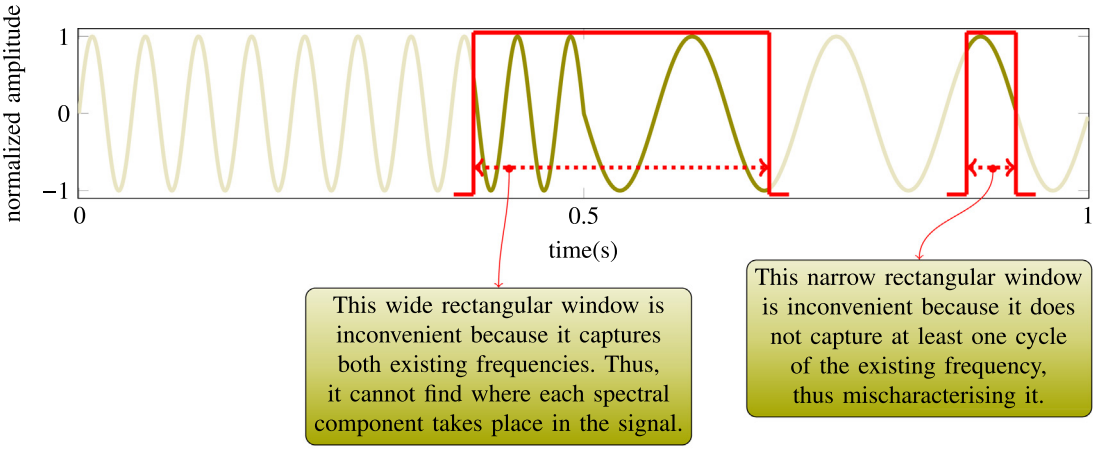
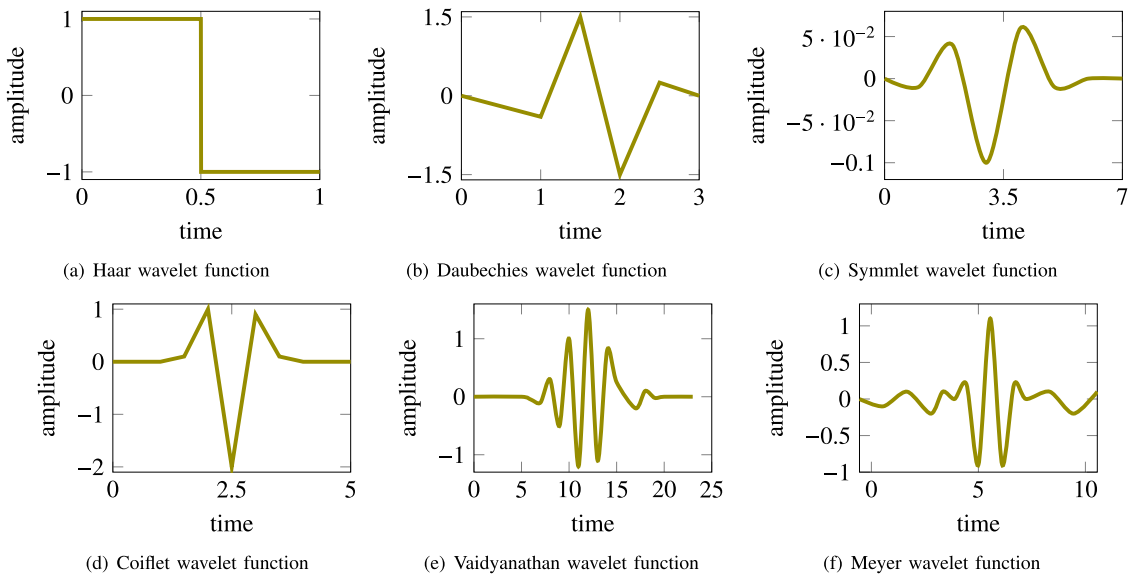


Fig. 3. A signal with two non-overlapped frequency components, as in Fig. 2-(b), i.e., 19 Hz and 7 Hz, where we can observe the main STFT drawback related to the window width used in the analysis: the adoption of a fixed-length sliding window, either wide or narrow, is inconvenient.



(a) Haar wavelet function (b) Daubechies wavelet function (c) Symmlet wavelet function

(d) Coiflet wavelet function (e) Vaidyanathan wavelet function (f) Meyer wavelet function

Fig. 4. Some of the popular and well-known wavelet functions: (a) Haar; (b) Daubechies; (c) Coiflets; (d) Symmlets; (e) Vaidyanathan; (f) Mexican Hat; (g) Meyer; (h) Morlet. Hypothetical normalized scales for both axes, i.e., time and amplitude, are adopted.

3. Types of WTs

3.1. Overview

Broadly divided into two categories, i.e., continuous (CWT) and discrete (DWT), WTs use *wavelets* instead of sinusoids to convert a signal from the time to the time–frequency domain. For CWTs, the associated wavelets operate over every possible window lengths and positions, whereas for DWTs they make use of a pre-defined subset of those possibilities. At a specific length and position, we use to say that the wavelet is scaled and shifted, or equivalently, dilated and translated, respectively. On one hand, CWTs aim at an ideal time–frequency mapping, creating a representation which offers excellent time and frequency resolutions, although with a considerable cost. On the other hand, DWTs are mathematically cheaper, presenting a sufficient resolution for a variety of applications.

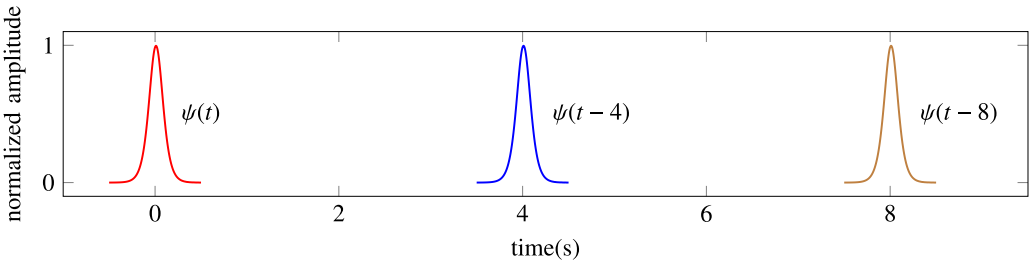
Curiously and as readers are about to understand, both CWT and DWT are used to analyze ***continuous***-time signals! Conversely, discrete-time signals are analyzed based on a DWT variant known as Discrete-Time Wavelet Transform (DTWT), which is widely employed in practice. In the next subsections, CWTs, DWTs, and DTWTs are explained. Complementarily, the interesting connection involving DWTs and DTWTs is also explained.

3.2. CWTs: where the story begun

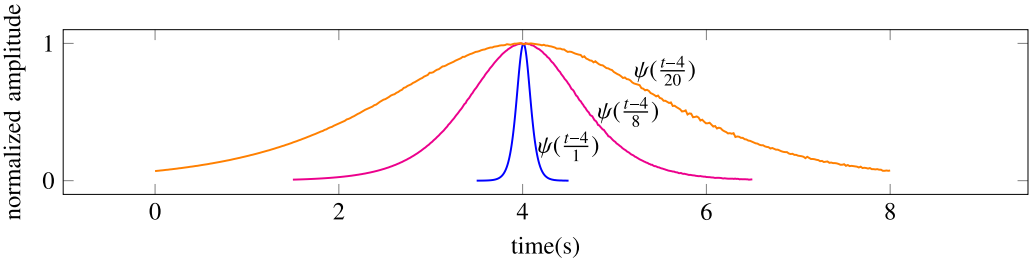
While commenting on FTs in Section 2, we had the opportunity to observe rectangular windows, shown in Fig. 3, covering specific portions of the signal under analysis. Nevertheless, for both FTs and WTs, rectangles are not always the most convenient waveform shapes. From traditional signal processing books, such as [15], we can learn that the application of a rectangular window, which is equivalent to perform an abrupt signal truncation, introduces ripples in frequency domain, such as those observed around the main spikes in Fig. 2-(c,d). To overcome that problem, a variety of wavelets might be used instead, as the popular ones in Fig. 4. Those wavelets are hereafter referred to as *wavelet functions of time* t or simply *wavelet functions* [1–3]. Since this tutorial does not emphasize the CWT, detailed comments on the corresponding wavelet shapes were omitted. Interested readers may want to consult Ref. [3] for further details.

Let $\psi(t)$ be the wavelet function of time t, regardless of its shape. Then, $\psi(t-b)$ for $b \in \Re$ is its b-second shifted version. Equivalently, $\psi(t-b)$ is the sliding window for which placement depends on b, as shown in Fig. 5-(a). Accordingly, we can insert the parameter $a \in \Re$ so that the function becomes $\psi(\frac{t-b}{a})$. As a varies, the wavelet function is dilated, i.e., the window length is modified, as we can observe in Fig. 5-(b). Thus, $\psi(\frac{t-b}{a})$ is the sole function intelligently controlling the entire process.

Observably, by shifting the wavelet function based on b and as shown in Fig. 5-(a), the time signal under analysis can be entirely covered. Accordingly, by stretching or shrinking $\psi(t)$ based on a, as shown in Fig. 5-(b), its width becomes proper to analyze certain frequencies, acting as a bandpass filter. Therefore, this is just how WTs work: the time signal under analysis becomes filtered as it passes through the dilated and translated versions of the wavelet function and, in addition, time–frequency localization depends primarily on the wavelet size. Consequently, narrow windows at high frequencies



(a) shifting the wavelet function



(b) wavelet dilation

Fig. 5. (a): A hypothetical wavelet function $\psi(\frac{t-b}{a})$, where $a = 1$ being shifted as b varies; (b): the same function where $b = 4$ being dilated as a varies.

can be used for better time resolution and, conversely, wide windows at low frequencies might be adopted for better frequency resolution. Noteworthy is the fact that *wavelet transform* is different from *wavelet function*: the latter is used to implement the continuous version of the former.

Grouping together all the concepts exposed up to this point, we can formally define CWTs according to Eq. (1).

$$\text{CWT}(a, b) = \frac{1}{\sqrt{a}} \int_{-\infty}^{\infty} x(t)\psi\left(\frac{t-b}{a}\right) \quad dt \qquad , \tag{1}$$

where $x(t)$ is the input signal under analysis and $\frac{1}{\sqrt{a}}$ is simply a normalization factor [4].

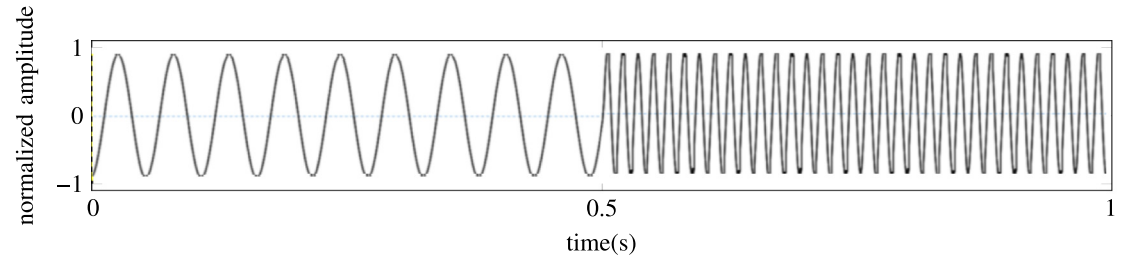
CWT working becomes clearer based on Eq. (1), i.e., CWT(a, b) correlates $x(t)$ with $\psi(\frac{t-b}{a})$ over time and according to the scale, quantifying the resulting likelihoods, as shown in the example signal of Fig. 6. Colors and their intensities, playing the role of a third axis in the graphical representation in sub Fig. 6-(b), are used to represent the respective correlation amplitudes. Observed from another angle, $C(a, b)$ converts the input signal from the time to the time–frequency domain, where the latter is formed by a set of likelihood amplitudes as a function of time and scale.

Specially relevant is the fact that CWTs present a drawback in case we are really interested in a full spectral coverage: they can ***not*** be computed, in practice, for all existing values of a and b. Thus, we should move forward to the discrete world.

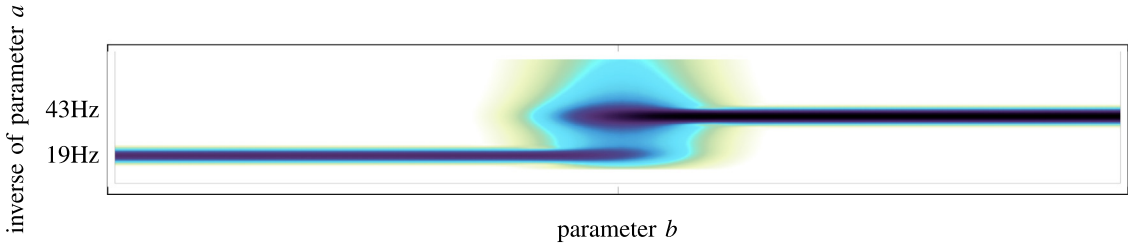
3.3. DWTs: a step towards computer-based approaches

Let us examine the specific case, known as DWT, where the CWT is adapted by adjusting a in such a way that the bandpass wavelet function is progressively stretched by factors of 2, implying in halved bandwidths. Readers are kindly requested to observe that, ***even using this particular discretization***, we would need an ***infinite*** number of related operations ***to cover the frequency 0 Hz***, as illustrated in Fig. 7. Thus, to circumvent that issue, we need not only a bandpass filter but also a lowpass filter, characterizing the *scaling function*, i.e., $\phi(t)$, which is the orthogonal time-domain complement of $\psi(t)$. Equivalently, DWTs make possible to represent a signal in terms of shifted and dilated versions of $\psi(t)$ complemented with shifted versions of $\phi(t)$. Thus, the scaling and wavelet functions, also known as *father wavelet* and *mother wavelet*, respectively, form a basis for signal representation.

The procedure just explained is mathematically represented by choosing $a = 2^{-j}$ and $b = k \cdot 2^{-j}$, for $j, k = 1, 2, \ldots$, and so on. Thus, $\psi(\frac{t-b}{a}) = \psi(\frac{t-k2^{-j}}{2^{-j}}) = \psi(2^j t - k)$. Assuming the normalization factor $\frac{1}{\sqrt{a}}$ as in Eq. (1), we have $\psi(\frac{t-b}{a}) = \frac{1}{\sqrt{a}}\psi(2^{-j}t - k) = \frac{1}{\sqrt{2^{-j}}}\psi(2^{-j}t - k) = 2^{\frac{j}{2}}\psi(2^{-j}t - k)$. Accordingly, the shifted and normalized version of the scaling function is $\phi(t - b) = a\phi(t - k2^{-j}) = 2^{-j}\phi(2^{-j}t - k)$. Particularly, interpreting j as being the decomposition level, or refinement level, and k as the integer variable controlling the shifting, our basis has been completely defined.



(a) the input time signal $x(t)$



(b) graphical representation of CWT(a, b) implemented by using a Gaussian-like wavelet

Fig. 6. (a): The signal $x(t)$ where from 0 to 0.5 s and from 0.5 to 1 s the frequencies are 19 Hz and 43 Hz, respectively; (b): CWT(a, b) plot for $x(t)$ where the vertical axis corresponds to $\frac{1}{a}$, i.e., the inverse of scale which, in practice, corresponds to the frequency.

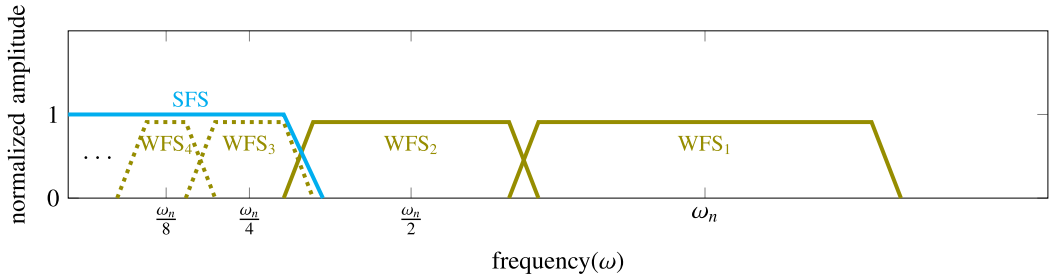


Fig. 7. In cyan: the scaling function spectrum (SFS) overcoming the infinite number of stretched wavelet functions that would be required to reach the frequency 0 Hz; in olive: four example wavelet function spectrums (WFSs). If, for instance, WFS_1 behaves as a bandpass filter allowing for frequencies from 10 to 20 Hz to go through, then WFS_2 matches the range from 5 to 10 Hz, WFS_3 concerns the range from 2.5 to 5 Hz, WFS_4 spectrum goes from 1.25 to 2.5 Hz, and so on. Thus, to reach the frequency 0 Hz, an ***infinite*** number of dilated wavelet functions would be required. Therefore, the feasible solution is to use a scaling function playing the role of a low-pass filter.

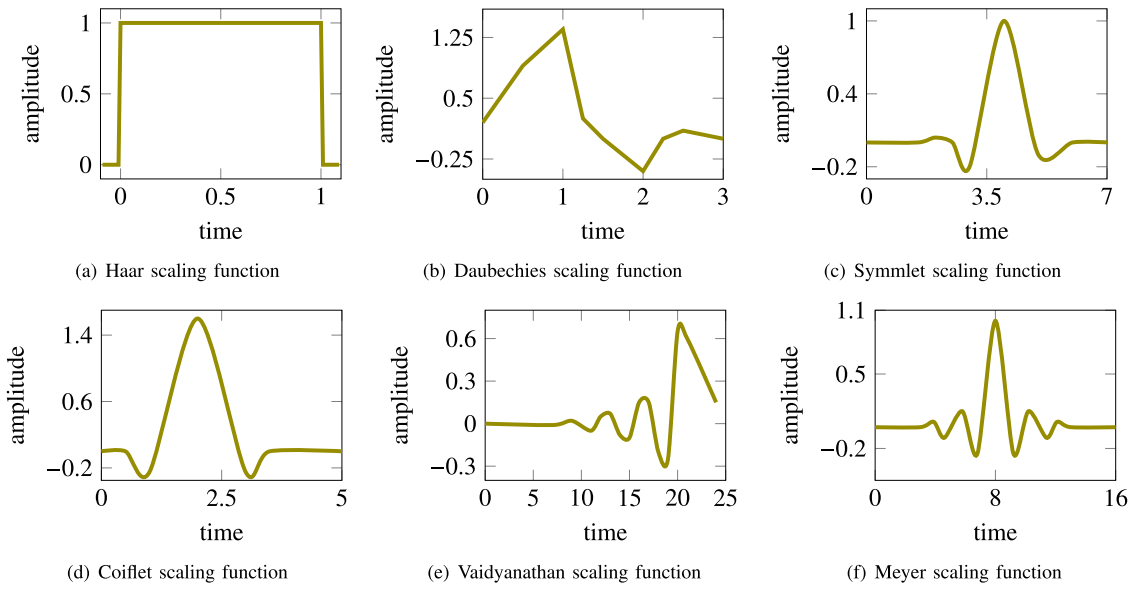
Readers are expected to note that CWTs adopt only wavelet functions, whereas DWTs make use of both scaling and wavelet functions! To complement the examples provided in Fig. 4, which exhibits only wavelet functions, Fig. 8 shows the respective scaling functions. Another noteworthy aspect is that DWTs use the term "discrete" in their names just because of the finite set of values used for a and b in the basis functions. Nevertheless, the input signals that DWTs analyse are still continuous in time! Thus, since computers were created for the discrete world, our entities have not yet matched perfectly!

The most natural solution for the inadequacies observed is to turn everything discrete: the input signal and the scaling and wavelet functions. That particular case originates the DTWT which is, theoretically and based on our considerations, different from the DWT. In practice, however, ***a considerable number of researchers worldwide use to refer to the former transform***, which characterizes the multiresolution analysis, ***by using the latter name and/or acronym***. This is a ***very important*** note for readers to consider in order to correctly distinguish among the different types of WTs.

3.4. DTWTs and wavelet-packets: everything discrete and realizable based on multiresolution analysis

To convert the input continuous-time signal under analysis to its discrete version, we basically observe Nyquist's criterion [3]. Consequently, readers might think that the basis functions we need for realizing the DTWT could be obtained just by sampling the scaling and wavelet functions as well. Nevertheless, this is ***not*** correct!

Although CWT and DWT use the scaling and wavelet functions as basis for conversion from the time to the time–frequency domain, DTWT adopts an analysis filterbank composed of two filters: $h[\cdot]$, exhibiting a lowpass frequency



(a) Haar scaling function (b) Daubechies scaling function (c) Symmlet scaling function

(d) Coiflet scaling function (e) Vaidyanathan scaling function (f) Meyer scaling function

Fig. 8. Some of the popular and well-known scaling functions: (a) Haar; (b) Daubechies; (c) Coiflets; (d) Symmlets; (e) Vaidyanathan; (f) Mexican Hat; (g) Meyer; (h) Morlet. Hypothetical normalized scales for both axes, i.e., time and amplitude, are adopted.

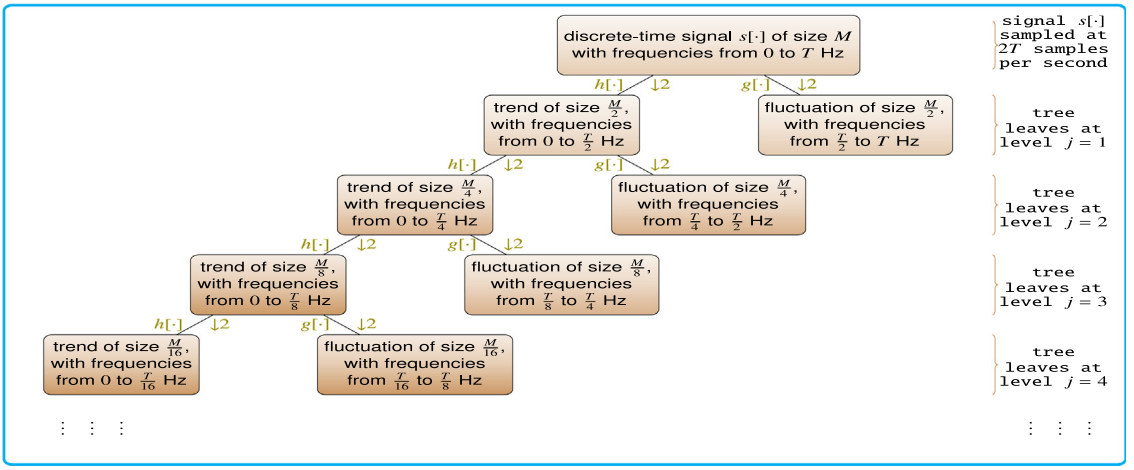


Fig. 9. Regular DTWT decomposition tree, exemplified for the specific resolution level $j = 4$. The downarrows followed by the number "2" represent downsamplings by two, as explained in the text.

response, and $g[\cdot]$, with a highpass mirrored frequency response. Those filters coefficients are ***not*** samples of the scaling and wavelet functions. Different from CWTs and DWTs, the transformation strategy in DTWTs is based on subsequent tree-structured filterings, followed by downsamplings, as shown in Fig. 9 and as detailed ahead.

From DSP principles, we know that a discrete-time signal sampled at a rate of $2T$ equally-spaced samples per second contains frequencies up to T Hz. Accordingly and opposite to analog filters, digital filters do ***not*** have a specific ***regular*** cutoff frequency in Hz. Instead, they exhibit an ***angular*** spectral response which matches a certain regular frequency depending on the sampling rate of the signal being filtered. As an example, if we have a discrete-time signal $s[\cdot]$ sampled at $2T = 2000$ samples per second, the maximum frequency it carries is $T = 1000$ Hz. In that case, if we apply, for instance, a halfband lowpass digital filter to it, frequencies up to 500 Hz remain in the resulting signal. Accordingly, keeping the digital filter intact and using a different input signal sampled at $2T = 3000$ samples per second, for instance, the result from filtering it is a signal containing frequencies up to 750 Hz.

Very important is the fact that halfband filters reduce the resulting signal bandwidth by a factor of two. Thus, and according to Nyquist's theorem, its length can also be halved. As an example, if an M-sample long signal $s[\cdot]$ with frequencies up to 5000 Hz passes through a lowpass halfband digital filter, the resulting signal with frequencies in the range $0 \sim 2500$ Hz may be subsampled, or downsampled, by a factor of two, being therefore represented by means of

$\frac{M}{2}$ equally-spaced samples only. Usually, filtering may occur either in time- or frequency-domain. The latter requires a conversion from time- to frequency-domain, followed by multiplications with the filter frequency response and, lastly, another conversion to bring back the signal from frequency- to time-domain, where both the transformations are based on FTs. Since that is a computationally-expensive procedure, we usually adopt an equivalent free-of-conversion strategy: discrete convolution, which is denoted by an asterisk, i.e., "*".

Detailed explanations on convolution and digital filters are outside the objective of this tutorial, however, since they are essential for a better comprehension of DTWT, this paragraph is dedicated to briefly comment on them. Let $p[\cdot] = \{p_0, p_1, \ldots, p_{N-1}\}$ be the N-sample long lowpass halfband digital filter interacting with the M-sample long discrete-time input signal $s[\cdot]$. Then, $y[\cdot] = s[\cdot] * p[\cdot] = p[\cdot] * s[\cdot]$ where $y_n = \sum_k p_k s_{n-k}$ for $(0 \leqslant k \leqslant N-1)$, $(0 \leqslant n-k \leqslant M-1)$, and $(0 \leqslant n \leqslant M+N-2)$. A simple tip might be used to carry out the convolution: consider each sample of $s[\cdot]$ and $p[\cdot]$ as being one digit in a multi-digit multiplication scheme, as children do in primary schools. Then, multiply the numbers without carrying the overflow to the next position as in the following example for which $s[\cdot] = \{1, 2, 3\}$ and $p[\cdot] = \{4, 5\}$, resulting in $y[\cdot] = \{4, 13, 22, 15\}$:

	1	2	3	M-sample long signal
		4	5	N-sample long filter
	5	10	15	
4	8	12	+	
4	13	22	15	resulting signal with M+N-1 samples

Indeed, discrete-time convolution produces a signal with $M+N-1$ samples. In practice, however, the $N-1$ additional samples are discarded. Particularly, in the case of DTWTs, we still have a downsampling by two, as mentioned and as detailed ahead. In case we downsample $y[\cdot]$ from our example, the resulting signal is $\{4, 22\}$, i.e., we take one sample each two.

Convolution-based filtering followed by downsampling, which constitutes the essence of DTWTs, is intelligently implemented by using Mallat's algorithm which consists of a simple matrix multiplication as follows:

$$\underbrace{\begin{pmatrix} h_0 & h_1 & h_2 & \cdots & \cdots & \cdots & h_{N-1} & 0 & 0 & 0 & 0 & \cdots & \cdots & 0 & 0 \\ g_0 & g_1 & g_2 & \cdots & \cdots & \cdots & g_{N-1} & 0 & 0 & 0 & 0 & \cdots & \cdots & 0 & 0 \\ 0 & 0 & h_0 & h_1 & h_2 & \cdots & \cdots & \cdots & h_{N-1} & 0 & 0 & \cdots & \cdots & 0 & 0 \\ 0 & 0 & g_0 & g_1 & g_2 & \cdots & \cdots & \cdots & g_{N-1} & 0 & 0 & \cdots & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdot & & & \cdot & & & \cdot & & & \cdot & & & & \cdot & \cdot \\ \cdot & & & \cdot & & & \cdot & & & \cdot & & & & \cdot & \cdot \\ \cdot & & & \cdot & & & \cdot & & & \cdot & & & & \cdot & \cdot \\ h_{N-2} & h_{N-1} & 0 & 0 & \cdots & \cdots & \cdots & 0 & 0 & h_0 & h_1 & \cdots & \cdots & h_{N-4} & h_{N-3} \\ g_{N-2} & g_{N-1} & 0 & 0 & \cdots & \cdots & \cdots & 0 & 0 & g_0 & g_1 & \cdots & \cdots & g_{N-4} & g_{N-3} \end{pmatrix}}_{MxM \text{ matrix} A_{[\cdot][\cdot]}} \cdot \underbrace{\begin{pmatrix} s_0 \\ s_1 \\ s_2 \\ s_3 \\ \cdots \\ \cdots \\ \cdot \\ \cdot \\ \cdot \\ s_{M-2} \\ s_{M-1} \end{pmatrix}}_{\text{input}(S_{[\cdot]})} = \underbrace{\begin{pmatrix} r_0 \\ r_1 \\ r_2 \\ \cdots \\ \cdots \\ \cdots \\ \cdot \\ \cdot \\ \cdot \\ r_{M-2} \\ r_{M-1} \end{pmatrix}}_{\text{output}(R_{[\cdot]})} .$$

Basically, three matrices are involved in the process: matrix $A_{[\cdot][\cdot]}$ which is formed by the analysis filters coefficients, matrix $S[\cdot]$ which contains the input signal and, lastly, matrix $R[\cdot]$ corresponding to the transformation output. As $A_{[\cdot][\cdot]}$ advances from the first pair of lines until the last, a shift to the right is required in such a way that $h_{[\cdot]}$ and $g_{[\cdot]}$ start to be written two positions ahead in each subsequent pair. Whenever some of the filters coefficients exceed the limits for the length of the lines, they continue to be written at the beginning of the same respective lines, being the exceeding positions of each line set to zero. This process is known as *wraparound*. This is just how the convolutions followed by the downsamplings are implemented.

Once $R[\cdot]$ is obtained, the resulting first-level DTWT corresponds to the concatenation of the sub-signal formed by the even indexes of $r_{[\cdot]}$, known as *trend* or *approximation* of length $\frac{M}{2}$, followed by that formed by the odd indexes of $r_{[\cdot]}$, called *fluctuation* or *detail* of length $\frac{M}{2}$. In consequence, $\text{DTWT}(s_{[\cdot]}) = \{s_0, s_2, s_4, s_{M-2}\ldots, s_1, s_3, s_5, \ldots, s_{M-1}\}$, with the same length of the input signal, i.e., M. That is the first decomposition level, i.e., $j = 1$. If the *fluctuation* sub-signal is kept intact and only the *trend* sub-signal is used as a new input to the same algorithm, then, two other sub-signals of length $\frac{M}{4}$ are obtained. Their concatenation, following the same process described above, generates one sub-signal of length $\frac{M}{2}$ replacing the original input of the same length. The new complete signal is an array of length M and corresponds to the second-level $\text{DTWT}(s_{[\cdot]})$, i.e., $j = 2$. The process can be repeated $log_2(M)$ times, i.e., until the length of the approximation sub-signal at the current level is turned to the unity, being this the reason why M is required to be a power of two. In other words, looking at the decomposition tree exemplified in Fig. 9, the DTWT of an input signal corresponds to the bottom-up left-to-right concatenation of the tree leaves.

Interestingly, as j advances, the approximation sub-signal used as input reduces its length to half, implying that, after a certain decomposition level, the dimension of the input becomes lesser than the filters support-size. If, for instance, $M = 16$ and $N = 4$, then, after the third-level transformation, the input subsignal to be used in the fourth level has length 2, i.e., the multiplication is impossible because matrix $A_{[\cdot][\cdot]}$ and the input are not compatible in terms of length. In that case, we are required to repeat the input so many times as necessary until it reaches the minimum required length in such a way that the matrix multiplication becomes possible, as detailed in [17].

The ideal level j depends on the temporal and spectral resolutions desired, which hold different criteria for ideality: the former is faultless whenever j and N are minimized at most, because of the specific part of $s[\cdot]$ captured by the

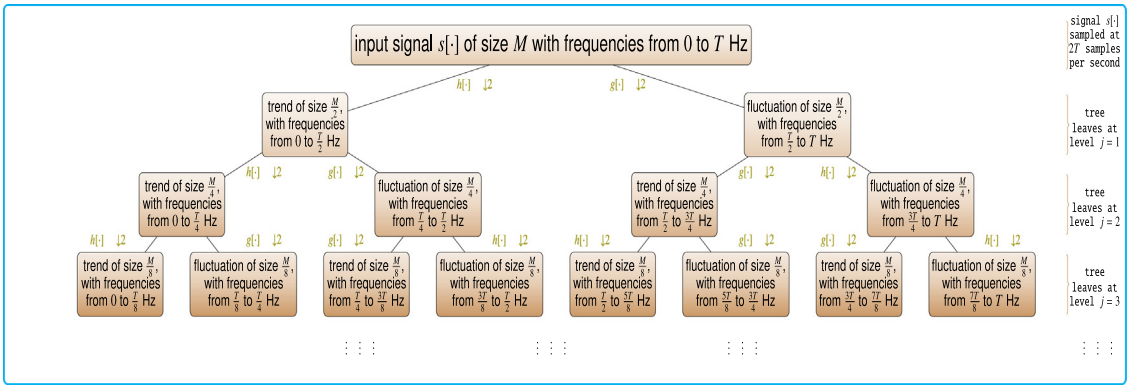


Fig. 10. The DTWPT structure.

filters. In opposition, the latter is excellent whenever j and N are maximized and the associated filters exhibit linear-phase and maximally-flat responses at their pass and stop-bands. Equilibrium can be obtained whenever the intermediary decomposition level is selected, i.e., j corresponds to the mean between the minimum, 1, and the maximum, $log_2(M)$, resulting in $j = \left\lfloor \frac{1+log_2(M)}{2} \right\rfloor$, where $\lfloor\cdot\rfloor$ is the floor operator. In such a case, the respective frequency resolution is $r = \frac{T}{2^j} = \frac{T}{2^{\lfloor(1+log_2(M))/2\rfloor}}$ Hz, where $2T$ is the sampling rate at which $s[\cdot]$ was digitalized and T is its maximum frequency content. For an equilibrated time and frequency accuracy, an intermediary value for N is also needed, for which 2 is the lower bound, there existing no theoretical upper bound. In practice, however, N around 20 or 30 are commonly used.

In addition to the regular decomposition tree inherent to the DTWT, which adopts both higher spectral resolution and lower temporal resolution for the lower frequencies and vice-versa, a variant known as Discrete-Time Wavelet-Packet Transform (DTWPT) provides a uniform and equally distributed time–frequency resolution for all spectrum, once j, N, and the filterbank are defined. DTWPT contains 2^j sub-band signals of size $\frac{M}{2^j}$ at the jth decomposition level, as shown in Fig. 10. Noteworthy in that Figure is the fact that DTWPT requires, for a correct spectral distribution known as natural frequency ordering (NFO), the alternating application of filters $h[\cdot]$ and $g[\cdot]$ in each decomposition level, as detailed in [18]-(pp.111) and mentioned in [19].

From now on, we discuss specific details involving the filterbank used to realize the DTWT and DTWPT, where two filter families exist: orthogonal and biorthogonal. The former family is defined in such a way that not only $h[\cdot]$ and $g[\cdot]$ are orthogonal but also their corresponding translations as disposed in the transformation matrix $A[\cdot][\cdot]$. Thus, every line of that matrix is orthogonal to each other, implying that the matrix itself is considered orthogonal. This has a very important consequence: the inverse of $A[\cdot][\cdot]$, i.e., $A^{-1}[\cdot][\cdot]$ is its transpose, i.e., $A^T[\cdot][\cdot]$, implying that to retrieve $S[\cdot]$ from $R[\cdot]$ we just need to compute $S[\cdot] = A^{-1}[\cdot][\cdot] \cdot R[\cdot] = A^T[\cdot][\cdot] \cdot R[\cdot]$, i.e.,

$$\underbrace{\begin{pmatrix} \bar{h}_0 & \bar{h}_1 & \bar{h}_2 & \dots & \dots & \dots & \bar{h}_{N-1} & 0 & 0 & 0 & 0 & \dots & \dots & 0 & 0 \\ \bar{g}_0 & \bar{g}_1 & \bar{g}_2 & \dots & \dots & \dots & \bar{g}_{N-1} & 0 & 0 & 0 & 0 & \dots & \dots & 0 & 0 \\ 0 & 0 & \bar{h}_0 & \bar{h}_1 & \bar{h}_2 & \dots & \dots & \dots & \bar{h}_{N-1} & 0 & 0 & \dots & \dots & 0 & 0 \\ 0 & 0 & \bar{g}_0 & \bar{g}_1 & \bar{g}_2 & \dots & \dots & \dots & \bar{g}_{N-1} & 0 & 0 & \dots & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \cdot & & & \cdot & & & \cdot & & & \cdot & & & \cdot & & \cdot \\ \cdot & & & \cdot & & & \cdot & & & \cdot & & & \cdot & & \cdot \\ \cdot & & & \cdot & & & \cdot & & & \cdot & & & \cdot & & \cdot \\ \bar{h}_{N-2} & \bar{h}_{N-1} & 0 & 0 & \dots & \dots & \dots & 0 & 0 & \bar{h}_0 & \bar{h}_1 & \dots & \dots & \bar{h}_{N-4} & \bar{h}_{N-3} \\ \bar{g}_{N-2} & \bar{g}_{N-1} & 0 & 0 & \dots & \dots & \dots & 0 & 0 & \bar{g}_0 & \bar{g}_1 & \dots & \dots & \bar{g}_{N-4} & \bar{g}_{N-3} \end{pmatrix}}_{MxM \text{ matrix} A_{[\cdot][\cdot]}} \cdot \underbrace{\begin{pmatrix} r_0 \\ r_1 \\ r_2 \\ r_3 \\ \dots \\ \dots \\ \cdot \\ \cdot \\ \cdot \\ r_{M-2} \\ r_{M-1} \end{pmatrix}}_{\text{input}(R_{[\cdot]})} \underbrace{\begin{pmatrix} s_0 \\ s_1 \\ s_2 \\ \dots \\ \dots \\ \cdot \\ \cdot \\ \cdot \\ s_{M-2} \\ s_{M-1} \end{pmatrix}}_{\text{output}(S_{[\cdot]})},$$

where $\bar{h}_k = h_{N-k-1}$, $\bar{g}_k = (-1)^{k+1}h_k$ and, in addition, $g_k = (-1)^k h_{N-k-1}$ in matrix $A[\cdot][\cdot]$. The set $\{h[\cdot], g[\cdot], \bar{h}[\cdot], \bar{g}[\cdot]\}$ is known as a quadrature mirror filter (QMF) bank, where the subsets $\{h[\cdot], g[\cdot]\}$ and $\{\bar{h}[\cdot], \bar{g}[\cdot]\}$ are known as analysis filter pair and synthesis filter pair, respectively.

Different from orthogonal filterbanks, those known as biorthogonal do not use an orthogonal transformation matrix. Thus, $A^{-1}[\cdot][\cdot] \neq A^T[\cdot][\cdot]$ and, consequently, the analysis and synthesis filter pairs are not directly related one to the other. Orthogonality is a condition used to build wavelet filters, implying that, once it is relaxed, originating biorthogonal filters, other important characteristics might replace that. Except for Haar wavelet system, no other orthogonal family of wavelets exhibits linear phase. Conversely, biorthogonal filters having a symmetric or anti-symmetric impulse response exhibit linear-phase.

3.5. Wavelet behind the scenes: connecting all the concepts related to filters, scaling functions, and wavelet functions

Usually, any piece of scientific work on wavelets refers to the following six elements:

- the wavelet function $\psi(t)$.
- the scaling function $\phi(t)$;
- the low-pass half-band analysis filter $h[\cdot]$;
- the high-pass half-band analysis filter $g[\cdot]$;
- the synthesis filter $\bar{h}[\cdot]$;
- the synthesis filter $\bar{g}[\cdot]$;

All of them have already been mentioned in this tutorial. As we discussed, $\psi(t)$ is used in the CWT as the sole function responsible for the corresponding transformation. Proceeding, both $\psi(t)$ and $\phi(t)$ are adopted in the DWT. Conversely, $h[\cdot]$ and $g[\cdot]$ are the essence of DTWT and DTWPT, which can be inverted based on $\bar{h}[\cdot]$ and $\bar{g}[\cdot]$. Thus, it seems that we have completely disconnected elements, however, we have not! The existing connection is the most important contribution we have on literature from Ingrid Daubechies [2].

Every time we invert a DTWT or a DTWPT, by using the synthesis filters $\bar{h}[\cdot]$ and $\bar{g}[\cdot]$, we are writing back the original signal as a linear combination of $\phi(t)$ and $\psi(t)$. This is, however, not obvious because explicit specification of the scaling and wavelet functions is not required to do so. The natural question now is "*since $h_{[\cdot]}$ and $g_{[\cdot]}$ are the only necessary elements to calculate the DTWT or the DTWPT* and, in addition, *$\bar{h}_{[\cdot]}$ and $\bar{g}_{[\cdot]}$ are the only elements we need to invert the process, why are the scaling and wavelet functions required?*" In order to answer it, we first note that most part of the applications based on the DTWT or DTWPT also requires the inversion: in speech, audio, and image compression, for instance, the transformation is performed, then some of the coefficients with low energy in the transformed signal are discarded or modified, and, lastly, the inverse transformation is perform to reconstruct the signal. Since the transformed signal is modified prior to the inversion, the reconstructed signal will be just similar, and not equal, to the original, i.e., the compression is a lossy operation. *This modification in the transformed signal prior to reconstruction is the point*. Specifically, *the more the transformed signal is disturbed prior to its reconstruction, the more the reconstructed signal assumes a **shape** corresponding to a combination of the **shapes** of the scaling and wavelets functions associated with the respective filters*. Thus, at least their shapes should be known. A more detailed explanation about this can be found in paper [20].

To obtain $\phi(t)$ and $\psi(t)$ associated with a certain filter bank, we can use either an analytical procedure or an iterated approach. The former appears frequently in literature and requires more math, as shown in paper [1] and in book [3]. The latter, however, requires less computation and, thus, will be described here, accompanied by a numerical example. When DTWTs or DTWPTs are performed, possibly based on Mallat's algorithm, downsamplings by 2 occur just after each convolution, as in Figs. 9 and 10. Consequently, the inversion requires not only the application of the synthesis filters but also upsamplings to restore the original signal. Since the reconstruction of the original signal is equivalent to writing it as the linear combination of $\phi(t)$ and $\psi(t)$, we hope to find those functions somewhere in the procedure. Indeed, we can easily see them!

Letting $s[\cdot]$ and $\mathcal{L}[\cdot]$ be the original signal under analysis and the transformed signal, respectively, we can restore the former from the latter as follows:

- BEGINNING
- STEP 1: $\mathcal{L}[\cdot]$ is upsampled by 2;
- STEP 2: $\bar{h}[\cdot]$ is convolved with the upsampled signal. This turns the leaf back from the jth to the $(j-1)$th level;
- REPEAT until $j = 1$:
  - BEGINNING
  - STEP 3: the resulting signal is upsampled by 2;
  - STEP 4: $\bar{h}[\cdot]$ is convolved with the upsampled signal, turning the leaf back from the $(j-1)$th to the $(j-2)$th level;
  - STEP 5: $j \leftarrow j-1$;
  - END;
- STEP 6: the resulting signal is upsampled by 2;
- STEP 7: $\bar{h}[\cdot]$ is convolved with the upsampled signal, reaching $s[\cdot]$, i.e., the tree root.
- END.

Since convolution is a commutative process, we can perform the procedure above starting from STEP 2 and, afterwards, apply STEP 1. This allows for the entire process, excluding STEP 1, to be generalized once $\bar{h}[\cdot]$ is defined. Notably, the multiple convolutions and upsamplings over $\bar{h}[\cdot]$ create a new "combined filter" which, obviously, does not depend on $\mathcal{L}[\cdot]$. The length of this "combined filter" is greater than that of $\bar{h}[\cdot]$, however, if the filter-bank has a particular property which characterizes it as a *wavelet filter*, then the borders of the "combined filter" vanish, forcing it to have a *compact support*, i.e., outside a certain small interval its coefficients will be equal to zero.

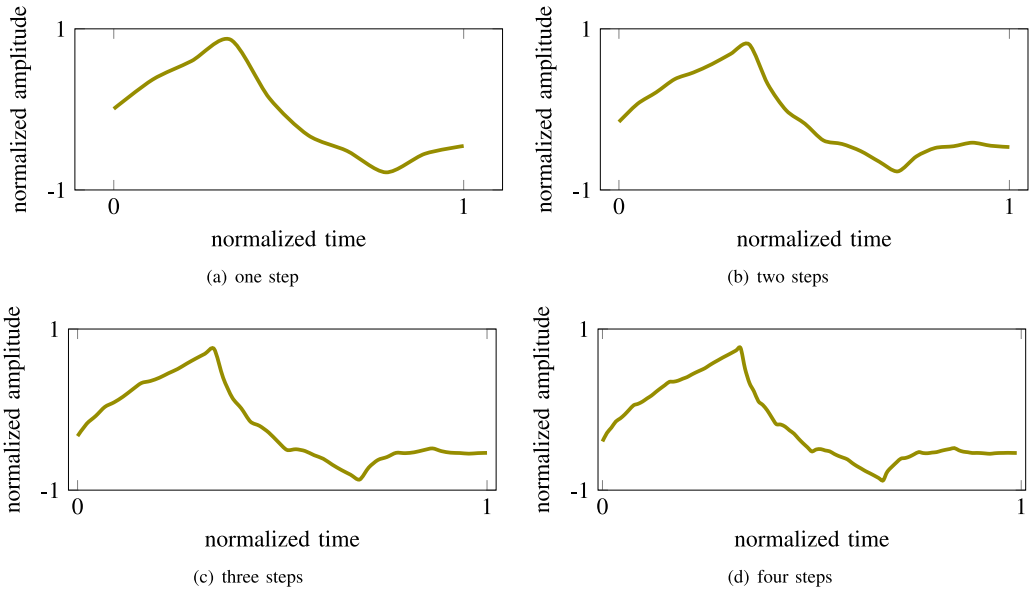


Fig. 11. Daubechies scaling function becomes evident as the process advances from (a) to (d). Particularly, not only the convergence of that function is notable. Its compact support, i.e., the tendency of its borders to stop fluctuating, also becomes clear.

As an example, let us use Daubechies filter $h[\cdot] = \{\frac{1+\sqrt{3}}{4\sqrt{2}}, \frac{3+\sqrt{3}}{4\sqrt{2}}, \frac{3-\sqrt{3}}{4\sqrt{2}}, \frac{1-\sqrt{3}}{4\sqrt{2}}\}$, with support-size 4, to carry out the procedure above, particularly starting from STEP 2 and repeating STEPS 3, 4, and 5 a few times to see the resulting "combined filter" as follows:

- $h[\cdot] = \{\frac{1+\sqrt{3}}{4\sqrt{2}}, \frac{3+\sqrt{3}}{4\sqrt{2}}, \frac{3-\sqrt{3}}{4\sqrt{2}}, \frac{1-\sqrt{3}}{4\sqrt{2}}\}$;
- $(h[\cdot])_{U2} = \{\frac{1+\sqrt{3}}{4\sqrt{2}}, 0, \frac{3+\sqrt{3}}{4\sqrt{2}}, 0, \frac{3-\sqrt{3}}{4\sqrt{2}}, 0, \frac{1-\sqrt{3}}{4\sqrt{2}}\}$;
- convolving $(h[\cdot])_{U2}$ with $h[\cdot]$ produces {0.1166284, 0.2020067, 0.2561341, 0.3186359, …, 0.0083736}, as shown in Fig. 11-(a);
- upsampling the resulting signal by 2 results in {0.1166284, 0, 0.2020067, 0, 0.2561341, 0, 0.3186359, 0, …, 0.0083736};
- convolving $h[\cdot]$ with the previous resulting signal produces {0.03982965, 0.06898710, 0.08747211, 0.10881703, …, −0.00076624}, as shown in Fig. 11-(b);
- upsampling the resulting signal by 2 results in {0.03982965, 0, 0.06898710, 0, 0.08747211, 0, 0.10881703, 0, …, −0.00076624};
- convolving $h[\cdot]$ with the previous resulting signal produces {0.013602183, 0.023559714, 0.029872514, 0.037161996, …, 0.000070117}, as shown in Fig. 11-(c);
- upsampling the resulting signal by 2 results in {0.013602183, 0, 0.023559714, 0, 0.029872514, 0, 0.037161996, 0, …, 0.000070117};
- convolving $h[\cdot]$ with the previous resulting signal produces {0.0046452679, 0.0080458545, 0.0102017325, 0.0126911560, …, −0.0000064162}, as shown in Fig. 11-(d);
- and so on...,

where $(h[\cdot])_{U2}$ corresponds to $h[\cdot]$ upsampled by 2. Equivalently, we have:

$$\underbrace{\Big(\underbrace{\Big(\underbrace{\Big(\underbrace{\big((h[\cdot])_{U2} * (h[\cdot])\big)_{U2}}_{\text{see Fig. 11-(a)}} * (h[\cdot])\Big)_{U2}}_{\text{see Fig. 11-(b)}} * (h[\cdot])\Big)_{U2}}_{\text{see Fig. 11-(c)}} * h[\cdot]\Big)}_{\text{see Fig. 11-(d)}} \quad \ldots \quad \ldots \quad \ldots \quad ,$$

where $*$ denotes the convolution. Notably, the convolved signals shown in Figs. 11-(a), (b), (c) and (d) approach the Daubechies scaling function for the filters with support-size 4. The more we advance, the more the resulting function shows a compact support, i.e., tends to be zero outside a short-time interval.

Readers might have observed that, in the convolutions used above to get the scaling function, only $h[\cdot]$ was used, never $g[\cdot]$. This is because, as we mentioned when studying the DWT and as shown in Fig. 7, the scaling function works as a low-pass filter used to cover the frequency 0 Hz. Accordingly, when using the DTWT, as in Fig. 9, the trend in the deepest tree level was the only leaf obtained based on $h[\cdot]$. All the other leaves used a combination of $h[\cdot]$ and $g[\cdot]$ to be calculated. Consequently, the scaling function corresponds to a combination of $h[\cdot]$ with the proper downsamples by 2.

Proceeding, if we repeat the previous process but starting from the high-pass filter $g[\cdot]$ instead of $h[\cdot]$, we will get:

- $g[\cdot] = \{\frac{1-\sqrt{3}}{4\sqrt{2}}, -\frac{3-\sqrt{3}}{4\sqrt{2}}, \frac{3+\sqrt{3}}{4\sqrt{2}}, -\frac{1+\sqrt{3}}{4\sqrt{2}}\}$;
- $(g[\cdot])_{U2} = \{\frac{1-\sqrt{3}}{4\sqrt{2}}, 0, -\frac{3-\sqrt{3}}{4\sqrt{2}}, 0, \frac{3+\sqrt{3}}{4\sqrt{2}}, 0, -\frac{1+\sqrt{3}}{4\sqrt{2}}\}$;
- convolving $(g[\cdot])_{U2}$ with $h[\cdot]$ results in {−0.031251, −0.054128, −0.068631, −0.085378, …, 0.031251}, as shown in Fig. 12-(a);
- upsampling the resulting signal by 2 results in {−0.031251, 0, −0.054128, 0, −0.068631, 0, −0.085378, 0, …, 0.031251};
- convolving $h[\cdot]$ with the previous resulting signal results in {−0.0106725, −0.0184853, −0.0234383, −0.0291577, …, −0.0028597}, as shown in Fig. 12-(b);
- upsampling the resulting signal by 2 results in {−0.0106725, 0, −0.0184853, 0, −0.0234383, 0, −0.0291577, 0, …, −0.0028597};
- convolving $h[\cdot]$ with the previous resulting signal results in {−0.00364475, −0.00631291, −0.00800443, −0.00995767, …, 0.00026168}, as shown in Fig. 12-(c);
- upsampling the resulting signal by 2 results in {−0.00364475, 0, −0.00631291, 0, −0.00800443, 0, −0.00995767, 0, …, 0.00026168};
- convolving $h[\cdot]$ with the previous resulting signal results in {−0.001244715, −0.002155913, −0.002733590, −0.003400641, …, −0.000023946}, as shown in Fig. 12-(d);
- and so on...

Equivalently, we have:

$$\Bigg(\Bigg(\Bigg(\Big(\underbrace{\big(\underbrace{(g[\cdot])_{U2} * (h[\cdot])}_{\text{see Fig. 12-(a)}}\big)_{U2} * (h[\cdot])}_{\text{see Fig. 12-(b)}}\Big)_{U2} * (h[\cdot])\Bigg)_{U2} * h[\cdot]\Bigg) \quad \ldots \quad \ldots \quad \ldots \quad .$$

see Fig. 12-(c)

see Fig. 12-(d)

As in the previous case, the convolved signals shown in Figs. 12-(a), (b), (c) and (d) tend to the Daubechies wavelet function for the filters with support-size 4. The more we advance, the more the resulting function tends to have a compact support, i.e., tends to be zero outside a short time interval.

This time, readers might have observed that, in the convolution we have just used to obtain the wavelet function, $g[\cdot]$ was used *once* and, then, $h[\cdot]$ was adopted repeatedly. This is because, according to Fig. 7, the wavelet function works as a band-pass filter. Accordingly, if the DTWT is the case, as in Fig. 9, all the leaves but the trend in the deepest tree level were obtained based on a combination of $g[\cdot]$, applied *once*, with $h[\cdot]$, applied one or more times according to the leaf level.

Thus, readers can now connect all the concepts, understanding the relationship between the filters coefficients and the scaling and wavelet functions!

4. Illustrative state-of-the-art applications

In this section, a review on state-of-the-art applications of wavelets in physics is provided. For four of them, descriptions enriched with suggestions on how to move research forward are provided, whereas for the others, just summarized comments and references were included.

4.1. Astrophysics: wavelet-based analysis of acoustic patterns from mars

Recently, National American Space Agency (NASA) has attracted researchers' attention by showing data from Mars [21] in an attempt to let us know whether life once flourished on the red planet or not. Particularly curious are the patterns of sounds we can download and hear directly from the agency website [22]. Thus, to show one interesting application of wavelets, the analysis of a 10-second recording made on March 2, 2021, during the 12th sol, i.e., Martian day, by Perseverance Rover, is provided in this subsection. The original signal, which consists of the sounds obtained from 30 impacts of a laser beam on a rock, was sampled at 48 000 samples per second and is 527 610-sample long. Initially, that signal passed through an amplitude normalization step and, then, a simple energy-based procedure [23] was used

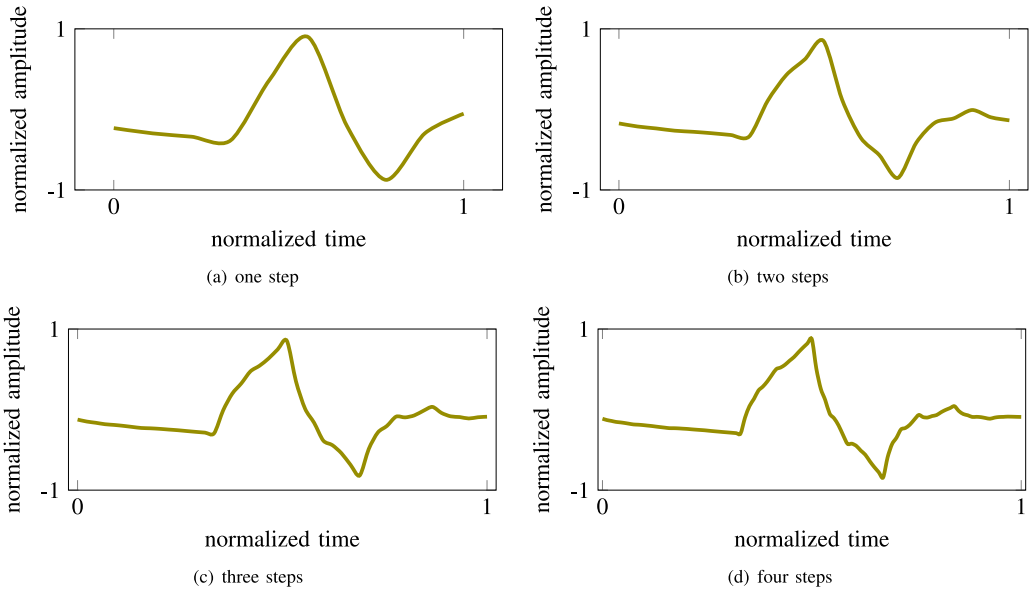


Fig. 12. Daubechies wavelet function becomes evident as the process advances from (a) to (d). Particularly, not only the convergence of that function is notable. Its compact support, i.e., the tendency of its borders to vanish, also becomes clear.

together with a hard threshold to identify the regions where each one of the impacts took place. Proceeding, a 256-sample long rectangular window was placed over each identified region, originating 30 short signals hereafter referred to as $s_i[\cdot]$, where $(0 \leqslant i \leqslant 29)$. Just to illustrate, one of those signals is shown in Fig. 13-(a). All the other ones are similar to that.

To allow for a balanced time–frequency resolution during the analyses of the short signals, the medium decomposition level was used for each one, i.e., $\frac{\text{maximum level}}{2} = \frac{\log_2(256)}{2} = \frac{8}{2} = 4$. Additionally, to prevent relevant phase distortions, Coiflet filters with support-size 24 were chosen. Then, the DTWPT-transformed signals, modified as detailed in paper [19] in order to better express the energy concentrations in each sub-band, were obtained. To illustrate, the fourth-level DTWT of the short signal in Fig. 13-(a) is shown in Fig. 13-(b). In general, we can easily note the time localization provided by the DTWPT: the positions of the main spikes in the original signal match the positions of the main peaks inside each sub-band of the corresponding transformed signal. Furthermore, we can observe that the transformed signal in Fig. 13-(b) presents just a few relevant peaks. Thus, an interesting and simple approach for studying the composition of Mars' rocks is to use both the index position and the normalized amplitude of the main peaks as being a feature vector for subsequent classification based on a supervised or unsupervised strategy [24], depending on the specific intention and information available to characterize each rock physically.

Additionally, as NASA scientists recently observed [25], the high-pitched sounds resulting from the laser impacts traveled 250 m per second, however, those for which lower frequencies are predominant, such as the ones coming from the whir of the Perseverance Rover helicopter rotor, traveled 10 m slower. Thus, based on the time–frequency properties of wavelets, the two categories of sounds from Mars can be easily distinguished just by using the index position and the amplitude of the main peaks in the DTWT-based transformed signals. Another possibility to advance the wavelet-based investigation of the Martian signals is to apply the Enhanced Teager Energy Operator (ETEO) [26] on the DTWT-based transformed acoustic signals to get features capable of describing precisely, as the squared product of the transformed signal amplitude by its instantaneous frequency, the energy contained in the original signal. Further details on how to implement such a procedure can be found in paper [19].

Another interesting aspect to consider while investigating acoustic patterns from Mars is the significant amount of data collected by the Perseverance Rover microphone. Since it increases considerably over time, a proper compression strategy is certainly useful to store millions of hours-long recordings. Consequently, noting that WTs play a significant role on audio compression [27], the following general procedure is suggested to encode those acoustic files: (i) the signal is converted to the DTWPT domain by using Coiflets or Symmlets filters with support-size around $16 \sim 24$, in order to minimize phase distortion and to balance time and frequency resolutions, considering an energy calculation procedure which approximately matches the masking properties of the human ear [28]; (ii) low-energy sub-bands of the transformed signal are discarded and a lossless subsequent step, possibly based on Huffman encoding [28], is performed, creating the encoded string; (iii) a reconstructed signal, acoustically transparent in comparison with its original version, can be retrieved from the encoded string by inverting the Huffman codification and the wavelet transformation.

Notably, efficient audio compression algorithms take advantage of the masking properties of the human ear, considering, for instance, the Bark scale [28], as exemplified in Table 1. Thus, the energy concentrations can be inspected

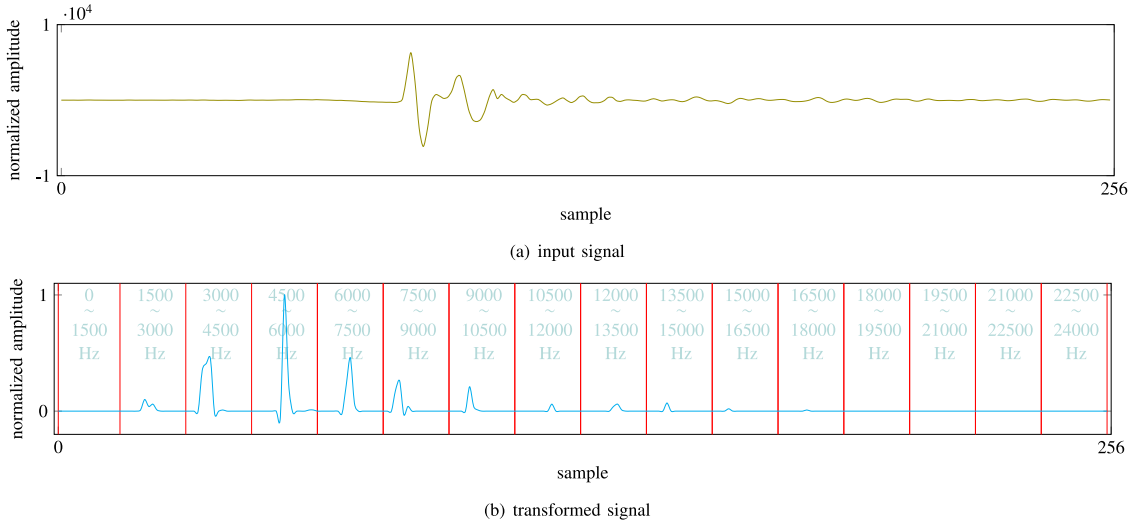


Fig. 13. (a): A 256-sample long segment from $s[\cdot]$, around an event; (b): the corresponding DTWPT in the fourth-level of decomposition, where the frequency resolution for each $\frac{256}{2^4} = 16$-sample long sub-band is a function of the maximum frequency of $s[\cdot]$ and the decomposition level, i.e., $\frac{(48000/2)}{4} = 1500$ Hz. The vertical red lines in the figure represent the transition frontiers between adjacent sub-bands, where their corresponding frequency ranges are indicated in light teal. Particularly, the dominant energy peak is in the range 4500 ∼ 6000 Hz, being localized close to the middle of the input signal, with relevant contributions also in the ranges 3000 ∼ 4500, 6000 ∼ 7500, 7500 ∼ 9000 and 9000 ∼ 10 500 Hz. The localization of the peaks within each subband in the transformed signal is directly and proportionally related to the localization of the corresponding spectral components within the original signal, as explained in paper [19].

according to the 25 Bark bands [28] obtained by using a proper wavelet decomposition tree. Assuming, as an example, 524 328-sample long audio segments, i.e., a length that is a power of two and is the closest to the original 527 610-sample long signal mentioned at the beginning of this subsection, and the sampling rate of 48 000 samples per second, an efficient representation of the Bark scale can be obtained, as shown in that Table In general, such a procedure can easily provide compression rates of about 15:1, which is certainly significant to store considerable amounts of acoustic data.

4.2. Neuroscience: wavelet-based spike sorting

Recent state-of-the-art approaches have considered not only time–frequency but also time–frequency–shape joint decompositions, such as those documented in papers [29–31], and [32], for investigation in neuroscience. One particularly interesting problem in such a research field is spike sorting. Based on that, we can, for instance, advance our understanding on the neural code [33].

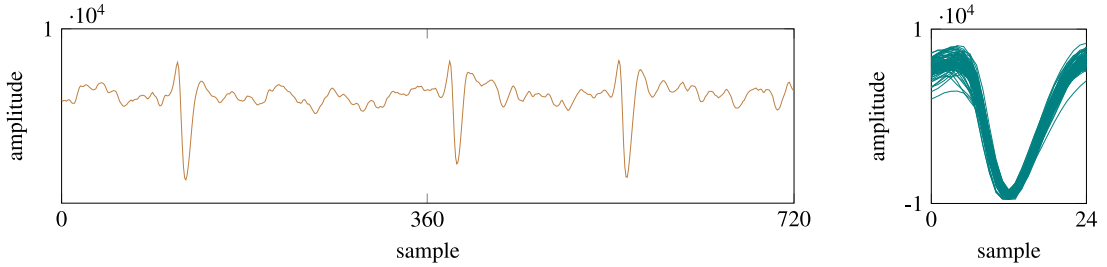
In the experiments reviewed in this subsection, we consider, among others, spikes coming from H1 [32,33], the motion-sensitive neuron of the fly's visual system, such as those shown in Fig. 14. They were acquired extracellularly by means of a specifically-designed hardware containing an iridium–platinum electrode, as detailed in [29] and as shown in Fig. 15-(a) to (d), while the fly views a pattern of random bars and complex images moving across its visual field. That is the general setup for a spike sorting apparatus, which was used in papers [29–31], and [32]. Particularly, in paper [29], a new technique to create a matched wavelet transform for spike sorting was presented. It consists of a modification in the original formulation used to obtain the well-known Daubechies' filters [1]. With that, the idea of wavelet-based joint time–frequency–shape analysis was established and successfully used to distinguish among seven specific spike shapes. Each shape was particularly correlated with one type of image displayed on the video monitor, as shown in Fig. 15-(d).

Proceeding, in paper [30], a new methodology for joint time–frequency–shape analysis based on wavelets, also considering a modification of Daubechies' filters, was proposed for spike sorting. Based on that strategy, the matching conditions were not significantly disturbed by the method used to solve the non-linear system of equations required to get the filters coefficients, as in the previous formulation [29]. Later, as shown in paper [31], the formulation in paper [30] was slightly modified in such a way that the two highest vanishing moments from Daubechies' filters were replaced by the matching conditions used to define the intended spike shape to be matched, bringing better pattern recognition accuracy. Subsequently, Daubechies maximally-flat equations were replaced by Symmlet equations in paper [32], implying that the matched wavelets obtained for spike sorting exhibit almost-linear phase responses, minimally distorting the filtered signals. Those matched wavelet filters in papers [30,31], and [32] were called the first (DST-I), second (DST-II), and third (DST-III) generations of Shapelets, respectively.

The main ideas adopted in papers [29–31], and [32], that are more recent, are not the only possibilities of using wavelets for spike sorting. As we can see in paper [35], for instance, a pyramidal algorithm playing the role of a bank of quadrature

Table 1
The 25 sets of samples from the maximum-level, i.e., $\log_2(524328) = 19$th level DTWPT are used to approximately represent the Bark scale. At that level and considering the original signal sampling rate of 48 000 samples per second, each sample of the transformed signal has a resolution of $\frac{(48\,000/2)}{2^{19}} = \frac{48\,000}{2^{19+1}} = 0.045776$ Hz. The symbol $\lfloor\cdot\rfloor$ in the fourth column of the table represents a rounding floor operation. It is required because a sample is obviously always an integer number. Due to that rounding, the frequency range we obtained from the wavelet decomposition tree is an approximation to the Bark scale, however, it does not make any significant difference in practice.

Bark band [28]	Exact band range (Hz)	Initial WPT sample	Final WPT sample	Approximate range in Bark band (Hz)
0	0∼100	0	$\lfloor\frac{100}{0.045776}\rfloor = 2184$	0∼99.9748
1	100∼200	2185	$\lfloor\frac{200}{0.045776}\rfloor = 4369$	100.0206∼199.9953
2	200∼300	4370	$\lfloor\frac{300}{0.045776}\rfloor = 6553$	200.0411∼299.9701
3	300∼400	6554	$\lfloor\frac{400}{0.045776}\rfloor = 8738$	300.0159∼399.9907
4	400∼510	8739	$\lfloor\frac{510}{0.045776}\rfloor = 11\,141$	400.0365∼509.9904
5	510∼630	11 142	$\lfloor\frac{630}{0.045776}\rfloor = 13\,762$	510.0362∼629.9693
6	630∼770	13 763	$\lfloor\frac{770}{0.045776}\rfloor = 16\,821$	630.0151∼769.9981
7	770∼920	16 822	$\lfloor\frac{920}{0.045776}\rfloor = 20\,097$	770.0439∼919.9603
8	920∼1080	20 098	$\lfloor\frac{1080}{0.045776}\rfloor = 23\,593$	920.006∼1079.9932
9	1080∼1270	23 594	$\lfloor\frac{1270}{0.045776}\rfloor = 27\,743$	1080.0389∼1269.9636
10	1270∼1480	27 744	$\lfloor\frac{1480}{0.045776}\rfloor = 32\,331$	1270.0093∼1479.9839
11	1480∼1720	32 332	$\lfloor\frac{1720}{0.045776}\rfloor = 37\,574$	1480.0296∼1719.9874
12	1720∼2000	37 575	$\lfloor\frac{2000}{0.045776}\rfloor = 43\,691$	1720.0332∼199.9992
13	2000∼2320	43 692	$\lfloor\frac{2320}{0.045776}\rfloor = 50\,681$	2000.0450∼2319.9735
14	2320∼2700	50 682	$\lfloor\frac{2700}{0.045776}\rfloor = 58\,982$	2320.0192∼2699.9600
15	2700∼3150	58 983	$\lfloor\frac{3150}{0.045776}\rfloor = 68\,813$	2700.0058∼3149.9839
16	3150∼3700	68 814	$\lfloor\frac{3700}{0.045776}\rfloor = 80\,828$	3150.0297∼3699.9825
17	3700∼4400	80 829	$\lfloor\frac{4400}{0.045776}\rfloor = 96\,120$	3700.0283∼4399.9891
18	4400∼5300	96 121	$\lfloor\frac{5300}{0.045776}\rfloor = 115\,781$	4400.0349∼5299.9911
19	5300∼6400	115 782	$\lfloor\frac{6400}{0.045776}\rfloor = 139\,811$	5300.0368∼6399.9883
20	6400∼7700	139 812	$\lfloor\frac{7700}{0.045776}\rfloor = 168\,210$	6400.0341∼7699.9810
21	7700∼9500	168 211	$\lfloor\frac{9500}{0.045776}\rfloor = 207\,532$	7700.0267∼9499.9848
22	9500∼12 000	207 533	$\lfloor\frac{12\,000}{0.045776}\rfloor = 262\,146$	9500.0306∼11 999.9953
23	12 000∼15 500	262 147	$\lfloor\frac{15\,500}{0.045776}\rfloor = 338\,605$	12 000.0411∼15 499.9825
24	15 500∼24 000	338 606	$\lfloor\frac{24\,000}{0.045776}\rfloor = 524\,328$	15 500.0283∼24 000.00

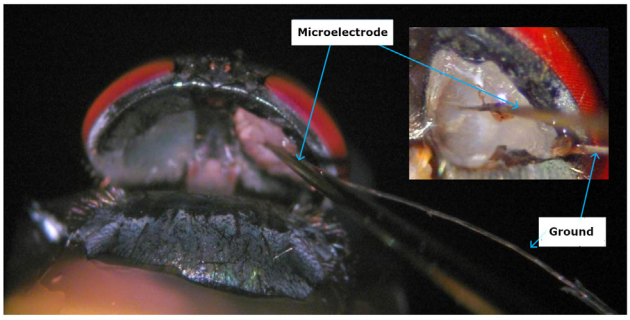


(a) A long signal, sampled at 44100 samples per second, 16-bit, captured with the specifically-designed hardware and the electrode shown in Figure 15-(b). Three spikes can be observed, corresponding to the moments that H1 fired.

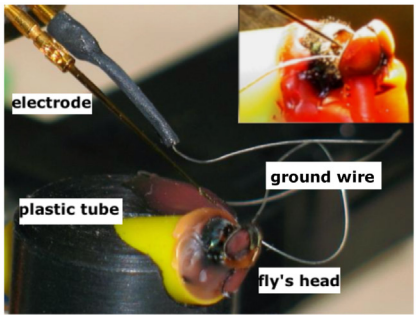
(b) Typical spikes coming from H1. All the curves are 25-sample long, corresponding to a period of approximately $\frac{25}{44100} \approx 667\mu s$.

Fig. 14. (a): A long signal coming from H1; (b): superimposed typical spikes generated by H1.

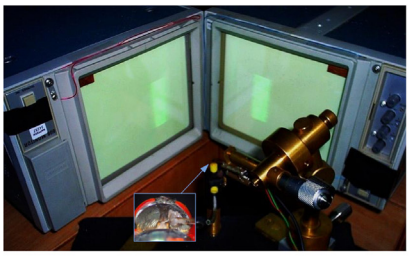
mirror filters was adopted to successfully identify patterns in an artificial spike train, providing better accuracy than principal component analysis (PCA) and reduced feature set (RFS) strategies. By using wavelet-packets, the authors of paper [36] propose a novel method for sorting recorded neural spikes based on best basis algorithm, Shannon information cost function, and local discriminant analysis. Notably, the authors' proposed approach showed to be superior both in removing noise and in identifying overlapped spikes. Similarly, the authors of paper [37] applied that idea to analyze signals from up to five neurons concomitantly.



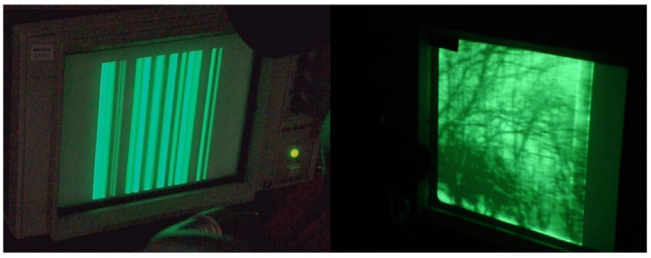
(a) The fly's head, magnified on the right, where the microelectrode used to get the signals coming H1 together with the ground cable appear.



(b) The fly, glued inside a yellow plastic tube, as magnified on the right, looking at a video screen where random bars and complex images appear. Adapted from the paper [32].



(c) Another view of the video screen. The fly is magnified on the bottom.



(d) sample bars and complex image displayed on the video screen for the fly to view.

Fig. 15. Four pictures, reproduced from the PhD thesis cited in [34] and authored by L. O. B. Almeida, related to the wavelet-based experiment with the fly.

In paper [38], the authors introduced a new technique for spike sorting combining wavelets with superparamagnetic clustering, allowing for automatic classification of data without assumptions such as low variance or Gaussian distributions, outperforming conventional techniques. The authors of paper [39] used wavelets to denoise and sort spikes, that were subsequently classified by using a Support Vector Machine (SVM), increasing the amount of information extractable from neural signals recorded with intraneural electrodes. Specific wavelet coefficients were adopted by the authors of paper [40] to create a new spike alignment technique based on multi-peak energy comparison and a dynamic codebook-based template-matching algorithm with a class-merging feature that is capable of working in real-time mode, outperforming the previous results.

An unsupervised feature selection method based on kernel density estimation was used by the authors of paper [41] to select wavelet coefficients with bimodal or multimodal distributions for spike sorting. Their technique was experimented with a simulated spike dataset, implying that the average misclassification rate after fuzzy C-means clustering was considerably reduced in comparison with previous approaches. Proceeding, Haar wavelets were used for spike sorting in an attempt to get a low-cost solution in paper [42]. The authors observed that, tested at different signal-to-noise ratios by using synthesized datasets with different spike shapes extracted from real neural recordings, an improvement of 3 to 10% was obtained in terms of accuracy in comparison with previous methods. To better remove interspike interference, the authors of paper [43] used a wavelet-packet based decomposition considering mutual information. They obtained a value of accuracy up to 99.76% on a dataset with significant noise level, outperforming the baseline by up to 22.35%.

Interestingly, by using maximum projection as the matching criterion, the authors of paper [44] designed a wavelet-based spike sorting algorithm which presented better clustering performance than classical Daubechies wavelets even with strong noise. Similarly and in order to avoid the overlapping phenomenon, a spike sorting method based on the wavelet packets and the mutual information was proposed by the authors of paper [45]. Performed on different public datasets, their tests demonstrate a higher clustering accuracy compared with the state-of-the-art techniques. Lastly, the authors of paper [46] presented a spike sorting strategy based on an optimized selection of parameters associated with the CWT. Tests with a simulated dataset and two public benchmark datasets outperformed both the WaveClus spike sorting method and the traditional PCA-based technique, demonstrating that improvements over neural decoding performance are feasible.

Summarizing, a significant number of spike sorting algorithms are based on wavelets, which are used for feature extraction. Some of the techniques adopt matched wavelets specially created to identify a pattern of interest. Contrary to this, other strategies make use of traditional wavelet families, taking advantage of their properties for pattern recognition. The perspective is that future research on spike sorting will focus on wavelets working together with recent artificial intelligence algorithms, enhancing the possibilities we have for the comprehension of the neural code. Particularly, new

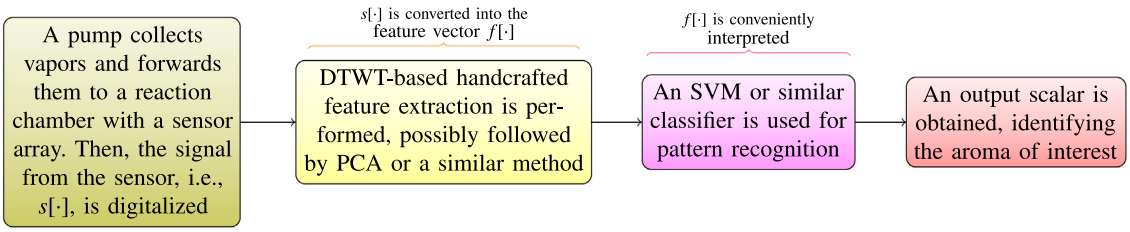


Fig. 16. General setup for most of the DTWT-based electronic nose systems.

matched wavelets could be designed to allow for more refined correlations between visual or related stimuli and the neurons' firing rates and behavior.

4.3. Artificial sensory: wavelet-based pattern recognition from electronic nose signals

Wavelets have also been used to conduct experiments on electronic noses, a research field that has received a significant attention but has advanced modestly, offering an interesting opportunity for new experiments. Possible starting points for them can be found in the papers mentioned hereafter in this subsection. Particularly, the authors of paper [47] presented a wavelet-based methodology to remove background aroma of mice themselves when detecting wound infection based on electronic nose, where fifteen gas sensors send data to a wavelet- and RBF-based classification strategy. Similarly, in paper [48], the authors also presented a method to classify electronic nose data from wound infection based on an SVM, used for classification, and on a wavelet analysis, adopted for feature extraction, where particle swarm optimization (PSO) was used to achieve optimization in the classifier.

The authors of paper [49] explored the potential of an electronic nose as a monitoring tool in clinical microbiology. They investigated, through two case studies and chemoresistive sensors, bacterial collections and urinary tract infection-suspected cases. A wavelet-based neural network and the concept of the fusion of multiple classifiers are the basis of their technique, which was successfully implemented and tested. Proceeding, the authors of paper [50] designed an electronic nose strategy, by using a metal oxide-based gas sensor array that identifies the odors of three kinds of herbal drinks. DTWT and PCA were applied for feature extraction and dimensionality reduction, respectively, providing a functional system with almost full accuracy.

The authors of paper [51], observing that food product safety is one of the most important fields for the application of electronic noses, proposed a method based on a fuzzy-wavelet neural network model not only to classify beef samples in their respective quality classes but also to predict their associated microbiological population directly from volatile compounds fingerprints. Similarly, in paper [52], the authors used a wavelet-based electronic nose algorithm to classify the spoilage of beef fillets stored either aerobically or under modified atmosphere packaging. Their system correctly classified 67 out of 70 aerobic samples, and 66 out of 71 samples in a modified temperature scenario, being thus a valuable resource for microbiology researchers.

The distinction of cayenne pepper from green cayenne pepper was the main objective of the authors of paper [53]. By using methane gas as the main element to be detected, they adopted wavelets for feature extraction associated with SVMs and a Linear Discriminant Analyser for classification. Their algorithm was capable of reaching a value of accuracy of 93%, considering hundreds of tests. Another wavelet-based food-related experiment was presented by the authors of paper [54], who created an electronic nose to detect meat spoilage micro-organisms during aerobic and modified atmosphere storage. By using the wavelet features as input, they adopted a fuzzy classifier and concluded that the proposed method could be used as a reliable scheme for meat spoilage detection. Lastly, the authors of paper [55] focused on getting a stable feature representation of the gas sensors responses based on wavelets. They successfully identified four different aromas of whiskies by using four commercial gas sensors. The classification results based on the proposed wavelet-based features outperformed those based on other features, reaching a value of accuracy of 92% for dozens of tests.

In a general manner, all the electronic nose systems briefly described in this subsection adopt DTWTs for handcrafted feature extraction from the signals containing the gas sensor array data. Particularly, wavelet-based features such as normalized subband energies, entropies, and zero-crossing rates, for instance, are grouped together to compose a feature vector which directly serves as input to an SVM or similar classifier, possibly having its dimension reduced by using PCA, as in Fig. 16. Notably, Fig. 17 shows the general structure of the SVM commonly used to interpret the wavelet-based feature vectors in the electronic nose systems. In view of the considerations and papers reviewed in this section, it is possible to state that future research on wavelet-based electronic nose systems should focus on find prominent wavelet filters and associated features that provide best signal representation, in association with specific gas sensors and classifiers.

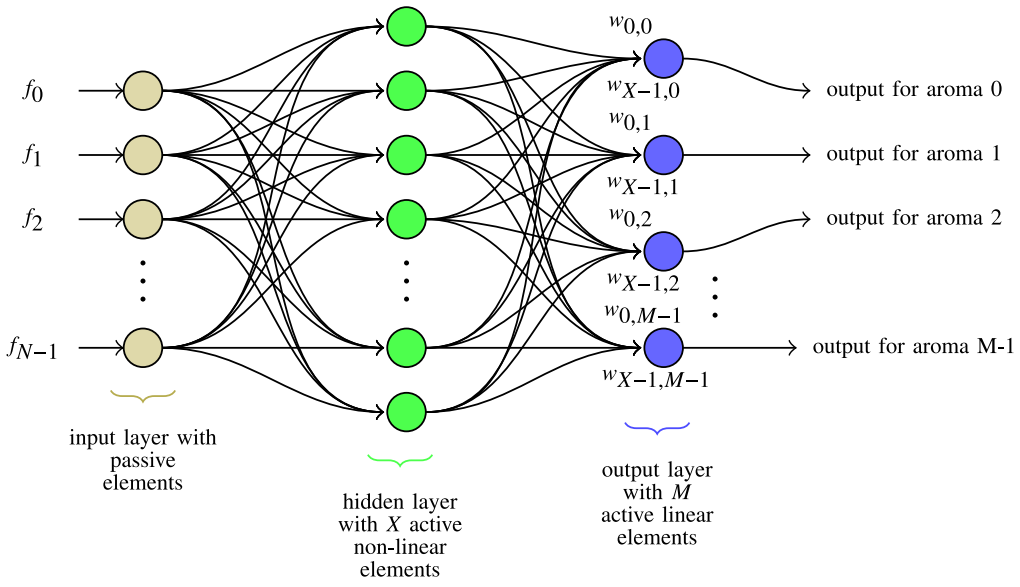


Fig. 17. The general structure of the multiclass SVM adopted in the electronic nose experiments. The weights determined during the supervised part of the training, based on the input features $f_0, f_1, f_2, \ldots, f_{N-1}$, are $\{w_{0,0}, \ldots, w_{X-1,0}\}, \{w_{0,1}, \ldots, w_{X-1,1}\}, \ldots, \ldots, \ldots, \{w_{0,M-1}, \ldots, w_{X-1,M-1}\}$. Specifically, N and X represent the feature vector length and the number of feature vectors used for training the SVM, respectively. Adopting X as being the number of training examples is a trick used to ease the training process, as explained in paper [56]. The non-linear activation functions for each element in the hidden layer can be, for instance, polynomial or Radial Basis (RBF), and the output elements linearly combine the outputs of the hidden layer with the weights found during the training stage. Notably, the M binary outputs are used as flags to indicate the input signal $s[\cdot]$ matches one particular aroma of interest, among M aromas.

4.4. Statistical physics: wavelet-based medical and biological image denoising

Different from the topics reviewed in the previous subsections, a long research record on wavelet-based image denoising exists. Essentially, wavelet-based image processing requires the DTWT and its inverse to be defined for bidimensional data. A number of books and scientific papers are dedicated to present that definition in detail. A must-read and internationally recognized book on general image processing techniques, which also includes explanations on the bidimensional WT, is [57]. Many relevant ideas and explanations can be found there. The bidimensional WT complemented with general image denoising techniques is also the main subject of two important books: [58,59]. The former includes a review on various existing algorithms and Bayesian approaches that produce high signal-to-noise ratio and less visual artifacts than other methods. In the latter, a focus on medical image processing is also provided, where the major challenging tasks taken up consist of the application of WTs for denoising. Complementarily, easy-to-follow explanations on how to calculate the bidimensional WT can be found in the tutorial paper I wrote a few years ago [17] and, for this reason, there is no need to repeat that content here.

Let us move forward, listing some wavelet-based medical and biological image denoising research works. In paper [60], for instance, Quaternion Wavelet Transform, which combines DWT and Quaternion Fourier Transform (QFT), was applied to image denoising. The results, assessed with peak signal-to-noise ratio (PSNR), are used to discuss the individual performance of Haar, Biorthogonal, Daubechies, and Coiflets wavelet bases. Interestingly, the authors of paper [61] showed an image denoising scheme based on Quantum Wavelet Transform (QWT) by embedding a noisy image into the wavelet coefficients of the original image. Daubechies filters of support-size 4 and a quantum oracle were combined and then compared with some state-of-art denoising techniques, providing a better result in terms of different image quality indexes. A similar idea based solely on the regular Biorthogonal DTW appears in paper [62], where the authors adopted the X-map denoising technique, providing visual results comparable to state-of-the-art ones in literature. Similarly, wavelets are combined with Gaussian Mixture Models (GMMs) in paper [63] to denoise low-light images, where the tests based on a set of natural low-luminous level images exhibit relevant values of PSNR and Structural Similarity Index (SSI).

The authors of paper [64] presented a wavelet- and autoencoder-based method for image compressive sensing dedicated to Magnetic Resonance Imaging (MRI), allowing for such an image to be reconstructed by using only a few under-sampled data. The experimental comparisons involving different sampling trajectories and ratios confirmed the efficacy of the authors' proposed technique. Another interesting approach appears in paper [65], where a wavelet-based medical image denoising strategy is presented. An optimal thresholding function is applied to the detail wavelet-transformed coefficients, that are subsequently used to invert the transformation. The assessment based on PSNR, mean square error (MSE), and SSI revealed improvements of about 30 dB when Rician noise was considered.

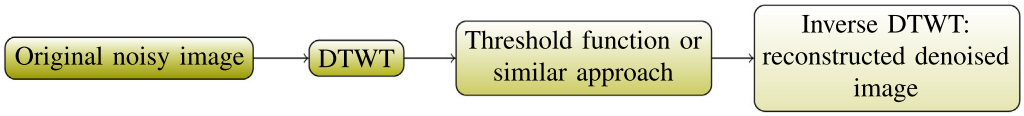


Fig. 18. The DTWT-based general procedure for image denoising.

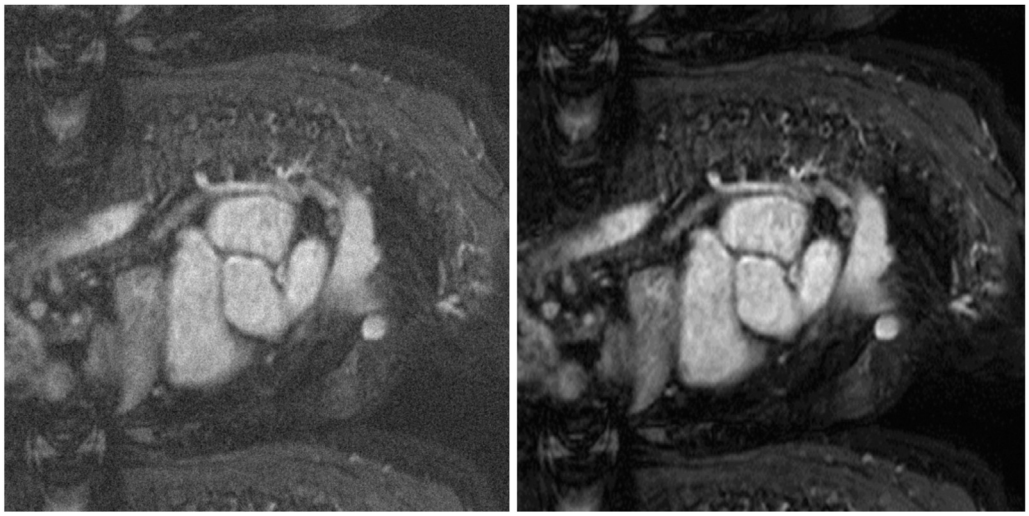


Fig. 19. [left]: An original MRI contaminated with noise; [right]: its DTWT-based denoised version. The visual quality of the latter image, where important details for medical interpretation are clear, is much better than that of the former one. Coiflets wavelets were used for denoising because of their low-phase distortion and nearly-flat frequency responses.

Wavelet-based filtration of super-resolved microscopic images is the problem treated in paper [66], where not only noise but also inhomogeneous background and reconstruction artifacts take place. Notably, the authors state that standard filtration techniques such as those based on Fourier Transform are not always appropriate because they may lower image resolution. The authors compare different wavelet filters and decomposition levels, successfully demonstrating the application of their algorithm to remove artifacts and undesirable background in images acquired by super-resolved structured illumination microscopy. Complementarily, wavelet-based image denoising is applied to COVID-19 detection in papers [67,68]. In the former, wavelets are used to enhance the quality of computerized tomography (CT) images based on a threshold function and PSO, with an adaptive behavior in accordance with the lung lobe and groundglass lesions. The authors' proposed approach increased the PSNR index by about 5 dB, accompanied by a reduction in the MSE. In the latter paper, the authors also consider CT images, presenting a comparison of different wavelet filters for image denoising, discussing peculiarities of COVID-19 images versus those filters.

Interestingly, the authors of paper [69] introduce the Complementary Color Wavelet Transform (CCWT), filling a gap between wavelets and color relationships and showing relevant applications in color filtering, texture retrieval, and image quality assessment with applications in medical image denoising. Complementarily, the authors of paper [70] show how to improve stereo endoscopic image quality based on wavelets, demonstrating the efficiency of their proposed enhancement methods in terms of perceived visual image quality.

Generally, all the papers mentioned in this subsection focus on DTWT-based image denoising by adopting a direct transformation, followed by a threshold function used to reduce the noise energy in specific sub-bands, ending with an inverse DTWT transformation to reconstruct the denoised image, as in Fig. 18. An example result from that procedure can be seen in Fig. 19. According to the papers reviewed and the respective discussions presented by their authors, it is clear that future research on medical image denoising, considering acquisitions coming from MRI and CT, is expected to be based on wavelets, used for optimized time–space representation, associated with deep learning algorithms for accurate classification. Particularly significant is the fact that wavelets, playing the role of a handcrafted strategy for feature representation, provide features with better physical meaning in comparison with those coming from learned features, such as deep autoencoders. Thus, this is, among others, one special reason for stating that wavelets have a long way forward in this field.

4.5. A brief summary on additional applications of wavelets

A variety of applications for wavelets can be found in literature, in addition to those involving the topics previously discussed. Thus, in this subsection, a condensed overview is presented to offer additional inspiration for readers interested

in using wavelets. First, relevant books are mentioned and, subsequently, a few selected scientific papers spanning the state-of-the-art are the focus of attention.

Published decades ago, book [71] presents important comparisons, assessments, methods, and application in a diversity of subfields involving wavelets. Soon after, book [72] introduces the underlying theory of wavelets by showing a wide range of applications in physics, signal processing, and communications. The basic principles of wavelets with general applications in pattern recognition are provided by the authors of book [73]. The same holds true in relation to book [74], where intuitive ideas, instead of abstract equations, are emphasized with focus on techniques used in commercial software apps.

An interesting connection between regular pure analyses and applied strategies, including measurement theory and harmonic investigation, are provided in book [75]. Accordingly, a very large scale integration (VLSI) design perspective related to wavelets is presented in book [76], particularly for the implementation of signal compression algorithms and the development of motion-compensated temporal filtering strategies for use in physics. Proceeding, book [77] contains unified overviews of the theory and applications of wavelets, considering both an algebraic point-of-view and a signal processing approach. A more focused discussion on signal processing-related topics is provided in book [78], highlighting the applications of wavelets for sparse signal representation, and the construction of orthogonal bases with fast algorithms. Signal compression, noise reduction, and inverse problems are the most common concepts discussed.

Proceeding, a theoretical approach together with historical details on wavelets is reported in book [79], from which readers might acquire intuition on how wavelets were designed. This subject is well complemented by the monograph [80]. Importantly, book [81] is mainly concerned with the stationary wavelet transform (SWT), which basically overcomes the lack of translation-invariance property of the DTWT. Notably, the authors of book [82] discuss how to enable the development of both representative and useful models of the superabundance of physical processes that surround us by using wavelets.

A theoretical development of wavelets based on a clear mathematical path, including the intermediate steps of derivation, is provided in book [83]. Current developments in the theory and applications of WTs and in the field of time–frequency signal analysis are provided in book [84], emphasizing fruitful directions for future advanced research. Speech processing, machine vision, information engineering, high-definition television, and telecommunication problems based on wavelets are explored in book [85] and, in complement, a linear algebra approach for understanding wavelets is the focus of the book [86]. Proceeding, efficiency in terms of computation, storage, and signal-to-noise ratio in wavelet-based reconstructed signals are explored in book [87]. The same can be found in book [88], where the focus is on signal compression. Extensions of multiresolution analysis and Newlands's harmonic wavelets accompanied by carefully selected worked examples, with special emphasis given to those topics in which students and professionals typically experience difficulty, can be found in book [89]. Significant applications of WTs in geoscience and biology have been elucidated in book [90], including the treatment of electroencephalogram signals, biometric recognition, and dimensionality reduction of the gait identification framework.

In book [91], lifting methods illustrated through examples, combined with computer algorithms, are presented. Accordingly, biology and geosciences applications of WTs are reported in book [92]. Following, in book [93] wavelets are applied to functional data analysis, offering a set of problems in which they can be adopted, including tumor analysis, functional magnetic resonance, and meteorological data. Speech enhancement, noise suppression, spectral analysis of speech signal, speech quality assessment, speech recognition, forensics by speech, and emotion recognition from speech, all based on wavelets, are presented in book [94]. Monogenic wavelet transforms (MWTs) with extension to multispectral signals, playing the role of a new multiscale analysis tool for geometric feature extraction, are presented in book [95]. Similarly, a diversity of applications of wavelets, smoothly explained, are presented in book [96]. Following, an extensive survey spanning new ideas to harmonic analysts, partial differential equation researchers, signal processing engineers, and other applied professionals are presented in book [97].

Lastly, a few additional papers describing additional applications of wavelets are presented hereafter. The authors of paper [98] proposed a novel decomposition strategy for instantaneous power quality indices (PQIs) monitoring in single- and three-phase systems considering inter-harmonics and transient disturbances based on wavelet-packets, showing significant advances for finding transitory changes in power systems. A double wavelet denoising method is presented by the authors of paper [99], allowing for more details related to the original signals under analysis to be preserved, for signals with a low signal-to-noise ratio and frequency overlaps. A wavelet-based method for feature extraction is used in paper [100] to diagnose piston seal wear and subsequent internal leakage from a double acting seal combination, adopted in the support oil cylinder of a mobile crane, where the corresponding algorithm effectively detects and classifies the piston seal wear with remarkable accuracy.

In paper [101], a new method for time–frequency analysis called the Entropy-optimized Paul Wavelet Transform (EoPWT) is shown and applied to direct hydrocarbon detection. As we can see in paper [102], the authors discuss about a unified wavelet-based attack framework against random delays, providing an effective solution to conquer random delayed countermeasures. Following, the authors of paper [103] presented a technique to determine whether it is possible to detect a voluntary gait 'starting' and 'stopping' intention, before it takes place, based on wavelets. A detection scheme also based on wavelet analysis was adopted to distinguish between the primary communication channel and a channel created from a modulation-based technique, as presented in paper [104], where the authors reported a significant advance on the state-of-the-art. Finally, the analysis of textures coming from microscopic images is the topic of paper [105], where

the authors demonstrated that the differentiation between oil-impregnated paper samples and different levels of thermal degradation using wavelet features is feasible and advantageous over related techniques.

From the books and papers briefly reviewed in this subsection, two essential approaches can be identified depending on the specific application: (i) only the direct wavelet transformation is applied to the signal under analysis and then some properties of that transformed signal, such as the energy, the entropy, and so on, are taken into account to characterize the feature vector which serves as input for a specific classifier; (ii) the direct wavelet transformation is applied to the signal under analysis, then the transformed signal is modified following a particular procedure, and lastly the transformation is inverted to synthesize an approximation of the original signal. On one hand, since the former approach does not make use of the inverse wavelet transformation, the wavelet family and filter support-size adopted can be chosen based just on the properties related to the filter frequency and phase responses. On the other hand, the shapes of the scaling and wavelet functions must also be taken into account for the latter approach because, as explained at the end of the previous section, the more the transformed signal is modified prior to the reconstruction, the more the shapes of those functions will influence and become evident in the reconstructed signal. Therefore, this is a possible and important criterion for choosing the best wavelet filter for a particular application.

5. Conclusions

Based on this tutorial, readers might have learned relevant concepts on wavelets, from the classical CWT to the elaborated DTWPT. Particularly, the connection between filter banks and the scaling and wavelet functions has been shown in detail, together with examples where wavelets are used to inspect the time–frequency properties of specific signals. Therefore, based on the concepts we have learned and on the history of applications where wavelets play a relevant role in physics and related fields, wavelets have an outstanding record of contributions and, certainly, a long way forward.

The author of this review paper can pleasantly provide, by request to *guido@ieee.org*, C/C++ source-code that implements both the DTWT and DTWPT, in one and two dimensions.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The author gratefully acknowledges the grants provided by the Brazilian agencies "National Council for Scientific and Technological Development (CNPq)", Brazil and "The State of São Paulo Research Foundation (FAPESP)", Brazil, respectively through the processes 306808/2018-8 and 2021/12407-4, in support of this research.

References

[1] J.R. Williams, K. Amaratunga, Introduction to wavelets in engineering, Internat. J. Numer. Methods Engrg. 37 (1994) 2365–2388.
[2] I. Daubechies, Ten lectures on wavelets, Bull. Amer. Math. Soc. (N.S.) 28 (1993) 350–360.
[3] G. Strang, T. Nguyen, Wavelets and Filter Banks, second ed., Wellesley-Cambridge Press, 1996.
[4] D. Sundararajan, Discrete Wavelet Transform: A Signal Processing Approach, Wiley, 2015.
[5] S.R. Devasahayam, Signals and Systems in Biomedical Engineering: Signal Processing and Physiological Systems Modeling, Kluwer Academic Press, 2000.
[6] P. Addison, J. Walker, R.C. Guido, Time-frequency analysis of biosignals, IEEE Eng. Med. Biol. Mag. 28 (5) (2009) 14–29.
[7] P. Addison, The Illustrated Wavelet Transform Handbook: Introductory Theory and Applications in Science, Engineering, Medicine and Finance, second ed., CRC Press, 2016.
[8] M.H. Farouk, Application of Wavelets in Speech Processing, second ed., Springer, 2017.
[9] R.X. Gao, R. Yan, Wavelets: Theory and Applications for Manufacturing, Springer, 2011.
[10] T.K. Sarkar, M. Salazar-Palma, M.C. Wicks, Wavelet Applications in Engineering Electromagnetics, Artech House, 2002.
[11] R.C. Guido, P. Addison, J. Walker, Time-frequency analysis of biosignals, IEEE Eng. Biol. Med. Mag. 28 (5) (2009) 14–29.
[12] C. Gargour, et al., A short introduction to wavelets and their applications, IEEE Circuits Syst. Mag. 9 (2) (2009) 57–68.
[13] H.P. Hsu, Schaum's Outline of Signals and Systems, fourth ed., McGraw-Hill Education, 2019.
[14] R.G. Lyons, Understanding Digital Signal Processing, third ed., Prentice Hall, 2010.
[15] A.V. Oppenheim, R.W. Schafer, Discrete-Time Signal Processing, third ed., Pearson Education, 2014.
[16] S. Haykin, B.V. Veen, Signals and Systems, second ed., Wiley, 2002.
[17] R.C. Guido, Practical and useful tips on discrete wavelet transforms, IEEE Signal Process. Mag. 32 (3) (2015) 162–166.
[18] A. Jensen, A.L. Cour-Harbo, Ripples in Mathematics: The Discrete Wavelet Transform, Springer, New York, 2000.
[19] R.C. Guido, Effectively interpreting discrete wavelet transformed signals, IEEE Signal Process. Mag. 34 (3) (2017) 89–100.
[20] R.C. Guido, A note on a practical relationship between filters coefficients and the scaling and wavelet functions of the discrete wavelet transform, Appl. Math. Lett. 24 (7) (2011) 1257–1259.
[21] https://mars.nasa.gov (accessed on May 2021).
[22] https://mars.nasa.gov/mars2020/participate/sounds/ (accessed on May 2021).
[23] R.C. Guido, A tutorial on signal energy and its applications, Neurocomputing 179 (2016) 264–282.
[24] C.M. Bishop, Pattern Recognition and Machine Learning, first ed., Springer, 2006.
[25] S. Maurice, et al., In situ recording of Mars soundscape, Nature (2022) http://dx.doi.org/10.1038/s41586-022-04679-0.
[26] R.C. Guido, Enhancing teager energy operator based on a novel and appealing concept: signal mass, J. Franklin Inst. B 356 (4) (2019) 2346–2352.

[27] M.J. Mendenhall, Wavelet-Based Audio Embedding and Audio/Video Compression, Biblioscholar, 2012.
[28] M. Bosi, R.E. Goldberg, Introduction to Digital Audio Coding and Standards, Springer, 2012, (reprint).
[29] R.C. Guido, et al., A new technique to construct a wavelet transform matching a specified signal with applications to digital, real-time, spike and overlap pattern recognition, Digit. Signal Process. 16 (1) (2006) 24–44.
[30] R.C. Guido, et al., Introduction to the Discrete Shapelet Transform and a New Paradigm: joint time-frequency-shape analysis, in: Proc. IEEE International Symposium on Circuits and Systems (IEEE ISCAS 2008), Seattle, WA, USA, 2008, pp. 2893–2896.
[31] R.C. Guido, Fusing time, frequency and shape-related information: introduction to the discrete shapelet transform's second generation (DST-II), Inf. Fusion (41) (2018) 9–15.
[32] R.C. Guido, Nearly symmetric orthogonal wavelets for time-frequency-shape joint analysis: Introducing the discrete shapelet transform's third generation (DST-III) for nonlinear signal analysis, Commun. Non-Linear Sci. Numer. Simul. 97 (2020) 105685.
[33] F. Rieke, W. Bialek, D. Warland, Spikes: Exploring the Neural Code, MIT Press, 1999.
[34] L.O.B. Almeida, Integrated Real Time Computational Instrumentation for Experiments with the Optic Flow of the Fly (Ph.D. thesis), Institute of Physics of São Carlos (IFSC), University of São Paulo (USP), Brazil, 2010, Available online (in Portuguese language) at https://teses.usp.br/teses/disponiveis/76/76132/tde-19032013-155627/en.php.
[35] J.C. Letelier, P.P. Weber, Spike sorting based on discrete wavelet transform coefficients, J. Neurosci. Methods 101 (2) (2000) 93–106.
[36] E. Hulata, et al., Detection and sorting of neural spikes using wavelet packets, Phys. Rev. Lett. 85 (21) (2000) 4637–4640.
[37] E. Hulata, et al., A method for spike sorting and detection based on wavelet packets and Shannon's mutual information, J. Neurosci. Methods 117 (1) (2002) 1–12.
[38] R.Q. Quiroga, et al., Unsupervised spike detection and sorting with wavelets and superparamagnetic clustering, Neural Comput. 16 (8) (2004) 1661–1687.
[39] L. Citi, et al., On the use of wavelet denoising and spike sorting techniques to process electroneurographic signals recorded using intraneural electrodes, J. Neurosci. Methods 172 (2) (2008) 294–302.
[40] H.L. Chan, et al., Unsupervised wavelet-based spike sorting with dynamic codebook searching and replenishment, Neurocomputing 73 (7–9) (2010) 1513–1527.
[41] X.L. Geng, G.S. Hu, Unsupervised feature selection by kernel density estimation in wavelet-based spike sorting, Biomed. Signal Process. Control 7 (2) (2012) 112–117.
[42] Y.N. Yang, A.J. Mason, Frequency band separability feature extraction method with weighted haar wavelet implementation for implantable spike sorting, IEEE Trans. Neural Syst. Rehabil. Eng. 25 (6) (2017) 530–538.
[43] Y. Chen, et al., HASS: High accuracy spike sorting with wavelet package decomposition and mutual information, in: IEEE International Conference on Bioinformatics and Biomedicine, Madri, Spain, 2018, pp. 831–838.
[44] L.S. Gao, et al., Neuronal spike sorting based on matched wavelet, in: 5th International Conference on Cloud Computing and Intelligence Systems, Nanjing, China, 2018, pp. 53–57.
[45] L. Huang, et al., WMsorting: Wavelet packets' decomposition and mutual information-based spike sorting method, IEEE Trans. Nanobiosci. 18 (3) (2019) 283–295.
[46] A. Soleymankhani, V. Shalchyan, A new spike sorting algorithm based on continuous wavelet transform and investigating its effect on improving neural decoding accuracy, Neuroscience 468 (2021) 139–148.
[47] J.W. Feng, A background elimination method based on wavelet transform in wound infection detection by electronic nose, Sensors Actuators B 157 (2) (2011) 395–400.
[48] Q.H. He, et al., Classification of electronic nose data in wound infection detection based on PSO-SVM classifier combined with wavelet transform, Intell. Autom. Soft Comput. 18 (7) (2012) 967–979.
[49] V.S. Kodogiannis, Point-of-care diagnosis of bacterial pathogens in vitro, utilising an electronic nose and wavelet neural networks, Neural Comput. Appl. 25 (2) (2014) 353–366.
[50] D.K. Agustika, K. Triyana, Application of principal component analysis and discrete wavelet transform in electronic nose for herbal drinks classification, in: International Conference on Science and Technology (ICST), Yogyakarta, Indonesia, 2016, 170012.
[51] V.S. Kodogiannis, A. Alshejari, A fuzzy-wavelet neural network model for the detection of meat spoilage using an electronic nose, in: IEEE International Fuzzy Systems Conference (IEEE FUZZ), Vancouver, Canada, 2016, pp. 710–717.
[52] V. Kodogiannis, Application of an electronic nose coupled with fuzzy-wavelet network for the detection of meat spoilage, Food Bioprocess Technol. 10 (4) (2017) 730–749.
[53] S.I. Sabilla, R. Sarno, Development of wavelet transforms to predict methane in chili using the electronic nose, in: IEEE International Conference on Advanced Mechatronics, Intelligent Manufacture, and Industrial Automation (IEEE ICAMIMIA), Surabaya, Indonesia, 2017, pp. 271–276.
[54] V.S. Kodogiannis, A rapid detection of meat spoilage using an electronic nose and fuzzy-wavelet systems, in: SAI Annual Conference on Areas of Intelligent Systems and Artificial Intelligence and their Applications to the Real World (IntelliSys), Vol. 15, London, England, 2018, pp. 521–539.
[55] T.P. Liu, et al., A multiscale wavelet kernel regularization-based feature extraction method for electronic nose, IEEE Trans. Syst. Man Cybern.-Syst. (2022) 1–12, http://dx.doi.org/10.1109/TSMC.2022.3151761.
[56] J.P.L. Escola, et al., Automated acoustic detection of a cicadid pest in coffee plantations, Comput. Electron. Agric. (169) (2020) 105215.
[57] R.C. Gonzalez, R.E. Woods, Digital Image Processing, third ed., Pearson, 2015.
[58] D. Cho, Image Denoising using Wavelet Transforms, VDM Verlag Dr. Mueller E.K, 2008.
[59] A. Khare, Wavelet Transform Based Techniques for Denoising of Medical Images, LAP Lambert Academic Publishing, 2010.
[60] A.K. Umam, M. Yunus, Quaternion wavelet transform for image denoising, in: International Conference on Mathematics - Pure, Applied and Computation (ICoMPAC), in: Journal of Physics Conference Series, vol. 974, Surabaya, Indonesia, 2018, 012006.
[61] S. Chakraborty, An image denoising technique using quantum wavelet transform, Internat. J. Theoret. Phys. 59 (11) (2020) 3348–3371.
[62] H.L. Dong, X-ray image denoising based on wavelet transform and median filter, Appl. Math. Non-Linear Sci. 5 (2) (2020) 435–442.
[63] S. Kannoth, H.C.S. Kumar, K.B. Raja, Denoising of low light images using patch priors and wavelets, Eng. Lett. 29 (3) (2021).
[64] S.Y. Wang, et al., Denoising auto-encoding priors in undecimated wavelet domain for MR image reconstruction, Neurocomputing 437 (2021) 325–338.
[65] N.E. Benhassine, A. Boukaache, D. Boudjehem, Medical image denoising using optimal thresholding of wavelet coefficients with selection of the best decomposition level and mother wavelet, Int. J. Image Syst. Technol. 31 (4) (2021) 1906–1920.
[66] M. Capek, et al., The wavelet-based denoising of images in Fiji, with example applications in structured illumination microscopy, Image Anal. Stereol. 40 (1) (2021) 3–16.
[67] M.A. Gungor, A comparative study on wavelet denoising for high noisy CT images of COVID-19 disease, Optik 235 (2021) 166652.
[68] G.W. Wang, et al., Asymptomatic COVID-19 CT image denoising method based on wavelet transform combined with improved PSO, Biomed. Signal Process. Control 76 (2022) 103707.
[69] Y. Chen, Complementary color wavelet: A novel tool for the color image/video analysis and processing, IEEE Trans. Circuits Syst. Video Technol. 29 (1) (2019) 12–27.

[70] B. Sdiri, et al., Efficient enhancement of stereo endoscopic images based on joint wavelet decomposition and binocular combination, IEEE Trans. Med. Imaging 38 (1) (2019) 33–45.
[71] A.K. Louis, et al., Wavelets: Theory and Applications, Wiley, 1997.
[72] R.M. Rao, A.S. Bopardikar, Wavelet Transforms: Introduction to Theory and Applications, Prentice Hall, 1998.
[73] J. Liu, et al., Wavelet Theory and its Application to Pattern Recognition, World Scientific Publishing Company, 2000.
[74] S. Qian, Introduction to Time Frequency and Wavelet Transforms, Prentice Hall PTR, 2001.
[75] D. Hong, et al., Real Analysis with an Introduction to Wavelets and Applications, Academic Press, 2004.
[76] L.-G. Che, VLSI Design of Wavelet Transform: Analysis, Architecture, and Design Examples, Imperial College Press, 2006.
[77] L.S. Burrus, et al., Introduction to Wavelets and Wavelet Transforms: A Primer, Prentice Hall, 2007.
[78] S. Mallat, G. Peyre, A Wavelet Tour of Signal Processing: The Sparse Way, third ed., Academic Press, 2008.
[79] B.B. Hubbard, The World According to Wavelets: The Story of a Mathematical Technique in the Making, second ed., A K Peters, 2008.
[80] R.S. Pathak, The Wavelet Transform (Atlantis Studies in Mathematics for Engineering and Science Book 4), Atlantis Press, 2009.
[81] L.M. Surhone, et al., Stationary Wavelet Transform, Betascript Publishing, 2010.
[82] A.N. Akansu, M.J.T. Smith, Subband and Wavelet Transforms: Design and Applications, Springer, 2011.
[83] S.V. Narasimhan, Introduction to Wavelet Transform: A Signal Processing Approach, Alpha Science International, 2011.
[84] L. Debnath, Wavelet Transforms and Time-Frequency Signal Analysis, Birkhauser, 2012.
[85] P.R. Haddad, Multiresolution Signal Decomposition: Transforms, Subbands, and Wavelets, Academic Press, 2012.
[86] M.W. Frazier, An Introduction to Wavelets through Linear Algebra, Springer, 2013.
[87] K.K. Shukla, et al., Efficient Algorithms for Discrete Wavelet Transform: With Applications to Denoising and Fuzzy Inference Systems, Springer, 2013.
[88] W.A. Pearlman, Wavelet Image Compression, Morgan & Claypool, 2013.
[89] L. Debnath, F.A. Shah, Wavelet Transforms and their Applications, second ed., Birkhauser, 2014.
[90] V. Nason, Principles and Applications of Wavelet Transform, NY Research Press, 2015.
[91] R.W. Goodman, Discrete Fourier and Wavelet Transforms: An Introduction Through Linear Algebra with Applications to Signal Processing, World Scientific Publishing Company, 2016.
[92] J. Winters, Wavelet Transforms and their Recent Applications in Biology and Geoscience, Scitus Academics Llc, 2016.
[93] P.A. Morettin, et al., Wavelets in Functional Data Analysis, Springer, 2017.
[94] M.H. Farouk, Application of Wavelets in Speech Processing, second ed., Springer, 2017.
[95] B.C. Thai, Monogenic Wavelet Transform: Extension to Multispectral Signal, LAP Lambert Academic Publishing, 2020.
[96] N. Bhatnagar, Introduction to Wavelet Transforms, Chapman and Hall/CRC, 2020.
[97] J.J. Benedetto, Wavelets: Mathematics and Applications, CRC Press, 2021.
[98] Y. Yu, et al., A two-stage wavelet decomposition method for instantaneous power quality indices estimation considering interharmonics and transient disturbances, IEEE Trans. Instrum. Meas. 70 (2021) 1–13, article 9001813.
[99] Y. Wu, et al., Improved wavelet denoising by non-convex sparse regularization under double wavelet domains, IEEE Access 7 (2019) 30659–30671.
[100] Y. Jin, et al., Fault diagnosis of hydraulic seal wear and internal leakage using wavelets and wavelet neural network, IEEE Trans. Instrum. Meas. 68 (4) (2019) 1026–1034.
[101] Y. Luan, et al., Seismic time-frequency analysis based on entropy-optimized paul wavelet transform, IEEE Geosci. Remote Sens. Lett. 17 (2) (2020) 342–346.
[102] F. Zhang, A systematic evaluation of wavelet-based attack framework on random delay countermeasures, IEEE Trans. Inf. Forensics Secur. 15 (2020) 1407–1422.
[103] S.M.S. Hasan, et al., Asynchronous prediction of human gait intention in a pseudo online paradigm using wavelet transform, IEEE Trans. Neural Syst. Rehabil. Eng. 28 (7) (2020) 1623–1635.
[104] W. Jang, W. Lee, Detecting wireless steganography with wavelet analysis, IEEE Wirel. Commun. Lett. 10 (2) (2021) 383–386.
[105] N.D. Jacob, et al., Assessment of power transformer paper ageing using wavelet texture analysis of microscopy images, IEEE Trans. Dielectr. Electr. Insul. 27 (6) (2020) 1898–1905.