Quantum spectral analysis: frequency in time

Mario Mastriani

Quantum spectral analysis: frequency at time

Mario Mastriani

Quantum Communications Group, Merxcomm LLC, 2875 NE 191 st, suite 801, Aventura, FL 33180, USA
mmastri@merxcomm.com

Abstract A *quantum time-dependent spectrum analysis*, or simply, *quantum spectral analysis* (QuSA) is presented in this work, and it's based on Schrödinger equation, which is a partial differential equation that describes how the quantum state of a non-relativistic physical system changes with time. In classic world is named *frequency at time* (FAT), which is presented here in opposition and as a complement of traditional spectral analysis frequency-dependent based on Fourier theory. Besides, FAT is a metric, which assesses the impact of the flanks of a signal on its frequency spectrum, which is not taken into account by Fourier theory and even less in real time. Even more, and unlike all derived tools from Fourier Theory (i.e., continuous, discrete, fast, short-time, fractional and quantum Fourier Transform, as well as, Gabor) FAT has the following advantages: a) compact support with excellent energy output treatment, b) low computational cost, $O(N)$ for signals and $O(N^2)$ for images, c) it doesn't have phase uncertainties (indeterminate phase for magnitude = 0) as Discrete and Fast Fourier Transform (DFT, FFT, respectively), d) among others. In fact, FAT constitutes one side of a triangle (which from now on is closed) and it consists of the original signal in time, spectral analysis based on Fourier Theory and FAT. Thus a toolbox is completed, which it is essential for all applications of Digital Signal Processing (DSP) and Digital Image Processing (DIP); and, even, in the latter, FAT allows edge detection (which is called flank detection in case of signals), denoising, despeckling, compression, and superresolution of still images. Such applications will be useful for signals, imagery and communication intelligence.

Keywords Digital Signal and Image Processing • Fourier Theory • Imagery Intelligence • Quantum Image Processing • Quantum Information Processing • Quantum Signal Processing • Schrödinger equation • Signals Intelligence • Spectral Analysis.

1 Introduction

The main concepts related to Quantum Information Processing (QuIn) may be grouped in the next topics: quantum bit (qubit, which is the elemental quantum information unity), Bloch's Sphere (geometric environment for qubit representation), Hilbert's Space (which generalizes the notion of Euclidean space), Schrödinger's Equation (which is a partial differential equation that describes how the quantum state of a physical system changes with time.), Unitary Operators, Quantum Circuits (in quantum information theory, a quantum circuit is a model for quantum computation in which a computation is a sequence of quantum gates, which are reversible transformations on a quantum mechanical analog of an *n*-bit register. This analogous structure is referred to as an *n*-qubit register.), Quantum Gates (in quantum computing and specifically the quantum circuit model of computation, a quantum gate or quantum logic gate is a basic quantum circuit operating on a small number of qubits), and Quantum Algorithms (in quantum computing, a quantum algorithm is an algorithm which runs on a realistic model of quantum computation, the most commonly used model being the quantum circuit model of computation) [1-4].

Nowadays, other concepts complement our knowledge about QuIn, the most important ones related to this work are:

Quantum Signal Processing (QuSP) - The main idea is to take a classical signal, sample it, quantify it (for example, between 0 and 2^8-1), use a classical-to-quantum interface, give an internal representation to that signal, make a processing to that quantum signal (denoising, compression, among others), measure the result, use a quantum-to-classical interface and subsequently detect the classical outcome signal. Interestingly, and as will be seen later, the quantum image processing has aroused more interest than QuSP. In the words of its creator: "many new classes of signal processing algorithms have been developed by emulating the behavior of physical systems. There are also many examples in the signal processing literature in which new classes of algorithms have been developed by artificially imposing physical constraints on implementations that are not inherently subject to these constraints". Therefore, Quantum Signal Processing (QuSP) is a signal processing framework [5, 6] that is aimed at developing new or modifying existing signal processing algorithms by borrowing from the principles of quantum mechanics and some of its interesting axioms and constraints. However, in contrast to such fields as quantum computing and quantum information theory, it does not inherently depend on the physics associated with quantum mechanics. Consequently, in developing the QuSP framework we are free to impose quantum mechanical constraints that we find useful and to avoid those that are not. This framework provides a unifying conceptual structure for a variety of traditional processing techniques and a precise mathematical setting for developing generalizations and extensions of algorithms, leading to a potentially useful paradigm for signal processing with applications in areas including frame theory, quantization and sampling methods, detection, parameter estimation, covariance shaping, and multiuser wireless communication systems." The truth is that to date, papers on this discipline are less than half a dozen, and its practical use is practically nil. Moreover, although it is an interesting idea, developed so far, does not withstand further comment.

Quantum Image Processing (QuIP) - it is a young discipline and it is in training now, however, it's much more developed than QuSP. QuIP starts in 1997. That year, Vlasov proposed a method of using quantum computation to recognize so-called *orthogonal images* [7]. Five years later, in 2002, Schutzhold described a quantum algorithm that searches specific patterns in binary images [8]. A year later, in October 2003, Beach, Lomont, and Cohen from Cybernet Systems Corporation, (an organization with a close cooperative relationship with the US Defense Department) demonstrated the possibility that quantum algorithms (such as Grover's algorithm) can be used in image processing. In that paper, they describe a method which uses a quantum algorithm to detect the posture of certain targets. Their study implies that quantum image processing may, in future, play a valuable role during wartime [9].

Later, we can found the works of Venegas-Andraca [10], where he proposes quantum image representations such as Qubit Lattice [11, 12]; in fact, this is the first doctoral thesis in the specialty, The history continues with the quantum image representation via the Real Ket [13] of Latorre Sentís, with a special interest in image compression in a quantum context. A new stage begins with the proposal of Le et al. [14], for a

flexible representation of quantum images to provide a representation for images on quantum computers in the form of a normalized state which captures information about colors and their corresponding positions in the images. History continues up to date by different authors and their innovative internal representation techniques of the image [15-36].

Very similar to the case of QuSP, the idea in back of QuIP is to take a classic image (captured by a digital camera or photon counter) and place it in a quantum machine through a classical-to-quantum interface (this process is known as preparation of qubits), give some internal representation to the image using the procedures mentioned above, perform processing on it (denoising, compression, among others), obtain the outcome via quantum measurement, which is the center of the quantum-to-classical interface, and ready. The contribution of a quantum machine over a classic machine when it comes to process images it is that the former has much more power of processing. This last advantage can handle images and algorithms of a high computational cost, which would be unmanageable in a classic machine in a practical sense, at least, this is the main idea for the future.

However, the problem of this discipline lies in its genetic, given that QuIP is the daughter of QuIn and DIP, thus, we fall into the old dilemma of teaching, i.e.: to teach latin to Peter, we should know more about Latin or more about Peter? The answer is simple: we should know very well of both, however in here the mission becomes impossible because exists a trade-off between DIP and QuIn, i.e., what is acceptable in QuIn, it is (at the same time) inadmissible in DIP. See more detail about this serious practical problem in [37].

The mentioned problem begins with the quantum measurement, then, if after quantum processing the quantum image within the quantum computer, we want to retrieve the result by tomography of quantum states, we will encounter a serious obstacle, this is:

> if we make a tomography of quantum states in QuIn (even, this can be extended to any method of quantum measurement after the tomography) with an error of 6% in our knowledge of the state, this constitutes an excellent measure of such state [38–41], but on the other hand, and this time from the standpoint of DIP [42-45], an error of 6% in each pixel of the outcome image constitutes a disaster, since this error becomes unmanageable and exaggerated noise. So over-whelming is the aforementioned disaster that the recovered image loses its visual intelligibility, i.e., its look and feel, and even up to its morphology, due to the destruction of edges and textures.

This speaks clearly (and for this purpose, one need only read the papers of QuIP cited above) that these works are based on computer simulations in classical machines, exclusively (in most cases in MATLAB® [46]), and they do not represent test in a laboratory of Quantum Physics. In fact, if these field trials were held, the result would be the aforementioned. We just have to go to the lab and try with a single pixel of an image, then extrapolate the results to the entire image and therefore the inconvenience will be explicit. On the other hand, today there are obvious difficulties to treat a full image inside a quantum machine, however, there is no difficulty for a single pixel, since that pixel represents a single qubit, and this can be tested in any laboratory in the world, without problems. Therefore, there are no excuses [37].

Definitely, the problem lies in the hostile relationship between the internal representation of the image (inside quantum machine), and the outcome measurement, the recovery of the image outside of quantum machine. Therefore, the only technique of QuIP that survives is QuBoIP [47]. This is because it works with CBS, exclusively, and the quantum measurement does not affect the value of states. However, it is important to clarify that both, i.e., traditional techniques QuIP and QuBoIP share a common enemy, and this is the decoherence [1, 47].

Quantum Fourier Transform (QuFT) - In quantum computing, the QuFT is a linear transformation on quantum bits and is the quantum analogue of the discrete Fourier transform. The QuFT is a part of many quantum algorithms, notably Shor's algorithm for factoring and computing the discrete logarithm, the quantum phase estimation algorithm for estimating the eigenvalues of a unitary operator, and algorithms for the hidden subgroup problem.

The QuFT can be performed efficiently on a quantum computer, with a particular decomposition into a product of simpler unitary matrices. Using a simple decomposition, the discrete Fourier transform can be implemented as a quantum circuit consisting of only $O(n^2)$ Hadamard gates and controlled phase shift gates, where n is the number of qubits [1]. This can be compared with the classical discrete Fourier transform, which takes $O(2n^2)$ gates (where n is the number of bits), which is exponentially more than $O(n^2)$. However, the quantum Fourier transform acts on a quantum state, whereas the classical Fourier transform acts on a vector, so not every task that uses the classical Fourier transform can take advantage of this exponential speedup. The best QuFT algorithms known today require only $O(n \log n)$ gates to achieve an efficient approximation [48].

Finally, this work is organized as follows: Fourier Theory are outlined in Section 2, where, we present the follow concepts: continuous, discrete, and fast Fourier transform. In Section 3, we show the proposed new spectral methods with its consequences. Section 4 provides conclusions and a proposal for future works.

2 Fourier's Theory

In this section, we discuss the tools, which are needs to understand the full extent to QuSA. These tools are: Continuous, Discrete (DFT), and Fast Fourier Transform (FFT). These tools are developed until the main concept of *uncertaintly principle*, which is fundamental to understand the theory behind QuSA/FAT.

Another transforms (which are members of Fourier Theory too) like Fractional Fourier Transform (FRFT), Short-Time Fourier Transform (STFT), and Gabor transform (GT), make a poor contribution in post to solve the problems of the Fourier Theory described in the abstract (i.e., the need for a time-dependent spectrum analysis), including (without doubt) to the wavelet transform in general and Haar basis in particular.

At the end of this section it should be clear: what is the ubiquity of QuSA in the context of a much larger, modern and full spectral analysis. On the other hand, this section will allow us to better understand the role QuSA as the origin of several tools used today in DSP, DIP, QuSP and QuIP. Finally, it will be clear why we say that QuSA completes a set of tools to date incomplete.

2.1 DFT and FFT

From all existing versions of the Fourier transform [49-56], that is to say, continuous-time, discrete, fractional, short-time (and a particular case of it due to Gabor), and quantum, not forgetting those versions based on the cosine [51-56] - in this section - we discuss the main characteristics of classical versions of Fourier transforms (including Gabor, and excluding cosine versions), their strengths and weaknesses, and as the two do not quite fill a gap in the field of spectral analysis, and in fact, any other tool it has done to date. Finally, and in a strict relation to this paper, the unique practice difference between DFT and FFT is their computational cost.

2.1.1 Fourier transform

The Fourier transform decomposes a function of time (a *signal*) into the frequencies that make it up, similarly to how a musical chord can be expressed as the amplitude (or loudness) of its constituent notes. The Fourier transform of a function of time itself is a complex-valued function of frequency, whose absolute value represents the amount of that frequency present in the original function, and whose complex argument is the phase offset of the basic sinusoid in that frequency. The Fourier transform is called the *frequency domain representation* of the original signal. The term *Fourier transform* refers to both the frequency domain representation and the mathematical operation that associates the frequency domain representation to a function of time. The Fourier transform is not limited to functions of time, but in order to have a unified language, the domain of the original function is commonly referred to as the *time domain*. For many functions of practical interest one can define an operation that reverses this: the *inverse Fourier transformation*, also called *Fourier synthesis*, of a frequency domain representation combines the contributions of all the different frequencies to recover the original function of time [49].

Linear operations performed in one domain (time or frequency) have corresponding operations in the other domain, which are sometimes easier to perform. The operation of differentiation in the time domain corresponds to multiplication by the frequency, so some differential equations are easier to analyze in the frequency domain. Also, convolution in the time domain corresponds to ordinary multiplication in the frequency domain. Concretely, this means that any linear time-invariant system, such as a filter applied to a signal, can be expressed relatively simply as an operation on frequencies. After performing the desired operations, transformation of the result can be made back to the time domain. Harmonic analysis is the systematic study of the relationship between the frequency and time domains, including the kinds of functions or operations that are "simpler" in one or the other, and has deep connections to almost all areas of modern mathematics [49].

Functions that are localized in the time domain have Fourier transforms that are spread out across the frequency domain and vice versa, a phenomenon known as the uncertainty principle. The critical case for this principle is the Gaussian function, of substantial importance in probability theory and statistics as well as in the study of physical phenomena exhibiting normal distribution (e.g., diffusion). The Fourier transform of a Gaussian function is another Gaussian function. Joseph Fourier introduced the transform in his study of heat transfer, where Gaussian functions appear as solutions of the heat equation [49].

There are several common conventions (see, [49]) for defining the Fourier transform $\hat{f}$ of an integrable function $f : \square \to \square$. In this paper, we will use the following definition:

$$\hat{f}(\xi) = \int_{-\infty}^{\infty} f(x) e^{-2\pi i x \xi} dx, \quad \text{for any real number } \xi. \tag{1}$$

When the independent variable x represents *time* (with SI unit of seconds), the transform variable ξ represents frequency (in hertz). Under suitable conditions f , is determined by $\hat{f}$ via the inverse transform:

$$f(x) = \int_{-\infty}^{\infty} \hat{f}(\xi) e^{2\pi i x \xi} d\xi, \quad \text{for any real number } x. \tag{2}$$

The statement that f can be reconstructed from $\hat{f}$ is known as the Fourier inversion theorem, and was first introduced in Fourier's *Analytical Theory of Heat*, although what would be considered a proof by modern standards was not given until much later. The functions f and $\hat{f}$ often are referred to as a *Fourier integral pair* or *Fourier transform pair* [49].

For other common conventions and notations, including using the angular frequency ω instead of the frequency ξ, see [49]. The Fourier transform on Euclidean space is treated separately, in which the variable x often represents position and ξ momentum.

Notes:

- In practice, the continuous-time version of the cosine transform is not used. Therefore, we will omit in this work.
- The properties of the Fourier transform will see in the next subsection, that is, for Discrete Fourier Transform (DFT), although only the most relevant in terms of this work.
- We will not develop here the two-dimensional version of the Fourier transform, if we instead for subsequent versions, using the property known as separability [51-56].
- Any extension on the Fourier transform shown in [49, 50].

2.1.2 DFT

In mathematics, the discrete Fourier transform (DFT) converts a finite list of equally spaced samples of a function into the list of coefficients of a finite combination of complex sinusoids, ordered by their frequencies, that has those same sample values. It can be said to convert the sampled function from its original domain (often time or position along a line) to the frequency domain [51].

The input samples are complex numbers (in practice, usually real numbers), and the output coefficients are complex as well. The frequencies of the output sinusoids are integer multiples of a fundamental frequency, whose corresponding period is the length of the sampling interval. The combination of sinusoids obtained through the DFT is therefore periodic with that same period. The DFT differs from the discrete-time Fourier transform (DTFT) in that its input and output sequences are both finite; it is therefore said to be the Fourier analysis of finite-domain (or periodic) discrete-time functions [51].

Since it deals with a finite amount of data, it can be implemented in computers by numerical algorithms or even dedicated hardware. These implementations usually employ efficient fast Fourier transform (FFT) algorithms; so much so that the terms "FFT" and "DFT" are often used interchangeably. Prior to its current usage, the "FFT" initialism may have also been used for the ambiguous term "finite Fourier transform" [51].

The sequence of *N* complex numbers $x_0, x_1, \ldots, x_{N-1}$ is transformed into an *N*-periodic sequence of complex numbers:

$$X_k \equiv \sum_{n=0}^{N-1} x_n . e^{-2\pi i k n / N}, \quad k \in \mathbf{Z} \ (\text{integers}) \tag{3}$$

Each X_k is a complex number that encodes both amplitude and phase of a sinusoidal component of function x_n . The sinusoid's frequency is *k* cycles per *N* samples. Its amplitude and phase are:

$$\begin{aligned} |X_k|/N &= \sqrt{\mathrm{Re}(X_k)^2 + \mathrm{Im}(X_k)^2}/N \\ \arg(X_k) &= \mathrm{atan2}\left(\mathrm{Im}(X_k), \mathrm{Re}(X_k)\right) = -i \ln\left(\frac{X_k}{|X_k|}\right), \end{aligned} \tag{4}$$

where atan2 is the two-argument form of the arctan function. Assuming periodicity (see Periodicity in [51]), the customary domain of *k* actually computed is [*0*, *N*-1]. That is always the case when the DFT is implemented via the Fast Fourier transform algorithm. But other common domains are [-*N*/2, *N*/2-1] (*N* even) and [-(*N*-1)/2, (*N*-1)/2] (*N* odd), as when the left and right halves of an FFT output sequence are swapped. Finally, from all its properties, the most important for this paper are the following [51-56]:

The unitary DFT - Another way of looking at the DFT is to note that in the above discussion, the DFT can be expressed as a Vandermonde matrix, introduced by Sylvester in 1867,

$$\mathrm{F} = \begin{bmatrix} \omega_N^{00} & \omega_N^{01} & \cdots & \omega_N^{0(N-1)} \\ \omega_N^{10} & \omega_N^{11} & \cdots & \omega_N^{1(N-1)} \\ \vdots & \vdots & \ddots & \vdots \\ \omega_N^{(N-1)0} & \omega_N^{(N-1)1} & \cdots & \omega_N^{(N-1)(N-1)} \end{bmatrix} \tag{5}$$

where

$$\omega_N = e^{-2\pi i / N} \tag{6}$$

is a primitive *Nth* root of unity called twiddle factor.

While for the case of discrete cosine transform (DCT), we have:

$$\omega_N = cos\left(2\pi/N\right) \tag{7}$$

The inverse transform is then given by the inverse of the above matrix,

$$\mathrm{F}^{-1} = \frac{1}{N}\mathrm{F}^{*} \tag{8}$$

For unitary normalization, we use a constant like $1/\sqrt{N}$, then, the DFT becomes a unitary transformation, defined by a unitary matrix:

$$\begin{aligned} &\mathrm{U} = \mathrm{F}/\sqrt{N} \\ &\mathrm{U}^{-1} = \mathrm{U}^{*} \\ &\left|\det\left(\mathrm{U}\right)\right| = 1 \end{aligned} \tag{9}$$

where *det*(•) is the *determinant function of* (•), and $(\bullet)^{*}$ means *conjugate transpose* of (•).

All this shows that the DFT is the product of a matrix by a vector, essentially, as follows:

$$\begin{bmatrix} X_0 \\ X_1 \\ \vdots \\ X_{N-1} \end{bmatrix} = \begin{bmatrix} \omega_N^{00} & \omega_N^{01} & \cdots & \omega_N^{0(N-1)} \\ \omega_N^{10} & \omega_N^{11} & \cdots & \omega_N^{1(N-1)} \\ \vdots & \vdots & \ddots & \vdots \\ \omega_N^{(N-1)0} & \omega_N^{(N-1)1} & \cdots & \omega_N^{(N-1)(N-1)} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_{N-1} \end{bmatrix} \tag{10}$$

No Compact Support – Based on Eq.(10), we can see that each element X_k of output vector results from multiplying the *kth* row of the matrix by the complete input vector, that is to say, each element X_k of output vector contains every element of the input vector. A direct consequence of this is that DFT scatters the energy to its output, in other words, DFT has a disastrous treatment of the output energy. Therefore, no compact support is equivalent to:

- DFT has a bad treatment of energy at the output
- DFT in not a time-varying transform, but frequency-varying transform

Time-domain vs frequency-domain measurements – As we can see in Fig. 1, thanks to DFT we have a new perspective regarding to signals measurement, i.e., the spectral viewing [55,56].

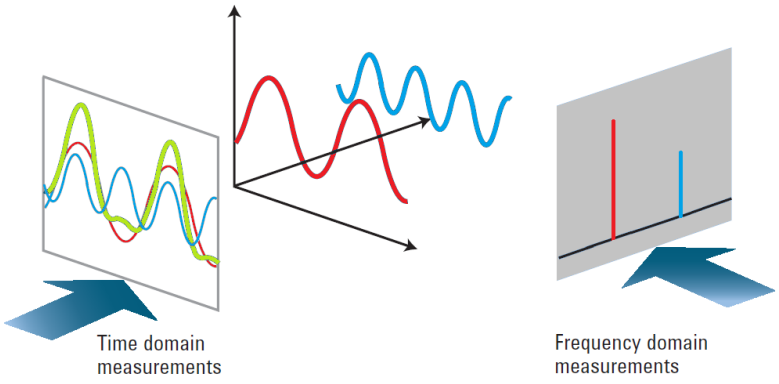

Fig. 1 Time domain vs frequency domain measurements.

Both point of view allow us to make a nearly complete analysis of the main characteristics of the signal [51-56]. As we can see in Eq.(10), DFT consists in a product between a complex matrix by a real vector (signal). This gives us a vector output also complex [55, 56]. Therefore, for practical reasons, it is more useful to use the Power Spectral Density (PSD) [51-56].

On the other hand, if we rewrite Eq.(10), we will have

$$X = \mathrm{F}\, x \tag{11}$$

where X is the output vector (frequency domain), F is the DFT matrix (see Eq.5), and x is the input vector (time domain), then

$$\mathrm{PSD} = \frac{X .\times conj\left(X\right)}{NFFT \times L} \tag{12}$$

In Eq.(12), "$.\times$" means infixed version of Hadamard's product of vectors [57], e.g., if we have two vectors $A = \{a_0,...,a_{N-1}\}$ and $B = \{b_0,...,b_{N-1}\}$, then $A.\times B = \{a_0\, b_0, a_1\, b_1,..., a_{N-1}\, b_{N-1}\}$, while *conj*(•) means *complex conjugate of* (•), while, "$\times$" means simply product of scalars.

In DSP, some authors work with the square root of PSD [51-54], and others - on the contrary - with the modulus (or absolute value) of X [55, 56], directly.

Spectral analysis - When the DFT is used for signal spectral analysis, the $\{x_n\}$ sequence usually represents a finite set of uniformly spaced time-samples of some signal $x(t)$, where t represents time. The conversion from continuous time to samples (discrete-time) changes the underlying Fourier transform of $x(t)$ into a discrete-time Fourier transform (DTFT), which generally entails a type of distortion called aliasing. Choice of an appropriate sample-rate (see *Nyquist rate*) is the key to minimizing that distortion. Similarly, the conversion from a very long (or infinite) sequence to a manageable size entails a type of distortion called *leakage*, which is manifested as a loss of detail (a.k.a. resolution) in the DTFT. Choice of an appropriate sub-sequence length is the primary key to minimizing that effect. When the available data (and time to process it) is more than the amount needed to attain the desired frequency resolution, a standard technique is to perform multiple DFTs, for example to create a spectrogram. If the desired result is a power spectrum and noise or randomness is present in the data, averaging the magnitude components of the multiple DFTs is a useful procedure to reduce the variance of the spectrum (also called a periodogram in this context); two examples of such techniques are the Welch method and the Bartlett method; the general subject of estimating the power spectrum of a noisy signal is called spectral estimation.

A final source of distortion (or perhaps *illusion*) is the DFT itself, because it is just a discrete sampling of the DTFT, which is a function of a continuous frequency domain. That can be mitigated by increasing the resolution of the DFT. That procedure is illustrated at sampling the DTFT [55, 56].

- The procedure is sometimes referred to as *zero-padding*, which is a particular implementation used in conjunction with the fast Fourier transform (FFT) algorithm. The inefficiency of performing multiplications and additions with zero-valued *samples* is more than offset by the inherent efficiency of the FFT.
- As already noted, leakage imposes a limit on the inherent resolution of the DTFT. So there is a practical limit to the benefit that can be obtained from a fine-grained DFT.

Summing-up, we summarize the most important advantages and disadvantages of DFT.

Disadvantages:

- DFT fails at the edges. This is the reason why in the JPEG algorithm (employed in image compression) we use the DCT instead of DFT [42-45]. Even, discrete Hartley transform has an outperform to DFT in DSP and DIP [42, 43].

- No compact support, therefore, to arrive at the frequency domain the correspondence element by element between the two domains (time and frequency) is lost, with a lousy treatment of energy.
- As a consequence of not having compact support, it is not at time. In fact, it moves away from the time domain. For this reason, in the last decades, the scientific community has created some palliatives with better performance in both domain simultaneously, i.e., time and frequency, such tools are: STFT, GT, and wavelets.
- DFT has phase uncertainties (indeterminate phase for magnitude = 0) [55, 56].
- As it arises from the product of a matrix by a vector, its computational cost is $O(N^2)$ for signals (1D), and $O(N^4)$ for images (2D).

All this would seem to indicate that it is a bad transform, however, they are its advantages that keep it afloat. Then, we describe here only some of them.

Advantages:

- As the decisions (relative to filtering and compression) are taken in the spectral domain, the DFT is in its element for both applications. Although as we mentioned before, given its problem with the edges, we use DCT instead DFT.
- It makes the convolutions easier when we use the fast release of DFT, i.e., FFT.
- It is separable (separability property), which is extremely useful when DFT should apply to two and three-dimensional arrays [42-45].
- Given its internal canonical form (distribution of twiddle factors within the DFT matrix), it allows faster versions of itself, such as FFT.

2.1.3 FFT

Fast Fourier Transform - FFT inherits all the disadvantages of the DFT, except the computational complexity of this. In fact, and unlike DFT, the computational cost of FFT is $O(N*log_2N)$ for signals (1D), and $O((N*log_2N)^2)$ for images (2D). For this, it is called fast Fourier transform.

FFT is an algorithm that computes the Discrete Fourier Transform (DFT) of a sequence, or its inverse. Fourier analysis converts a signal from its original domain (often time or space) to the frequency domain and vice versa. An FFT rapidly computes such transformations by factorizing the DFT matrix into a product of sparse (mostly zero) factors [58, 59]. As a result, it manages to reduce the complexity of computing the DFT from $O(N^2)$, which arises if one simply applies the definition of DFT, to $O(N*log_2N)$, where N is the data size. The computational cost for this technique is never greater than the conventional approach and usually significantly less. Further, the computational cost as a function of n is highly continuous, so that linear convolutions of sizes somewhat larger than a power of two.

FFT are widely used for many applications in engineering, science, and mathematics. The basic ideas were popularized in 1965, but some algorithms had been derived as early as 1805 [60]. In 1994 Gilbert Strang described the fast Fourier transform as *the most important numerical algorithm of our lifetime* [61] and it was included in Top 10 Algorithms of 20th Century by the IEEE journal on Computing in Science & Engineering [62].

Overview - There are many different FFT algorithms involving a wide range of mathematics, from simple complex-number arithmetic to group theory and number theory; this article gives an overview of the available techniques and some of their general properties, while the specific algorithms are described in subsidiary articles linked below.

The best-known FFT algorithms depend upon the factorization of N, but there are FFTs with $O(N*log_2N)$ complexity for all N, even for prime N. Many FFT algorithms only depend on the fact that $e^{-2\pi i/N}$ is an N-th primitive root of unity, and thus can be applied to analogous transforms over any finite field, such as number-theoretic transforms. Since the inverse DFT is the same as the DFT, but with the opposite sign in the exponent and a $1/N$ factor, any FFT algorithm can easily be adapted for it.

2.1.4 Another transforms members or not of Fourier Theory

- The ***short-time Fourier transform*** (**STFT**), or alternatively *short-term Fourier transform*, is a Fourier-related transform used to determine the sinusoidal frequency and phase content of local sections of a signal as it changes over time [63, 64]. In practice, the procedure for computing STFTs is to divide a longer time signal into shorter segments of equal length and then compute the Fourier transform separately on each shorter segment. This reveals the Fourier spectrum on each shorter segment. One then usually plots the changing spectra as a function of time [63-67].

- The ***Gabor transform*** (GT) is a special case of the short-time Fourier transform. It is used to determine the sinusoidal frequency and phase content of local sections of a signal as it changes over time [64, 67]. This simplification makes the Gabor transform practical and realizable, and with very important applications, such as: face and fingerprint recognition, texture features and classification, facial expression classification, face reconstruction, fingerprint recognition, facial landmark location, and iris recognition [42-45], etc.

- In mathematics, in the area of harmonic analysis, the ***fractional Fourier transform*** (FRFT) is a family of linear transformations generalizing the Fourier transform [68-81]. It can be thought of as the Fourier transform to the *n*-th power, where *n* need not be an integer - thus, it can transform a function to any *intermediate* domain between time and frequency. Its applications range from filter design and signal analysis to phase retrieval and pattern recognition.

- The ***wavelet transform*** (WT) is based on a *wavelet series*, which is a representation of a square-integrable (real-or complex-valued) function by a certain orthonormal series generated by a wavelet. Nowadays, WT is one of the most popular candidates of the time-frequency-transformations [82-131].

Nevertheless, these internal and external improvements to the Fourier Theory (respectively) do not represent a practical contribution in determining the spectral components (frequency) of a signal for each instant.

2.2 Fourier Uncertainty Principle

In quantum mechanics, the uncertainty principle [132], also known as Heisenberg's uncertainty principle, is any of a variety of mathematical inequalities asserting a fundamental limit to the precision with which certain pairs of physical properties of a particle, known as complementary variables, such as energy *E* and time *t*, can be known simultaneously, although *p* and *x* are other important, i.e., position and momentum, respectively. They cannot be simultaneously measured with arbitrarily high precision. There is a minimum for the product of the uncertainties of these two measurements. Introduced first in 1927, by the German physicist Werner Heisenberg, it states that the more precisely the position of some particle is determined, the less precisely its momentum can be known, and vice versa. The formal inequality relating the uncertainty of energy ΔE and the uncertainty of time Δt was derived by Earle Hesse Kennard later that year and by Hermann Weyl in 1928:

$$\Delta E \, \Delta t \geq \hbar/2 \tag{13}$$

where $\hbar$ is the reduced Planck constant, $h / 2\pi$. The energy associated to such system is

$$E = \hbar\omega \tag{14}$$

where $\omega = 2\pi f$, being f the frequency, and ω the angular frequency.

Then, any uncertainty about ω is transferred to the energy, that is to say,

$$\Delta E = \hbar \, \Delta\omega \tag{15}$$

Replacing Eq.(15) into (13), we will have,

$$\hbar\,\Delta\omega\,\Delta t \geq \hbar/2 \tag{16}$$

Finally, simplifying Eq.(16), we will have,

$$\Delta\omega\,\Delta t \geq 1/2 \tag{17}$$

Eq.(17) say us that a simultaneous decimation in time and frequency is impossible for FFT. Therefore, we must make do with decimate in time or frequency, but not both at once. The last four transforms (STFT, GT, FrFT, and WT) represent a futile effort -to date- to link more closely (individually) each sample in time with its counterpart in frequency in a biunivocal correspondence. That is to say, they are transforms without compact support. Although one of them (WT) sometimes has compact support [82-131].

3 Quantum Spectral Analysis (QuSA)

3.1 In the beginning … Schrödinger's equation

3.1.1 Qubits and Bloch's sphere

The bit is the fundamental concept of classical computation and classical information. Quantum computation and quantum information are built upon an analogous concept, the quantum bit, or qubit for short. In this section we introduce the properties of single and multiple qubits, comparing and contrasting their properties to those of classical bits [1]. The difference between bits and qubits is that a qubit can be in a state other than $|0\rangle$ or $|1\rangle$ [1, 2]. It is also possible to form linear combinations of states, often called superpositions:

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle, \tag{18}$$

where $|\psi\rangle$ is called *wave function*, $|\alpha|^2 + |\beta|^2 = 1$, with the states $|0\rangle$ and $|1\rangle$ are understood as different polarization states of light. Besides, a column vector $|\psi\rangle$ is called a *ket* vector $[\alpha\ \beta]^{\mathrm{T}}$, where, $(\bullet)^{\mathrm{T}}$ means transpose of (•), while a row vector $\langle\psi|$ is called a *bra* vector $[\alpha\ \beta]$. The numbers α and β are complex numbers, although for many purposes not much is lost by thinking of them as real numbers. Put another way, the state of a qubit is a vector in a two-dimensional complex vector space. The special states $|0\rangle$ and $|1\rangle$ are known as Computational Basis States (CBS), and form an orthonormal basis for this vector space, being

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad \text{and} \quad |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

One picture useful in thinking about qubits is the following geometric representation.

Because $|\alpha|^2 + |\beta|^2 = 1$, we may rewrite Eq.(18) as

$$|\psi\rangle = e^{i\gamma}\left(\cos\frac{\theta}{2}|0\rangle + e^{i\phi}\sin\frac{\theta}{2}|1\rangle\right) = e^{i\gamma}\left(\cos\frac{\theta}{2}|0\rangle + (\cos\phi + i\sin\phi)\sin\frac{\theta}{2}|1\rangle\right) \tag{19}$$

where $0 \leq \theta \leq \pi$, $0 \leq \phi < 2\pi$. We can ignore the factor of $e^{i\gamma}$ out the front, because it has no observable effects [1], and for that reason we can effectively write

$$|\psi\rangle = \cos\frac{\theta}{2}|0\rangle + e^{i\phi}\sin\frac{\theta}{2}|1\rangle \tag{20}$$

The numbers θ and ϕ define a point on the unit three-dimensional sphere, as shown in Fig.2.

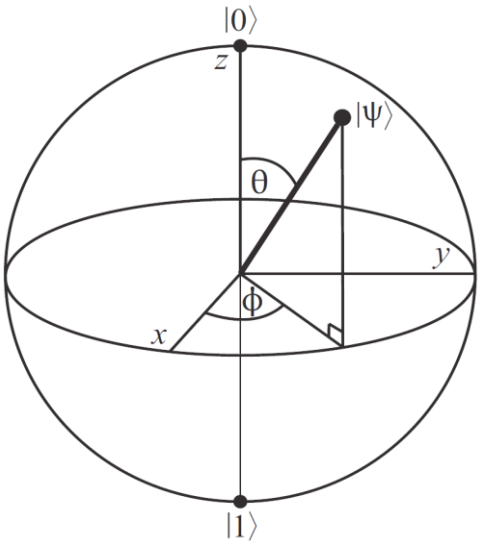

Fig. 2 Bloch's Sphere.

Quantum mechanics is mathematically formulated in Hilbert space or projective Hilbert space. The space of pure states of a quantum system is given by the one-dimensional subspaces of the corresponding Hilbert space (or the "points" of the projective Hilbert space). In a two-dimensional Hilbert space this is simply the complex projective line, which is a geometrical sphere.

This sphere is often called the Bloch's sphere; it provides a useful means of visualizing the state of a single qubit, and often serves as an excellent testbed for ideas about quantum computation and quantum information. Many of the operations on single qubits which can be seen in [1] are neatly described within the Bloch's sphere picture. However, it must be kept in mind that this intuition is limited because there is no simple generalization of the Bloch's sphere known for multiple qubits [1, 2].

Except in the case where $|\psi\rangle$ is one of the ket vectors $|0\rangle$ or $|1\rangle$ the representation is unique. The parameters θ and ϕ, re-interpreted as spherical coordinates, specify a point $\vec{a} = (sin\theta cos\phi, sin\theta sin\phi, cos\theta)$ on the unit sphere in $\mathbb{R}^3$ (according to Eq.19).

Figure 3 highlights all components (details) concerning the Bloch's sphere, namely

$$\textit{Spin down} = |\downarrow\rangle = |0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \textit{qubit basis state} = \textit{North Pole} \tag{21}$$

and

$$\textit{Spin up} = |\uparrow\rangle = |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \textit{qubit basis state} = \textit{South Pole} \tag{22}$$

Both poles play a fundamental role in the development of the quantum computing [1]. Besides, a very important concept to the affections of the development quantum information processing, in general, i.e., the notion of latitude (parallel) on the Bloch's sphere is hinted. Such parallel as shown in green in Fig.3, where we can see the complete coexistence of poles, parallels and meridians on the sphere, including computational basis states ($|0\rangle, |1\rangle$).

Finally, the poles and the parallels form the geometric bases of criteria and logic needed to implement any quantum gate or circuit.

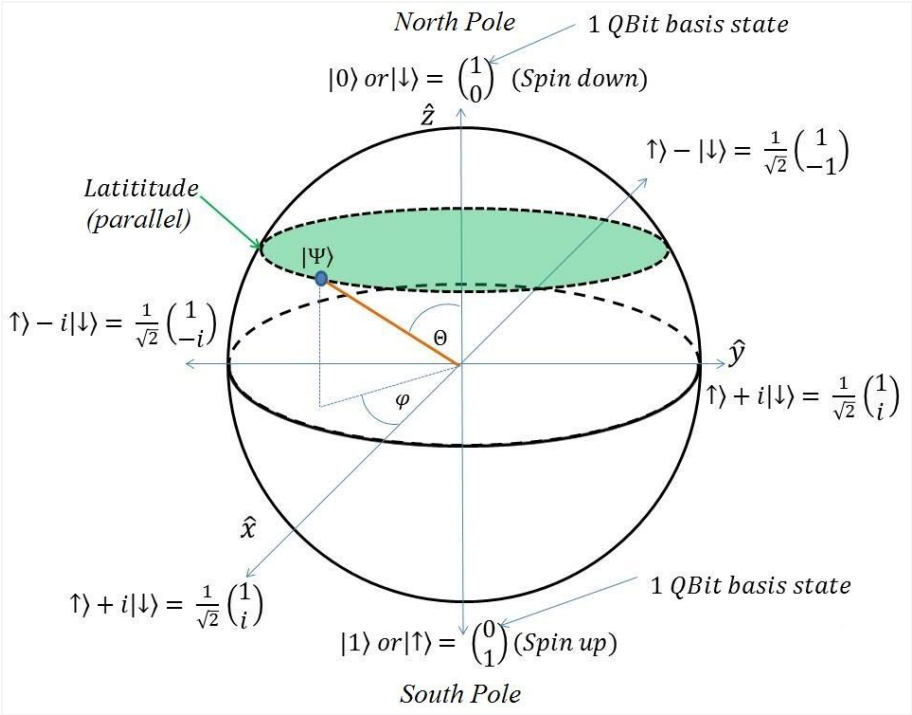

Fig. 3 Details of the poles, as well as an example of parallel and several qubit states on the sphere.

3.1.2 Schrödinger's equation and unitary operators

A quantum state can be transformed into another state by a unitary operator, symbolized as U ($U : H \rightarrow H$ on a Hilbert space H, being called an unitary operator if it satisfies $U^{\dagger}U = UU^{\dagger} = I$, where $(\bullet)^{\dagger}$ is the adjoint of (•), and I is the identity matrix), which is required to preserve inner products: If we transform $|\chi\rangle$ and $|\psi\rangle$ to $U|\chi\rangle$ and $U|\psi\rangle$, then $\langle\chi|U^{\dagger}U|\psi\rangle = \langle\chi|\psi\rangle$. In particular, unitary operators preserve lengths:

$\langle\psi|U^{\dagger}U|\psi\rangle = \langle\psi|\psi\rangle = 1$, if $|\psi\rangle$ is on the Bloch's sphere (i.e., it is a pure state). (23)

On the other hand, the unitary operator satisfies the following differential equation known as the Schrödinger equation [1-4]:

$$\frac{d}{dt}U(t+\Delta t,t) = \frac{-i\hat{H}}{\hbar}U(t+\Delta t,t) \tag{24}$$

where $\hat{H}$ represents the Hamiltonian matrix of the Schrödinger equation, $i = \sqrt[2]{-1}$, and $\hbar$ is the reduced Planck constant, i.e., $\hbar = h/2\pi$. Multiplying both sides of Eq.(24) by $|\psi(t)\rangle$ and setting

$$|\psi(t+\Delta t)\rangle = U(t+\Delta t,t)|\psi(t)\rangle \tag{25}$$

Being $U(t+\Delta t,t) = U(t+\Delta t-t) = U(\Delta t)$ an unitary transform (operator and matrix), yields

$$\frac{d}{dt}|\psi(t)\rangle = \frac{-i\hat{H}}{\hbar}|\psi(t)\rangle \tag{26}$$

The solution to the Schrödinger equation is given by the matrix exponential of the Hamiltonian matrix, that is to say, the unitary operator:

$$U(t+\Delta t,t)=e^{\frac{-i\hat{H}\Delta t}{\hbar}} \quad \text{(if Hamiltonian is not time dependent)} \tag{27}$$

and

$$U(t+\Delta t,t)=e^{\frac{-i}{\hbar}\int_0^t \hat{H}\,dt} \quad \text{(if Hamiltonian is time dependent)} \tag{28}$$

Thus the probability amplitudes evolve across time according to the following equation:

$$|\psi(t+\Delta t)\rangle = e^{\frac{-i\hat{H}\Delta t}{\hbar}}|\psi(t)\rangle \quad \text{(if Hamiltonian is not time dependent)} \tag{29}$$

or

$$|\psi(t+\Delta t)\rangle = e^{\frac{-i}{\hbar}\int_0^t \hat{H}\,dt}|\psi(t)\rangle \quad \text{(if Hamiltonian is time dependent)} \tag{30}$$

The Eq.(29) is the main piece in building circuits, gates and quantum algorithms, being U who represents such elements [1].

Finally, the discrete version of Eq.(26) is

$$|\psi_{k+1}\rangle = \frac{-i\hat{H}}{\hbar}|\psi_k\rangle, \tag{31}$$

for a time dependent (or not) Hamiltonian, being *k* the discrete time.

3.1.3 Quantum Circuits, Gates, and Algorithms; Reversibility and Quantum Measurement

As we can see in Fig.4, and remember Eq.(25), the quantum algorithm (identical case to circuits and gates) viewed as a transfer (or mapping input-to-output) has two types on output:

a) the result of algorithm (circuit of gate), i.e., $|\psi_n\rangle$, with $n=\Delta k$ and $k=0$

b) part of the input $|\psi_0\rangle$, i.e., $|\underline{\psi}_0\rangle$ (underlined $|\psi_0\rangle$), in order to impart reversibility to the circuit, which is a critical need in quantum computing [1].

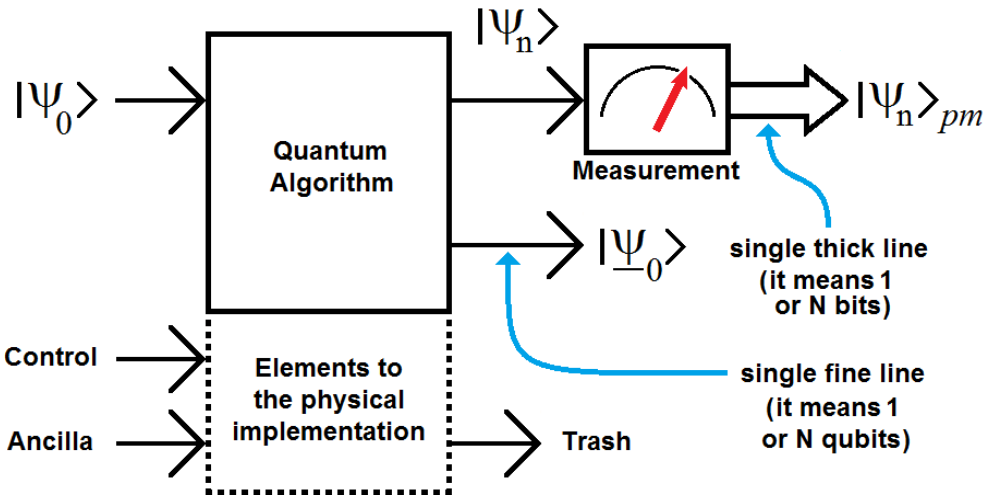

Fig. 4 Module to measuring, quantum algorithm and the elements needs to its physical implementation.

Besides, we can see clearly a module for measuring $|\psi_n\rangle$ with their respective output, i.e., $|\psi_n\rangle_{pm}$ (or, $|\psi_n\rangle$ post-measurement), and a number of elements needed for the physical implementation of the quantum algorithm (circuit or gate), namely: control, ancilla and trash [1].

In Fig.4 as well as in the rest of them (unlike [1]) a single fine line represents a wire carrying *1* qubit or *N* qubits (qudit), interchangeably, while a single thick line represents a wire carrying *1* or *N* classical bits, interchangeably too. However, the mentioned concept of reversibility is closely related to energy consumption, and hence to the Landauer's Principle [1].

On the other hand, computational complexity studies the amount of time and space required to solve a computational problem. Another important computational resource is energy. In [1], the authors show the energy requirements for computation. Surprisingly, it turns out that computation, both classical and quantum, can in principle be done without expending any energy! Energy consumption in computation turns out to be deeply linked to the reversibility of the computation. In other words, it is inexcusable the need of the $|\underline{\psi}_0\rangle$ presence to the output of quantum gate [1].

On the other hand, in quantum mechanics, measurement is a non-trivial and highly counter-intuitive process. Firstly, because measurement outcomes are inherently probabilistic, i.e. regardless of the carefulness in the preparation of a measurement procedure, the possible outcomes of such measurement will be distributed according to a certain probability distribution. Secondly, once the measurement has been performed, a quantum system in unavoidably altered due to the interaction with the measurement apparatus. Consequently, for an arbitrary quantum system, pre-measurement and post-measurement quantum states are different in general [1].

Postulate. Quantum measurements are described by a set of measurement operators $\{\hat{M}_m\}$, index *m* labels the different measurement outcomes, which act on the state space of the system being measured. Measurement outcomes correspond to values of *observables*, such as position, energy and momentum, which are Hermitian operators [1] corresponding to physically measurable quantities.

Let $|\psi\rangle$ be the state of the quantum system immediately before the measurement. Then, the probability that result *m* occurs is given by

$$p(m) = \langle\psi|\hat{M}_m^\dagger \hat{M}_m|\psi\rangle \tag{32}$$

and the post-measurement quantum state is

$$|\psi\rangle_{pm} = \frac{\hat{M}_m|\psi\rangle}{\sqrt{\langle\psi|\hat{M}_m^\dagger \hat{M}_m|\psi\rangle}} \tag{33}$$

Operators $\hat{M}_m$ must satisfy the completeness relation of Eq.(34), because that guarantees that probabilities will sum to one; see Eq.(35) [1]:

$$\sum_m \hat{M}_m^\dagger \hat{M}_m = I \tag{34}$$

$$\sum_m \langle\psi|\hat{M}_m^\dagger \hat{M}_m|\psi\rangle = \sum_m p(m) = 1 \tag{35}$$

Let us work out a simple example. Assume we have a polarized photon with associated polarization orientations 'horizontal' and 'vertical'. The horizontal polarization direction is denoted by $|0\rangle$ and the vertical polarization direction is denoted by $|1\rangle$.

Thus, an arbitrary initial state for our photon can be described by the quantum state $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ (remembering Subsection 3.1.1, Eq.18), where α and β are complex numbers constrained by the normalization condition $|\alpha|^2 + |\beta|^2 = 1$, and $\{|0\rangle, |1\rangle\}$ is the computational basis spanning H^2. Then, we construct two measurement operators $\hat{M}_0 = |0\rangle\langle 0|$ and $\hat{M}_1 = |1\rangle\langle 1|$ and two measurement outcomes a_0, a_1. Then, the full observable used for measurement in this experiment is $\hat{M} = a_0|0\rangle\langle 0| + a_1|1\rangle\langle 1|$. According to Postulate, the probabilities of obtaining outcome a_0 or outcome a_1 are given by $\mathrm{p}(a_0) = |\alpha|^2$ and $\mathrm{p}(a_1) = |\beta|^2$. Correspon-ding post-measurement quantum states are as follows: if outcome = a_0, then $|\psi\rangle_{pm} = |0\rangle$; if outcome = a_1 then $|\psi\rangle_{pm} = |1\rangle$.

3.2 QuSA properly speaking

The Eq.(26) represents the Schrödinger equation, which we are going to write it in a better way, so as to simplify notation

$$|\dot{\psi}(t)\rangle = -i\,\Omega(t)\,|\psi(t)\rangle \tag{36}$$

where $|\dot{\psi}(t)\rangle = \frac{d}{dt}|\psi(t)\rangle$ and $\Omega(t) = \frac{\hat{H}(t)}{\hbar}$, being Ω the angular frequency matrix, and $\hat{H}$ the Hamiltonian matrix. Both time dependents, simultaneously, i.e., at each instant, we will have a matrix.

On the other hand, Ω depends on the respective –non relativistic– system, that is to say, where the most general form for one qubit is

$$\Omega(t) = \begin{bmatrix} \omega_{11}(t) & \omega_{12}(t) \\ \omega_{21}(t) & \omega_{22}(t) \end{bmatrix} \tag{37}$$

Two interesting particular cases are represented by

$$\Omega(t) = \begin{bmatrix} \omega_1(t) & 0 \\ 0 & \omega_2(t) \end{bmatrix} \tag{38}$$

and

$$\Omega(t) = \begin{bmatrix} \omega(t) & 0 \\ 0 & \omega(t) \end{bmatrix} = \omega(t)\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \omega(t)\,I = \omega(t) \tag{39}$$

being *I* the identity matrix. Thus, replacing Eq.(39) in Eq.(36), we will have,

$$|\dot{\psi}(t)\rangle = -i\,\omega(t)\,|\psi(t)\rangle\,. \tag{40}$$

Now, we multiply both sides (by left) of Eq.(40) by $\langle\psi(t)|$,

$$\langle\psi(t)|\dot{\psi}(t)\rangle = -i\,\omega(t)\,\langle\psi(t)|\psi(t)\rangle\,. \tag{41}$$

Finally, $\omega(t)$ results,

$$\omega(t) = i\,\frac{\langle\psi(t)|\dot{\psi}(t)\rangle}{\langle\psi(t)|\psi(t)\rangle}\,. \tag{42}$$

Equation (42) represents QuSA for the monotone case. Now, and considering Equations (36) and (37), where Ω represents an irreducible matrix, then, we are going to multiply both sides (by right) of Eq.(36) by $\langle\psi(t)|$, therefore,

$$|\dot{\psi}(t)\rangle\langle\psi(t)| = -i\,\Omega(t)\;|\psi(t)\rangle\langle\psi(t)| \tag{43}$$

Finally, $\Omega(t)$ results,

$$\begin{aligned}\Omega(t) &= i\;|\dot{\psi}(t)\rangle\langle\psi(t)|\left[|\psi(t)\rangle\langle\psi(t)|\right]^{-1} \\ &= i\;|\dot{\psi}(t)\rangle\langle\psi(t)|^{\dagger}\end{aligned} \tag{44}$$

where

$$\langle\psi(t)|^{\dagger} = \langle\psi(t)|\left[|\psi(t)\rangle\langle\psi(t)|\right]^{-1} \text{ (is the pseudoinverse of } |\dot{\psi}(t)\rangle\text{)} \tag{45}$$

Equation (44) represents QuSA for the multitone case, although in practice it is not used.

3.3 Frequency at time (FAT)

Once we have arrived to the classical world (after the collapse of the wave function), we can then apply an adaptation of QuSA to classical signals called *frequency at time* (FAT). In fact, the experimental evidence indicates that FAT give us the frequency of that classical signal at each time. Curiously, this concept is extensive to quantum signals too, including the case of classical and quantum images.

3.3.1 For signals

In the classical version of Eq.(42) we are going to replace qubits by samples of a real signal, therefore, inner products disappear, and the classical version of Eq.(42) in a symbolic form is

$$\Delta\varpi(t) = i\frac{1}{S(t)}\frac{\partial S(t)}{\partial t} = i\;\frac{\dot{S}(t)}{S(t)}, \tag{46}$$

where $\dot{S}(t) = \frac{\partial S(t)}{\partial t}$, and S is a signal defined in $\mathbb{R}^N$, being N the size of the signal, and $\Delta\omega$ the frequency (before -in the context of Schrödinger equation- it is the imaginary angular frequency). Moreover, in certain cases Eq.(46) will be,

$$\Delta\varpi(t) = i\frac{1}{S(t)}\frac{\Delta S(t)}{\Delta t} = i\;\frac{\dot{S}(t)}{S(t)}, \tag{47}$$

This happens because for gate (square signal with a flank with infinite slope in the transition) and semi-gate (square signal with a flank with finite slope in the transition) Eq.(46) and (47) give identical results. On the other hand, and appealing (for simplicity) to the discrete version of $\Delta\varpi$, we will have,

$$\Delta\varpi = i\;\dot{S}./S, \tag{48}$$

where "$./$" represents the infixed version of Hadamard's quotient of vectors [57], $S = [s_0\ s_1\ s_2 \ldots s_{N-1}]$ is a signal of N samples, $\dot{S} = [\dot{s}_0\ \dot{s}_1\ \dot{s}_2 \ldots \dot{s}_{N-1}]$ is its derivative, and $\Delta\varpi = [\Delta\varpi_0\ \Delta\varpi_1\ \Delta\varpi_2 \ldots \Delta\varpi_{N-1}]$. That is to say, for each sample, we will have,

$$\Delta\varpi_n = i\;\dot{s}_n / s_n \qquad \forall n \in [0, N-1], \text{ being: } \dot{s}_n = (s_{n+1} - s_{n-1})/2 \text{ , and } n \text{ the discrete time.} \tag{49}$$

Equation (49) is the discrete version of $\Delta\varpi$ in its most inapplicable form, given that this is not applicable in cases where the denominator is zero (although unlike the FFT, $\Delta\varpi$ has a definite value in FAT via a simple correction), without mentioning that $\Delta\varpi$ is an imaginary operator to be applied to real signals. Therefore, this form is called raw version. To overcome this drawback, we use an enhanced version based on *root mean square* (RMS) of the signal, as the following,

$$\Delta\varpi_{RMS} = i\ \dot{S} / s_{RMS}, \tag{50}$$

where s_{RMS} –in its discrete form- is defined as follows [133]:

$$s_{RMS} = \sqrt{\frac{1}{N}\sum_{n=0}^{N-1} s_n^2}, \tag{51}$$

with,

$$\Delta\varpi_{n,RMS} = i\ \dot{s}_n / s_{RMS}, \tag{52}$$

$\forall n \in [0, N-1]$. On the other hand, and to save the fact that $\Delta\varpi$ is an imaginary operator to be applied to real signals, we will use (based on Eq.48) a more pure and useful version of *frequency at time* (FAT), i.e.:

$$\begin{aligned} \Delta\omega &= \sqrt{\Delta\varpi\ .\times conj(\Delta\varpi)} \\ &= \sqrt{(i\ \dot{S}./S).\times conj(i\ \dot{S}./S)} \\ &= |\dot{S}./S| = |\dot{S}|./|S| = \frac{1}{|S|}\frac{|\Delta S|}{\Delta t} \end{aligned} \tag{53}$$

Being $\Delta f = \Delta\omega / 2\pi = [\Delta f_0\ \Delta f_1\ \Delta f_2 \ldots \Delta f_{N-1}]$, the frequencies in hertz. Besides, $\Delta\omega$ is now a real operator to be applied to real signals. Remember that, this version (the original) depends on a possible denominator equal to zero, therefore, we will use (based on Eq.50) the next version directly dependent on the frequency:

$$\begin{aligned} \Delta f_{RMS} &= \frac{1}{2\pi}\sqrt{\Delta\varpi_{RMS}\ .\times conj(\Delta\varpi_{RMS})} \\ &= \frac{1}{2\pi}\sqrt{(i\ \dot{S}/s_{RMS}).\times conj(i\ \dot{S}/s_{RMS})} \\ &= \frac{1}{2\pi}|\dot{S}|/s_{RMS} \end{aligned} \tag{54}$$

that is to say,

$$\Delta f_{n,RMS} = \frac{1}{2\pi}|\dot{s}_n|/s_{RMS}, \quad \forall n \in [0, N-1] \tag{55}$$

Note: if $s_{RMS} = 0$, that means that the complete signal *S* is null in all its samples (i.e., $s_n = 0$, $\forall n \in [0, N-1]$) then $\Delta\omega_{n,RMS} = 0$, and hence, $\Delta f_{n,RMS} = 0, \forall n \in [0, N-1]$. In that case, we don't need spectral analysis.

Example **-** Next, we will implement the RMS version of FAT to a signal as shown in Fig.5, which is an electrocardiographic (ECG) signal of 80 pulses per second, with 256 samples per cycle. Top of Fig.5 shows the ECG, while its down shows the waterfall of ECG signal, where the positive peaks are clear, the negative peaks are dark, and the intermediates are gray.

On the other hand, Fig.6 shows the same signal of Fig.5, i.e., ECG of 80 cycles per second, with 256 samples per cycle, however, in this case: ECG signal in blue, and FAT in red with their respective scales, i.e., ECG

scale in blue to the left and FAT scale in red to the right. It is important to mention that the bottom of the figure shows a sequence of witness bars [134]. The distribution of such witness bars is in each case the FAT itself, that is to say, the accumulation of said bars has to do with the flanks of the original signal, in other words, the most pronounced flanks accumulate more bars, while less steep flanks accumulate less bars [134]. This indicates us that the bars are witnessing an indirect flank detection, and thanks to FAT, and the existing spectral components thanks to the steep flank [134].

As we can see in Fig.6, FAT is a flank detector, i.e., it reacts with the spectral components which are represented by the degree of inclination of flanks in time. Finally, in Fig.6, we can notice that the FAT reaches the maximum where the signal has more pronounced flanks.

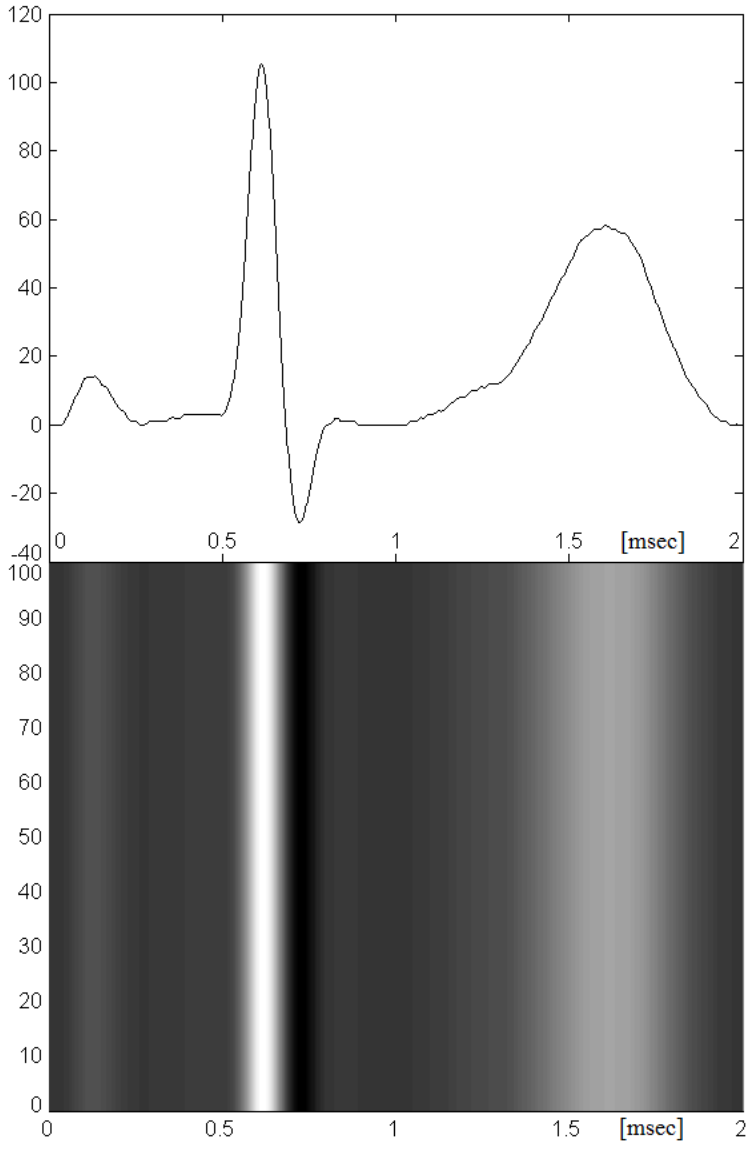

Fig. 5 Top: electrocardiographic signal. Down: its waterfall.

Finally, in [134] we can find several complementary versions of FAT for signals and images. Such versions implies the overlapping of samples (for signals) or pixels (for images) which are part of a mask (of the convolution type). In fact, this feature was used in both examples of this paper. However, it is important to clarify the existence of another versions based on no-overlapping mask, which generate approximation sub-bands (low frequency) and detail (high frequency), being very useful in many other applications [134].

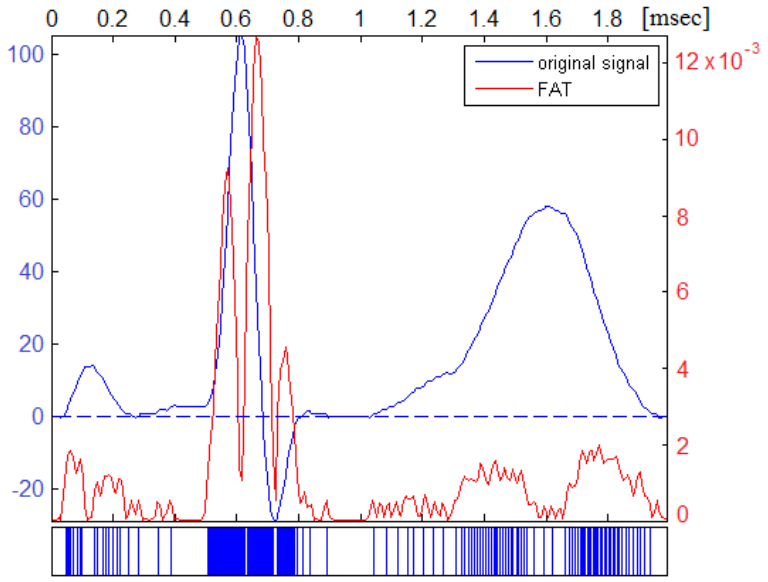

Fig. 6 Here, we have an ECG signal of 80 cycles per second, with 256 samples in blue, the FAT in red and a sequence of witness bars in blue at the bottom of the figure. The distribution of such witness bars is in a perfect relationship with the flank of the ECG signal, i.e., the most pronounced flanks accumulate more bars, while less steep flanks accumulate less bars, in a perfect harmony with the peaks of FAT.

3.3.2 For images

In the classical version of Eq.(42) but in the 2D case, and for each color channel (i.e., red-green-blue), we are going to replace qubits by pixels of a real image, therefore, FAT for this case is represented by three directional components, depending on the direction of each derivative for each color.

Consequently, the image is padded depending on the value of the mask (M = 3), i.e., if the image (e.g., red channel: I_R) has a ROW-by-COL size, then, $I_{R,P}$ (padded I_R) will have a (ROW+2L)-by-(COL+2L) size, where L = (M-1)/2. Therefore, the original image I_R will be in the middle of the padded image $I_{R,P}$, which will have four lateral margins of L size to each side of I_R composed exclusively by zeros.

Besides, we will have two masks, namely:

$$N_H = \frac{[-1\,0\,1]}{2}, \quad \text{(horizontal mask), and} \tag{56}$$

$$N_V = N_H^T, \quad \text{(vertical mask).} \tag{57}$$

The procedure begins with a two-dimensional convolution (first horizontal and then, vertical rafters) between N_H and $I_{R,P}$, i.e.,

$$I_H = N_H * I_{R,P} \tag{58}$$

After that, we continue with another two-dimensional convolution (first vertical and then, horizontal rafters) between N_V and $I_{R,P}$, i.e.,

$$I_V = N_V * I_{R,P} \tag{59}$$

Finally, $\dot{I}$ is obtained via Pythagoras between I_H, and I_V, that is to say,

$$\dot{I} = \sqrt{I_H^2 + I_V^2} \tag{60}$$

Then, we obtain the two-dimensional version of Eq.(47), that is,

$$\Delta\varpi = i\ \dot{I}./\ I\,, \tag{61}$$

While, for each pixel, we will have,

$$\Delta\varpi_{r,c} = i\ \dot{I}_{r,c} / I_{r,c} \qquad \forall r \in [1, ROW], and\ c \in [1, COL] \tag{62}$$

Similar to signal case, Eq.(62) is the discrete version of $\Delta\varpi$ in its most inapplicable form, given that this is not applicable in cases where the denominator is zero (although unlike the FFT, $\Delta\varpi$ has solution), without mentioning that $\Delta\varpi$ is an imaginary operator to be applied to real images. Therefore, this form is called raw version. To overcome this drawback, we use an equalized version (because it is the most practical case for images), as the following,

$$\Delta\varpi_{eq} = i\ \dot{I}_{eq}./\ I_{eq}\,, \tag{63}$$

where subscript "*eq*" means *equalized*. In general, we will pass each pixel of each channel of color of I from $[0,2^8\text{-}1]$ to $[1, 2^8]$.

On the other hand, and to save the fact that $\Delta\varpi$ is an imaginary operator to be applied to real images, we will use (based on Eq.61) a more pure and useful version of *frequency at time* (FAT), i.e.:

$$\begin{aligned}\Delta\omega &= \sqrt{\Delta\varpi\ .\times conj\left(\Delta\varpi\right)} \\ &= \sqrt{\left(i\ \dot{I}./\ I\right).\times conj\left(i\ \dot{I}./\ I\right)} \\ &= \left|\dot{I}./\ I\right| = \left|\dot{I}\right|./\ I\end{aligned} \tag{64}$$

Being $\Delta f = \Delta\omega / 2\pi$ a matrix of ROW-by-COL frequencies in hertz, besides, $|I| = I$, because all values of each color channel are positive. Remember that, this version (raw) depends on a possible denominator equal to zero, therefore, we will use (based on Eq.63) the next version directly dependent on the frequency:

$$\begin{aligned}\Delta f_{eq} &= \frac{1}{2\pi}\sqrt{\Delta\varpi_{eq}\ .\times conj\left(\Delta\varpi_{eq}\right)} \\ &= \frac{1}{2\pi}\sqrt{\left(i\ \dot{I}_{eq}./\ I_{eq}\right).\times conj\left(i\ \dot{I}_{eq}./\ I_{eq}\right)} \\ &= \frac{1}{2\pi}\left|\dot{I}_{eq}./\ I_{eq}\right| = \frac{1}{2\pi}\left|\dot{I}_{eq}\right|./\ I_{eq}\end{aligned} \tag{65}$$

Note: here too, $\left|I_{eq}\right| = I_{eq}$, because all values of each color channel are positive.

Example **-** Next, we will implement the seen version, for which, we select a color image, i.e.: Angelina, a picture of 1920-by-1080 pixel with 24 bpp. See Fig.7.

Figure 8 show us the FAT over Angelina for the equalized version, where (first column, first row) is the original image, (second column, first row) is for red channel, (first column, second row) is for green channel, and (second column, second row) is for blue channel. Besides, in this figure, we can see the texture and edges of the different color channels thanks to FAT. The same set of images show us Regions of Interest (ROIs), which include ergodic areas with a notable impact in the filtering (denoising) and compression contexts.

On the other hand, the FAT by each color indicate us the weight of this one over the main morphological characteristics of the image.

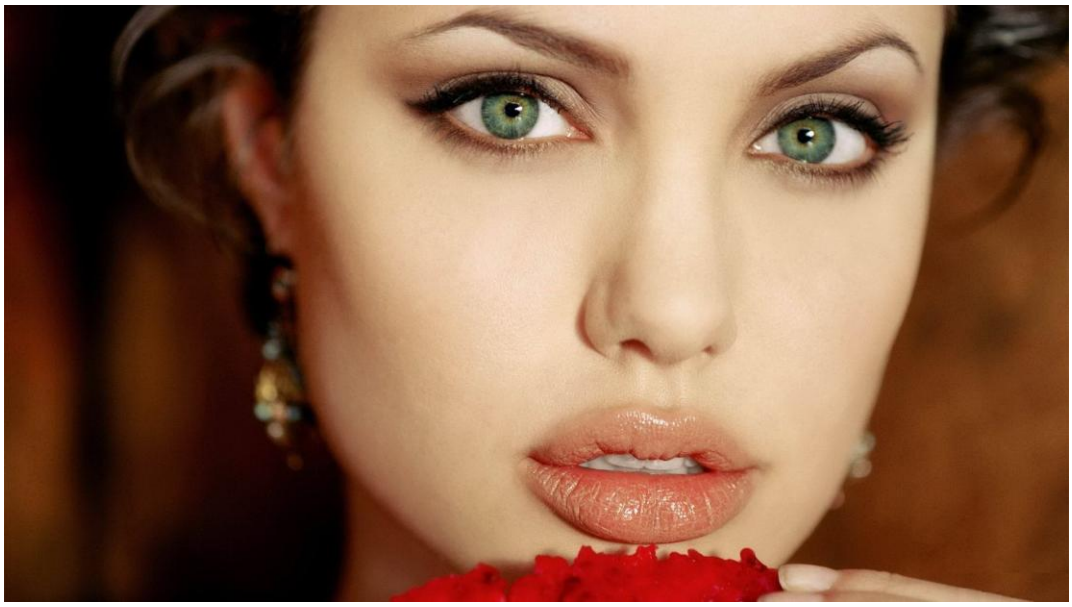

Fig. 7 Angelina: 1920-by-1080 pixels, with 24 bpp.

On the other hand, an important aspect to mention is that although Fig.7 and 8 have different scales, however, the amount of pixels is the same, i.e., FIL-by-COL. Besides, for the three color channel we have manipulated the brightness and contrast for better display scroll of them. Finally, FAT permit us to observe spectral components per pixel by color with a particular emphasis in texture and edges, which are notably important in applications such as visual intelligence for computer vision, image compression [134], filtering (denoising) [134], superresolution [134], forensic analysis of images, image restoration and enhancement [42-45].

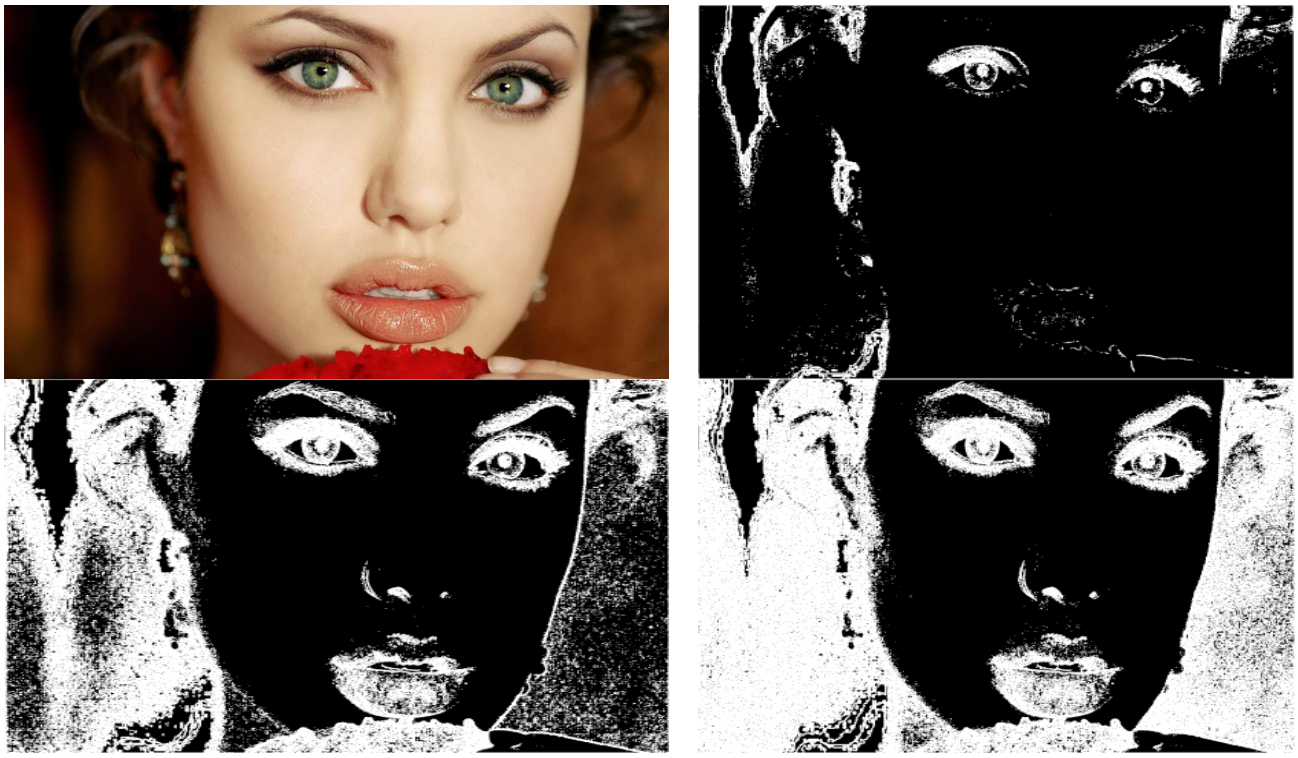

Fig. 8 FAT over Angelina for the equalized version, where (first column, first row) is the original image, (second column, first row) is for red channel, (first column, second row) is for green channel, and (second column, second row) is for blue channel.

As we can see from the above example for signals, the effect of indeterminate FAT when the sample is a value equal to 0 has solution, through the RMS version. Instead, the effect of indeterminate angle (phase) when magnitude = 0 in FFT has no solution [135-138]. Besides, while FFT has no compact support, FAT has it. The latter brings about a lousy treatment of energy by FFT, and an excellent treatment of it by the FAT, to the output of both procedures. Another important comparative aspect between FFT and FAT is the poor performance of the FFT at the edges (both signals and images), whereby the FFT is replaced by the Fast Cosine Transform (FCT) in applications of compression and filtering [42-45]. This problem does not exist in FAT. Besides, the FAT acts as a detector [134], which indicates that encode for the case of compression by the witness bar, similar to PPM or nonlinear sampling [139]. In this sense, it is very convenient to use the bars witness both rows and columns on pictures as a new type of profilometry instead of histograms [42-45], or complementing these [134]. Moreover, the advantages of nonlinear sampling are obvious in the reduction of consumed frequency in communications and signal compression [139].

Other relevant advantages of FAT regarding to FFT are:

- FAT give us an instant notion of the spectral components of the signal or image. In other words, FAT demonstrates directly responsibility of flanks on the characteristics and values of such spectral components.
- FAT is responsive to ergodicity, the *regions of interest* (ROIs), textures, noises, flanks or edges tilt and their relationship with Shannon and Nyquist for nonlinear sampling for Communications [139].
- FFT loses the link with time (because, it doesn't have compact support) [134].
- FAT can be calibrated and related with FFT, easily. See Figures (9) and (10).
- FAT gives frequency in terms of time, directly, i.e., $\Delta f(t) = \Delta\omega(t) / (2\pi)$.
- Two-dimensional QuSA/FAT is directional, and via Pythagoras it is consistent with the idea of directional QuSA for images and N-dimensional arrays.
- In the case of FAT, the convolution mask is (in themselves) a direct filtering processes (denoising). We can see in detail that in [134].
- In FAT, everything is parallelizable: in that case the use of General-purpose graphics processing units (GPGPUs) is recommended [140], and, in fact, FAT is faster than FFT on them.
- In FAT, the Hamiltonian's basal tone [1] is associated with the spectral bands directly. This fact makes calibration be considerably easier, as simple as tuning an instrument. In fact, FAT is known as the spectral analyzer of the poor people.
- Flank detection is equivalent to edge detection in visual intelligence. Besides, FAT detects the sign change and texture and thus assess how compress. Otherwise, FAT permits a nonlinear sampling more efficient than the traditional linear sampling regularly employed, all this from the point of view of the Information Theory [1]. In fact. QuSA/FAT can perform edge detection equal or better than methods Prewitt, Roberts, Sobel and Canny [134]. Although you can easily prove that all of them derive from QuSA/FAT.
- Figure 9 shows in symbolic way both complementarity as the perfect linkage between the two theories, i.e., FFT and QuSA/FAT, instead, Fig.10 shows us such complementarity and linkage in a rigorous form.

Both graphs clearly show a quadrature between FFT and FAT via equalization.

FFT and FAT give information about the same physical element, i.e., the frequency, but in a very different way, in fact, FAT is far superior and accurate (in its ambit) regarding FFT. Besides, unlike FFT, FAT has compact support. However, both are complementary.

Thanks to these two tools (FFT and FAT) we can get the whole universe linked to spectral and temporal analysis (simultaneously) of a signal, image or video. Therefore, we can locate (indirectly) to the FFT at the exact time of the signal by its components. This fact implies a significant advance in the Fourier's theory after almost two and a half centuries.

Since in signals it becomes much more evident everything said, a conspicuous proof (which certifies everything said) is constituted by the following figures (11, 12 and 13).

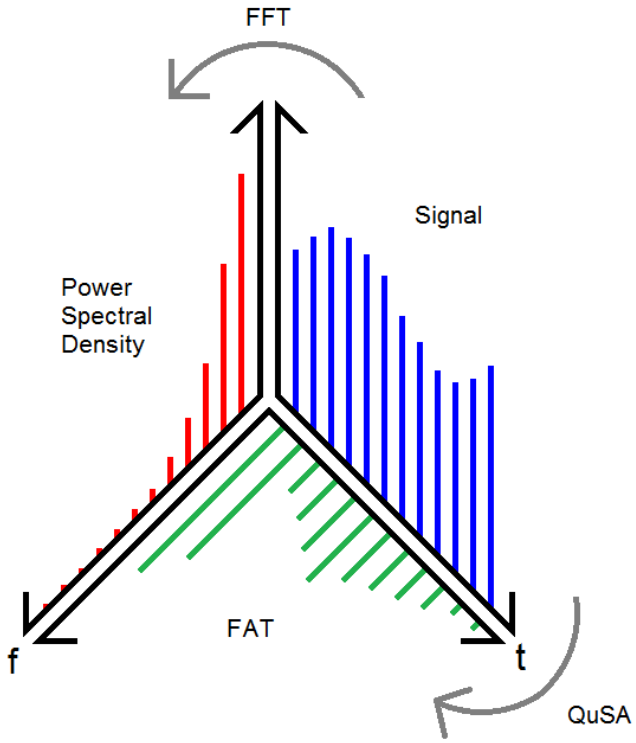

Fig.9 Symbolic relationship between FAT and FFT (PSD).

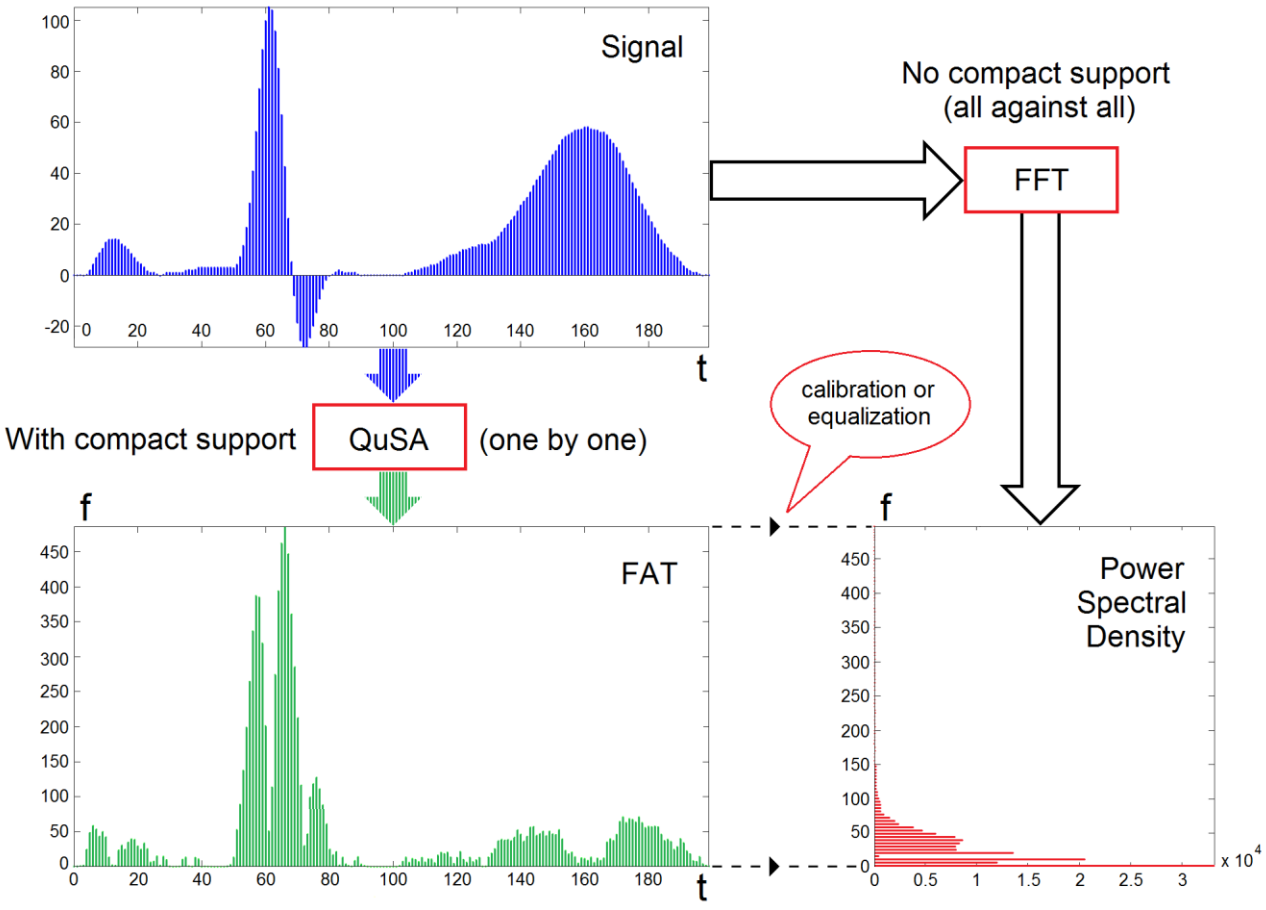

Fig.10 Rigorous relationship between FAT and FFT (PSD).

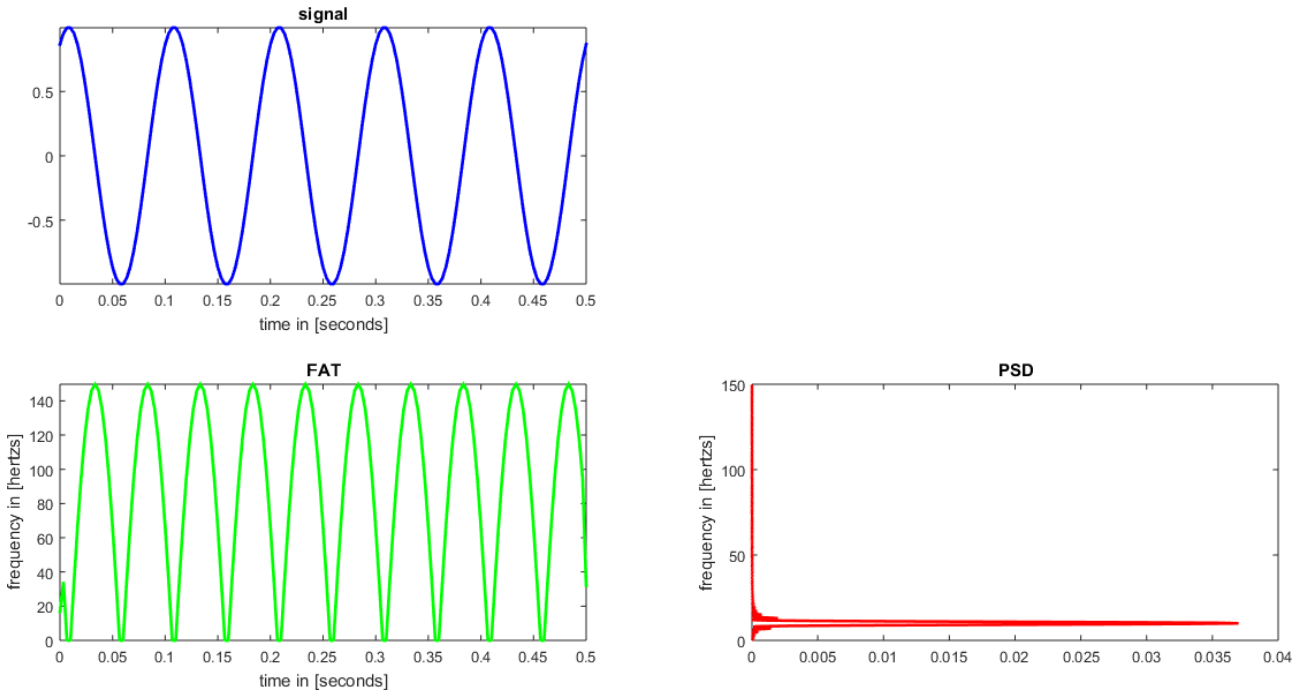

Fig.11 Original signal is a sine with a frequency of 5 Hertz.

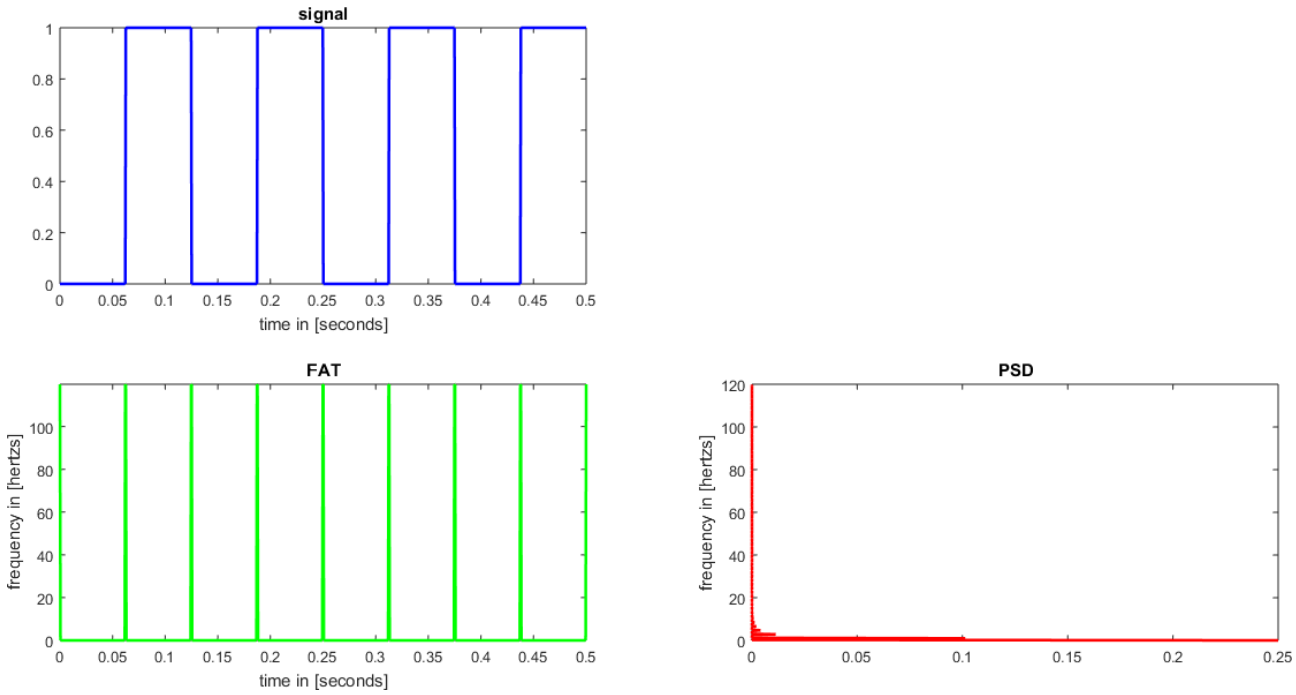

Fig.12 Original signal is a semi-gate with a frequency of 5 Hertz.

In Fig.11 we have a sine of 5 cycles with 1024 samples, in Fig.12 we have a train of semi-gates of 4 cycles with 1024 samples, and in Fig.13 we have a non stationary signal with 1024 samples too. In all of them, we can see (after equalization) the coincidence between the maximum frequency of PSD with the peaks of FAT.

The most relevant aspect regarding this comparison is the fact that FFT and FAT work clearly in quadrature, which is a perfect complement. In fact, this complement allows to complete the indispensable toolbox required in the spectral analysis of signals, images, and video.

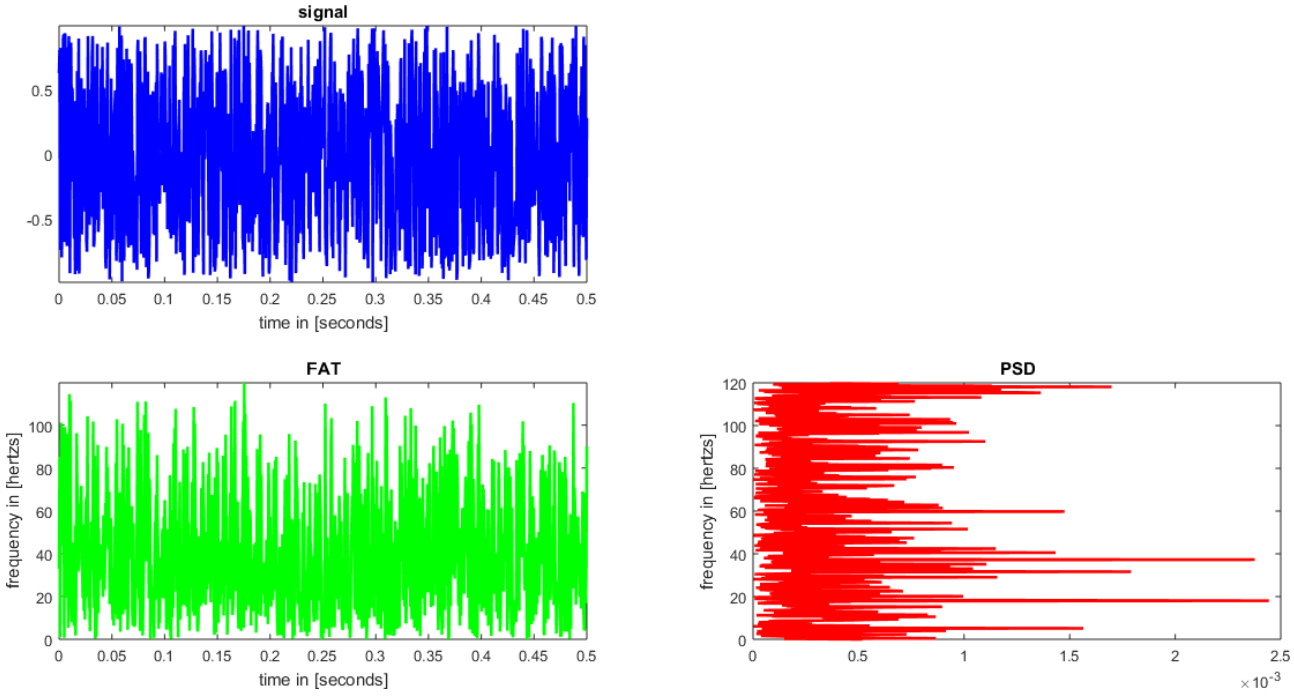

Fig.13 Original signal is a non stationary series.

An important aspect -at this point- we can see it in Fig.12, where we talk about a semi-gate signal. The question is: why do we say of semi-gate instead of gate directly? The answer is in the Fig.14, where we show in detail a few samples of the semi-gate of the Fig.12.

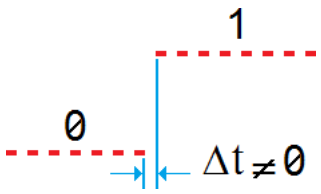

Fig.14 Some samples of Fig.12 (in detail).

Figure 14 shows us -in detail- the distance between two samples of Fig.12 which is a signal simulated (in blue) in MATLAB® code:

```
% Initial parameters
f = 8; % frequency
overSampRate = 30;
fs = overSampRate*f; % sampling frequency
nCyl = 4; % number of cycles
NFFT = 1024; % number of points of FFT
nfft = NFFT/8;
t = 0:nCyl*1/f/(NFFT-1):nCyl*1/f; % time axis
x = [ zeros(1,nfft) ones(1,nfft) zeros(1,nfft) ones(1,nfft) zeros(1,nfft) ones(1,nfft) zeros(1,nfft) ones(1,nfft) ]; % signal

% Calculation of FFT
L = length(x); % length of signal
X = fftshift(fft(x,NFFT));
PSD = X.*conj(X)/(NFFT*L);
fVals = fs*(0:NFFT/2-1)/NFFT; % frequency axis
```

```
% Calculation of FAT
x_RMS = sqrt(x*x'/L);
xp = [ x(L) x x(1) ]; % padding for a cyclic signal. For a non-cyclic signal is xp = [ 0 x 0 ];
dx = [];
for n = 1:L
  dx(n) = (xp(n+2)-xp(n))/2;
end
FAT = abs(dx)/x_RMS/2/pi;
FAT = (FAT-min(FAT))/(max(FAT)-min(FAT))*(max(fVals)-min(fVals))+min(fVals);

subplot(221),plot(t,x,'b','LineWidth',2)
axis([ 0 nCyl*1/f min(x) max(x) ])
title('signal')
xlabel('time in [seconds]')

subplot(223),plot(t,FAT,'g','LineWidth',2)
axis([ 0 nCyl*1/f min(FAT) max(FAT) ])
title('FAT')
xlabel('time in [seconds]')
ylabel('frequency in [hertz]')

subplot(224),plot(PSD(NFFT/2+1:NFFT),fVals,'r','LineWidth',2)
title('PSD')
ylabel('frequency in [hertz]')
```

Clearly, $\Delta t \neq 0$ (then, $\Delta\omega \neq \infty$), as we can see in Fig.14. In fact, Δt = nCyl*1/f/(NFFT-1). This is the reason why we speak of semi-gate signal instead of gate. Instead, if we have $\Delta t = 0$ (then, $\Delta\omega = \infty$), then, we will speak of a gate signal.

On the other hand, the distribution of the witness bars is consistent with the possibility of locating a particle by the wave function, or rather, the probability distribution that arises from this function.

Given the signal y = f (*t*), the witness bars [134] arise as follows:

1. *N* equidistant lines are distributed along the ordinate axis
2. In those settings where these lines intercept the signal, we identify the projections on the axis of abscissae. At these points we place the witness bars, which (if the signal is nonlinear) shall be separated in a not equidistant way depending on the flanks of the signal at each point. This is a nonlinear sampling itself.

Some final considerations:

- The transition from QuSA to FAT represents the collapse of the wave function, i.e., from vector to scalar at each moment.
- Hamiltonian is real, i.e., it isn't hermitic for a confined single particle
- QuSA/FAT can be used in time filtering
- The frequency of a pure tone (sine) is proportional to its higher slope derivative. Instead, if the signal is a gate, FAT will be infinite on the flanks, then, the density of the witness bars is infinite too in these flanks. This is very useful for a better understanding of Sampling and Nyquist theorems [139].
- Like the FFT, the FAT will help in the development of new algorithms for signal, image and video compression, replacing or complementing to FFT or DCT in new versions of, MP3 (audio [141]), JPEG (images [142]) and, H.264 and VP9 (video [143-148]).
- Unlike FFT, FAT does not require decimation in time or frequency.
- For one-dimension FFT has a computational cost of $O(N*log_2(N))$, and FAT of $O(N)$.
- For two-dimensions FFT has a computational cost of $O(N^2*log_2(N)^2)$, and FAT of $O(N^2)$.
- For two-dimensions FFT has a computational cost of $O(N^3*log_2(N)^3)$, and FAT of $O(N^3)$.
- Being so simple, FAT is easily implementable in software, Field-programmable gate array (FPGA) [149], GPGPU [140], firmware [150], and Advanced RISC Machine (ARM) architecture [151].

3.4 *QuSA/FAT Uncertainty Principle*

From Eq.(53) at each instant (without subscript by simplicity), we have,

$$\Delta\omega = \frac{1}{|S|}\frac{|\Delta S|}{\Delta t} \tag{66}$$

With a simple clearance, thus,

$$\left|\frac{\Delta S}{S}\right| = \Delta\omega\,\Delta t \tag{67}$$

If we have present Eq.(17), then

$$\left|\frac{\Delta S}{S}\right| = \Delta\omega\,\Delta t \geq 1/2 \tag{68}$$

Based on Fig.15, if we define a quantum signal as a qubit sequence, and remembering Equations (21) and (22) of Section 3.1.1, spin-down is, $|\downarrow\rangle = |0\rangle$, and spin-up is $|\uparrow\rangle = |1\rangle$, then, we will have:

- *top-left:* a threshold from zero to one for a classical signal (two states)
- *top-right:* a threshold from zero to one for a classical signal (four states)
- *bottom-left:* a threshold from zero to one for a quantum signal (two states)
- *bottom-right:* a threshold from zero to one for a quantum signal (four states)

In the four cases, the flank responsible of state transition happens in an instant, i.e., $\Delta t = 0$. In fact, if we combine Eq.(15) and (66), we will have the ΔE according to QuSA/FAT, that is to say,

$$\Delta E = \frac{\hbar}{|S|}\frac{|\Delta S|}{\Delta t} \tag{69}$$

Besides, from Eq.(68) we can see a trade-off between $\Delta\omega$ and Δt, therefore, if $\Delta t = 0$, then, $\Delta\omega = \infty$, and hence, $\Delta E = \infty$. See Eq.(17), (68) and (69). The fact that $\Delta E = \infty$ does not mean that energy $E = \infty$, nothing to do. It only means that FAT is infinite, nothing more. On the other hand, if $\Delta\omega = \infty$ (and hence, $\Delta E = \infty$) then $\Delta t = 0$. In fact, Fig.15 is and extreme case of Fig.12, with $\Delta t = 0$ and hence $\Delta\omega = \infty$. Therefore, this signal is indeed a gate.

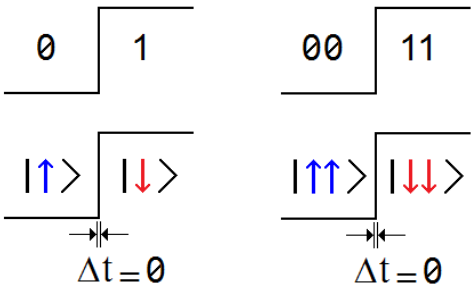

Fig.15 Flank transitions for a gate signal with Δt = 0, so that on the top we have classical signals (i.e., sequence of bits), on the bottom we have quantum signals (i.e., sequence of qubits).

On the other hand, if we apply FAT on a channel, and considering that such channel would allow the survival of such type of signals, i.e., infinite frequency thanks to instantaneous transitions or flanks, then Eq.(68) can be rewritten as,

$$FR\,TL \geq 1/2, \tag{70}$$

where *FR* is the acronym of *frequency*, while *TL* is the acronym of *time-latency*. This means that this instantaneous change or flank generates spectral components with infinite frequency. At this point, it is necessary a better analysis of different aspects regarding the nature and origin of the channel for a more deep understanding of QuSA/FAT on classical and quantum channels.

Another important concept regarding to QuSA/FAT comes up from Eq.(68). That equation shows us the trade-off between Δt and $\Delta\omega$, through which the change in one drag the change in the other. That is to say, we have seen that if $\Delta t = 0$ then $\Delta\omega = \infty$, and instead, if $\Delta\omega = \infty$, then $\Delta t = 0$. And even more important, this attribute of functional dependence is interchangeable. This very strong dependence from Heisenberg Uncertaintly Principle [132] with the mentioned characteristics ensures the projection of FAT on elements as important of Quantum Physics as is the Quantum Entanglement [1-3], in particular, in the latter's implication in Quantum Communication [152-155].

4 Conclusions and future works

This work began with an extensive tour on traditional spectral techniques based on Fourier's Theory, without compact support and completely disconnected from the link between time and frequency (this tour included WT which sometimes has compact support), and the responsibility of each flank with respect to final spectral components of a signal, image or video. For that reason QuSA/FAT was created, i.e., to cover such space and also as a complement to the aforementioned Fourier's Theory, in particular, FFT. A simply comparison between QuSA/FAT and FFT sheds some initial conclusions, which can be seen synthesized in Table I.

Moreover, when the wave function collapses, we pass from QuSA to FAT. This point is essential, because of this begins to be necessary to use the Hadamard's quotient of vectors [57], among others practical concepts.

At this point, it is important to mention that the applications of FAT are obvious, e.g.:
- It's a support and it allows a better understanding of the Information Theory and Quantum Information Theory aimed at improving current signal, image a video compression algorithms, and develop new.
- Its applications range from filter design and signal analysis to phase retrieval and pattern recognition.
- It's an excellent complement to Spectrogram in speech processing [7, 137, 138].
- It's very useful in radar signals analysis, analysis of phase migration in Synthetic Aperture Radar (SAR) raw data, Radioastronomy, sonar, etc.
- It's particularly useful in analysis of time-varying spectral characteristics
- It represents a major contribution in Signals Intelligence (SIGINT), Imagery Intelligence (IMINT), and Communications Intelligence (COMINT) up to day.
- It retains a direct relationship with compressed sensing
- Time series analysis: as a complement of moving average, time analysis of stock exchange, etc
- Biomedical signal and image analysis: electrocardiograph, electroencephalography, evoked potential, brain computer interface
- Study of seismic signals, in general, and, earthquakes, in particular
- Bioinformatics: Signal Processing for DNA Sequence Analysis
- Analysis, synthesis, and speech recognition
- Nonlinear spectral analysis
- Conditioning of acoustic spaces
- Quantum Chaos
- Besides, its applications are obvious in a fine processing signal, namely: power spectral density (with a strict sense of time); frequency-hopping spread spectrum; analysis of stationarity; nonlinear sampling for a most efficient compression schema instead of linear sampling, among many others, see [134].

As we have already said, BAS is an extraordinary tool to assess the importance of the flanks (or edges) in a compression process weighting in real time and sample by sample (or pixel by pixel) the importance of temporal spectral components in the final result.

TABLE I
COMPARISON BETWEEN FFT AND FAT

Characteristics	**FFT**	**FAT**
Separability	Yes	Yes
Compact support	No	Yes
Instantaneous spectral attributes	No	Yes
1D computational cost	$O(N*\log_2(N))$	$O(N)$
2D computational cost	$O(N^2*\log_2(N)^2)$	$O(N^2)$
Energy treatment	Disastrous	Excelent
Decimation	In time or frequency	It does not require
Parallelization	No	Yes

Finally, and as we have seen, FFT doesn't have compact support, therefore, we say that FFT is a non-local process, while, FAT has compact support, so that, we say that FAT is a local process, with all that this indicates when we apply this tool to the study of the quantum entanglement.

References

1. Nielsen, M.A., Chuang, I.L.: Quantum Computation and Quantum Information. Cambridge University Press, Cambridge (2004)
2. Kaye, P., Laflamme, R., Mosca, M.: An Introduction to Quantum Computing. Oxford University Press, Oxford (2004)
3. Stolze, J., Suter, D.: Quantum Computing: A Short Course from Theory to Experiment. WILEY-VCH Verlag GmbH & Co. KGaA, Weinheim (2007)
4. Busemeyer, J.R., Wang, Z., Townsend, J.T.: Quantum dynamics of human decision-making. J. Math. Psychol. **50**, 220–241 (2006) doi:10.1016/j.jmp.2006.01.003
5. Eldar, Y.C.: Quantum Signal Processing. Doctoral Thesis, MIT, Dec. 2001
6. Eldar, Y.C., Oppenheim, A.V.: Quantum Signal Processing. Signal Process. Mag. **19**, 12–32 (2002)
7. Vlaso, A. Y.: Quantum Computations and Images Recognition. arXiv:quant-ph/9703010 (1997)
8. Schützhold, R.: Pattern recognition on a quantum computer. Phy. Rev. A **67**(6), 062311 (2003)
9. Beach, G., Lomont, C., Cohen, C.: Quantum Image Processing (QuIP). Proc. Appl. Imagery Pattern Recognit. Workshop, 39-44 (2003)
10. Venegas-Andraca, S.E., Bose, S.: Storing, processing and retrieving an image using quantum mechanics. Proc. SPIE Conf. Quantum Inf. Comput. vol. 5105, 137–147 (2003)
11. Venegas-Andraca, S.E.: Discrete Quantum Walks and Quantum Image Processing. Thesis submitted for the degree of Doctor of Philosophy at the University of Oxford (2005)
12. Venegas-Andraca, S.E., Ball, J.L.: Processing images in entangled quantum systems. Quantum Inf. Process. **9**(1), 1-11 (2010)
13. Latorre, J.I.: Image compression and entanglement. arXiv:quant-ph/0510031 (2005)
14. Le, P.Q., Dong, F., Hirota, K.: A flexible representation of quantum images for polynomial preparation, image compression, and processing operations. Quantum Inf. Process. **10**(1), 63-84 (2011)
15. Sun, B., Le, P.Q., Iliyasu, A.M., *et al*.:AMulti-channel representation for images on quantum computers using the RGB? color space. Proc. IEEE 7th Intern. Symp. Intelligent Signal Proces, 160-165 (2011)
16. Yan, F., *et al*.: Assessing the Similarity of Quantum Images based on Probability Measurements. 2012 IEEE Cong. Evolutionary Computation (CEC), 1-6 (2012)
17. Le, P.Q., Iliyasu, A.M., Dong, F., Hirota, K.: Efficient color transformations on quantum images. J. Adv. Comput. Intell. Intell. Inf. **15**(6), 698-706 (2011)
18. Le, P.Q., Iliyasu, A.M., Dong, F.Y., Hirota, K.: Fast Geometric Transformations on Quantum Images. IAENG Intern. J. of Applied Mathematics. **40**(3) (2010)
19. Le, P.Q., Iliyasu, A.M., Dong, F.Y., Hirota, K.: Strategies for designing geometric transformations on quantum images. Theoretical Computer Science. **412**(15), 1506-1418 (2011)
20. Srivastava, M., Panigrah, P.K.: Quantum Image Representation Through Two-Dimensional Quantum States and Normalized Amplitude. arXiv:quant-ph/1305.2251 (2013)

21. Li, H.S., Zhu, Q.X., Lan, S., *et al*.: Image storage, retrieval, compression and segmentation in a quantum system. Quantum Inf. Process. **12**(6), 2269-2290 (2013)
22. Li, H.S., Zhu, Q.X. Li, M.C., *et al*.: Multidimensional color image storage, retrieval, and compression based on quantum amplitudes and phases. Information Sciences. **273**, 212-232 (2014)
23. Hu, B.Q., Huang, XD, Zhou, R.G. *et al*.: A theoretical framework for quantum image representation and data loading scheme Science China on Information Science. **57**(3),1-11 (2014)
24. Zhang, Y., Lu, K., Gao, Y., Wang, M.: NEQR: a novel enhanced quantum representation of digital images. Quantum Inf. Process. **12**(8), 2833-2860 (2013)
25. Wang, M., Lu, K., Zhang, Y.: FLPI: representation of quantum images for log-polar coordinate. Fifth Intern. Conf. on Digital Image Processing: ICDIP'2013 (2013)
26. Zhang, Y., Lu, K., Gao, Y., Wang, M.: A novel quantum representation for log-polar images. Quantum Inf. Process. **12**(8), 3103-3126 (2013)
27. Yuan, S., Mao, X., Chen, L. *et al*.: Quantum digital image processing algorithms based on quantum measurement. Optik - International Journal for Light and Electron Optics. **124**(23), 6386-6390 (2013)
28. Yuan, S., Mao, X., Xue, Y., et al.: SQR: a simple quantum representation of infrared images. Quantum Inf. Process. **13**(6), 1353-1379 (2014)
29. Zhang, W.W., Gao, F., Liu B.: A quantum watermark protocol. Int. J. Theor. Phys. **52**(2), 504-513 (2013)
30. Zhang, W.W., Gao, F., Liu B., *et al*.: A watermark strategy for quantum images based on quantum fourier transform. Quantum Inf. Process. **12**(2), 793-803 (2013)
31. Yang, Y.G., Xia, J., Jia, X., *et al*.: Novel image encryption/decryption based on quantum Fourier transform and double phase encoding. Quantum Inf. Process. **12**(11), 3477-3493 (2013)
32. Yang, Y.G., Jia, X., Sun, S.J., *et al*.: Quantum cryptographic algorithm for color images using quantum Fourier transform and double random-phase encoding. Information Sciences. 277, 445-457 (2014)
33. Song, X.H., Niu, X.M.: Comment on: Novel image encryption/decryption based on quantum fourier transform and double phase encoding. Quantum Inf. Process. **13**(6), 1301-1304 (2014)
34. Jiang, N., Wu, W.Y., Wang, L.: The quantum realization of Arnold and Fibonacci image scrambling Quantum Information Processing. Quantum Inf. Process. **13**(5), 1223-1236 (2014)
35. Zhou, R.G., Wu, Q., Zhang, M.Q., Shen, C.Y.: Quantum image encryption and decryption algorithms based on quantum image geometric transformations. Int. J. Theor. Phys. **52**(6), 1802-1817 (2013)
36. Tseng, C.C., Hwang, T.M.: Quantum Digital Image Processing Algorithms. 16th IPPR Conf. on Computer Vision, Graphics and Image Processing: CVGIP'2003. Kinmen, Taiwang (2003)
37. Mastriani, M.: Quantum Image Processing? Quantum Inf Process (2017) 16:27. doi:10.1007/s11128-016-1457-y
38. Altepeter, J.B., Branning, D., Jeffrey, E., Wei, T.C., Kwiat, P.G., Thew, R.T., O'Brien, J.L., Nielsen, M.A., White, A.G.: Ancilla-assisted quantum process tomography. Phys. Rev. Lett. **90**, 193601 (2003)
39. Niggebaum, A.: Quantum State Tomography of the 6 qubit photonic symmetric Dicke State. Thesis submitted for the degree of Doctor of Physics. Ludwig-Maximilians-Universität München (2011)
40. Gross, D., Liu,Y.-K., Flammia, S.T., Becker, S., Eisert, J.: Quantum state tomography via compressed sensing. arXiv:0909.3304 [quant-ph] (2010)
41. Audenaert, K.M.R., Scheel, S.: Quantum tomographic reconstruction with error bars: a Kalman filter approach. N. J. Phys. **11**, 023028 (2009)
42. Jain, A.K.: Fundamentals of Digital Image Processing. Prentice Hall Inc., Englewood Cliffs, NJ (1989)
43. Gonzalez, R.C., Woods, R.E.: Digital Image Processing, Prentice-Hall, Englewood Cliffs (2002)
44. Gonzalez, R.C.,Woods, R.E., Eddins, S.L.: Digital Image Processing Using Matlab. Pearson Prentice Hall, Upper Saddle River (2004)
45. Schalkoff, R.J.: Digital Image Processing and Computer Vision. Wiley, New York (1989)
46. MATLAB® R2015a (Mathworks, Natick, MA). http://www.mathworks.com/
47. Mastriani, M.: Quantum Boolean image denoising. Springer Quantum Information Processing. 14(5), 1647-1673 (2015)
48. Weinstein, Y.S., Lloyd, S., Cory, D.G.: Implementation of the Quantum Fourier Transform (1999) quant-ph arXiv:quant-ph/9906059v1
49. Wikipedia. https://en.wikipedia.org/wiki/Fourier_transform
50. Hsu, H.P.: Fourier Analysis. Simon & Schuster, New York (1970)
51. Wikipedia. https://en.wikipedia.org/wiki/Discrete_Fourier_transform

52. Tolimieri R., An M., Lu C.: Algorithms for Discrete Fourier Transform and convolution. Springer Verlag, New York (1997)
53. Tolimieri R., An M., Lu C.: Mathematics of multidimensional Fourier Transform Algorithms. Springer Verlag, New York (1997)
54. Briggs, W.L., Van Emden, H.: The DFT: An Owner's Manual for the Discrete Fourier Transform. SIAM, Philadelphia (1995)
55. Oppenheim, A.V, Willsky, A.S., Nawab, S. H.: Signals and Systems. Second Edition, Prentice Hall, Upper Saddle River, NJ (1997)
56. Oppenheim, A.V, Schafer, R.W.: Digital Signal Processing. Prentice Hall, Englewood Cliffs, NJ (1975)
57. De Graeve, R., Parisse, B.: Symbolic algebra and Mathematics with Xcas. University of Grenoble I (2007) https://www-fourier.ujf-grenoble.fr/~parisse/giac/cascmd_en.pdf
58. Wikipedia. https://en.wikipedia.org/wiki/Fast_Fourier_transform
59. Van Loan, C.: Computational Frameworks for the Fast Fourier Transform, SIAM (1992)
60. Heideman, M.T., Johnson, D.H., Burrus, C.S.: Gauss and the history of the fast Fourier transform. IEEE ASSP Magazine 1(4), 14–21 (1984) doi:10.1109/MASSP.1984.1162257
61. Strang, G.: Wavelets. American Scientist 82(3), 253 (1994)
62. Dongarra, J., Sullivan, F: Guest Editors Introduction to the top 10 algorithms. Computing in Science Engineering 2(1), 22–23 (2000) doi:10.1109/MCISE.2000.814652. ISSN 1521-9615
63. Wikipedia. https://en.wikipedia.org/wiki/Short-time_Fourier_transform
64. Sejdić, E., Djurović, I., Jiang, J.: Time-frequency feature representation using energy concentration: An overview of recent advances. Digital Signal Processing. 19(1), 153-183 (2009)
65. Jacobsen, E., Lyons, R.: The sliding DFT. Signal Processing Magazine. 20(2), 74–80 (2003)
66. Allen, J.B.: Short Time Spectral Analysis, Synthesis, and Modification by Discrete Fourier Transform. IEEE Trans. on Acoustics, Speech, and Signal Processing. ASSP-25 (3), 235–238 (1977)
67. https://en.wikipedia.org/wiki/Gabor_transform
68. Wikipedia. https://en.wikipedia.org/wiki/Fractional_Fourier_transform
69. Condon, E.U.: Immersion of the Fourier transform in a continuous group of functional transformations. *Proc. Nat. Acad. Sci. USA* **23**, 158–164 (1937)
70. Namias, V.: The fractional order Fourier transform and its application to quantum mechanics. *J. Inst. Appl. Math.* **25**, 241–265 (1980)
71. Wiener, N.: Hermitian Polynomials and Fourier Analysis. *J. Mathematics and Physics* **8**, 70-73 (1929)
72. Almeida, L.B.: The fractional Fourier transform and time-frequency representations. *IEEE Trans. Sig. Processing* **42**(11), 3084–3091 (1994)
73. Tao, R., Deng, B., Zhang, W.-Q, Wang, Y.: Sampling and sampling rate conversion of band limited signals in the fractional Fourier transform domain. *IEEE Trans. on Signal Processing*, **56**(1), 158–171 (2008)
74. Bhandari, A., Marziliano, P.: Sampling and reconstruction of sparse signals in fractional Fourier domain. *IEEE Signal Processing Letters*, **17**(3), 221–224 (2010)
75. Bailey, D.H., Swarztrauber, P.N.: The fractional Fourier transform and applications. SIAM Review **33**, 389-404 (1991)
76. Shi, J., Zhang, N.-T., Liu, X.-P.: A novel fractional wavelet transform and its applications. Sci. China Inf. Sci. **55**(6), 1270-1279 (2012)
77. De Bie, H.: Fourier transform and related integral transforms in superspace. (2008) http://www.arxiv.org/abs/0805.1918
78. Fan, H.-Y., Hu, L.-Y.: Optical transformation from chirplet to fractional Fourier transformation kernel (2009) http://www.arxiv.org/abs/0902.1800
79. Klappenecker, A., Roetteler, M.: Engineering Functional Quantum Algorithms (2002) http://www.arxiv.org/abs/quant-ph/0208130
80. Sejdić, E., Djurović, I., Stanković, L.J.: Fractional Fourier transform as a signal processing tool: An overview of recent developments. Signal Processing. 91(6), 1351-1369 (2011)
81. Pégard, N.C., Fleischer, J.W.: Optimizing holographic data storage using a fractional Fourier transform. Opt. Lett. 36, 2551-2553 (2011)
82. Ding, J.J.: Time frequency analysis and wavelet transform class note, the Department of Electrical Engineering, National Taiwan University (NTU), Taipei, Taiwan (2007)
83. Wikipedia. https://en.wikipedia.org/wiki/Wavelet_transform

84. Meyer, Y.: Wavelets and Operators. Cambridge: Cambridge University Press. (1992)
85. Chui, C.K.: *An Introduction to Wavelets*. San Diego: Academic Press. (1992)
86. Akansu, A.N., Haddad, R.A.: Multiresolution Signal Decomposition: Transforms, Subbands, Wavelets. San Diego: Academic Press. (1992)
87. Wikipedia. https://en.wikipedia.org/wiki/JPEG_2000
88. Malmurugan, N., Shanmugam, A., Jayaraman, S., Chander, V.V.D.: A New and Novel Image Compression Algorithm Using Wavelet Footprints. Academic Open Internet Journal. 14 (2005)
89. Ho, T.W., Jeoti, V.: A wavelet footprints-based compression scheme for ECG signals. *IEEE Region 10 Conference TENCON 2004*. **A**., 283. (2004) doi:10.1109/TENCON.2004.1414412
90. Krantz, S.G.: A Panorama of Harmonic Analysis. Mathematical Association of America. (1999)
91. Drozdov, A.: Comparison of wavelet transform and fourier transform applied to analysis of non-stationary processes. Nanosystems: physics, chemistry, mathematics **5**: 363–373. (2014)
92. Martin, E.: Novel method for stride length estimation with body area network accelerometers. IEEE BioWireless 2011, Univ. of California, Berkeley, CA, USA, 79-82 (2011)
93. Liu, J.: Shannon wavelet spectrum analysis on truncated vibration signals for machine incipient fault detection. Measurement Science and Technology **23** (5): 1–11 (2012) doi:10.1088/0957-0233/23/5/055604
94. Akansu, A.N., Serdijn, W.A., Selesnick, I.W.: Emerging applications of wavelets: A review. Physical Communication **3**: 1. (2010) doi:10.1016/j.phycom.2009.07.001
95. Donoho, D.L.: De-noising by soft-thresholding. IEEE Trans. Inform. Theory. **41**(3), 613-627 (1995)
96. Donoho, D.L., Johnstone, I.M.: Adapting to unknown smoothness via wavelet shrinkage. Journal of the American Statistical Assoc. **90**(432), 1200-1224 (1995)
97. Donoho, D.L., Johnstone, I.M.: Ideal spatial adaptation by wavelet shrinkage. Biometrika. 81, 425-455 (1994)
98. Daubechies, I.: Ten Lectures on Wavelets. SIAM, Philadelphia, PA (1992)
99. Daubechies, I.: Different Perspectives on Wavelets. Proceedings of Symposia in Applied Mathematics, vol. 47, American Mathematical Society, USA (1993)
100. Mallat, S.G.: A theory for multiresolution signal decomposition: The wavelet representation. IEEE Trans. Pattern Anal. Machine Intell. **11**(7), 674–693 (1989)
101. Mallat, S.G.: Multiresolution approximations and wavelet orthonormal bases of L2 (R). Transactions of the American Mathematical Society, **315**(1), 69-87 (1989)
102. Zhang, X.-P., Desai, M.: Nonlinear adaptive noise suppression based on wavelet transform. Proceedings of the ICASSP98, vol. 3, 1589-1592, Seattle (1998)
103. Zhang, X.-P.: Thresholding Neural Network for Adaptive Noise reduction. IEEE Transactions on Neural Networks. **12**(3), 567-584 (2001)
104. Zhang, X.-P., Desai, M.: Adaptive Denoising Based On SURE Risk. IEEE Signal Proc. Letters. **5**(10), 265-267 (1998)
105. Zhang, X.-P., Luo, Z.Q.: A new time-scale adaptive denoising method based on wavelet shrinkage. Proceedings of the ICASSP99, Phoenix, AZ., March 15-19 (1999)
106. Lang, M., Guo, H., Odegard, J., Burrus, C., Wells, R.: Noise reduction using an undecimated discrete wavelet transform. IEEE Signal Proc. Letters, **3**(1), 10-12 (1996)
107. Chipman, H., Kolaczyk, E., McCulloch, R.: Adaptive Bayesian wavelet shrinkage. J. Amer. Statist. Assoc. 92, 1413–1421 (1997)
108. Chang, S.G., Yu, B., Vetterli, M.: Spatially adaptive wavelet thresholding with context modeling for image denoising. IEEE Trans. Image Processing. **9**(9), 1522–1531 (2000)
109. Chang, S.G., Yu, B., Vetterli, M.: Adaptive wavelet thresholding for image denoising and compression. IEEE Trans. Image Processing. **9**(9), 1532–1546 (2000)
110. Chang, S.G., Vetterli, M.: Spatial adaptive wavelet thresholding for image denoising. Proc. ICIP. vol. 1, 374–377 (1997)
111. Crouse, M.S., Nowak, R.D., Baraniuk, R.G.: Wavelet-based statistical signal processing using hidden Markov models. IEEE Trans. Signal Processing. **46**(4), 886–902 (1998)
112. Malfait, M., Roose, D.: Wavelet-based image denoising using a Markov random field *a priori* model. IEEE Trans. Image Processing. **6**(4), 549–565 (1997)
113. Mihcak, M.K., Kozintsev, I., Ramchandran, K., Moulin, P.: Low complexity image denoising based on statistical modeling of wavelet coefficients. IEEE Trans. Signal Processing Lett. **6**(12), 300–303 (1999)

114. Simoncelli, E.P.: Bayesian denoising of visual images in the wavelet domain. Bayesian Inference in Wavelet Based Models. New York: Springer-Verlag, pp. 291–308 (1999)
115. Simoncelli, E., Adelson, E.: Noise removal via Bayesian wavelet coring. Proc. ICIP, vol. 1, pp. 379–382 (1996)
116. Belge, M., Kilmer, M.E., Miller, E.L.: Wavelet domain image restoration with adaptive edge-preserving regularization. IEEE Trans. Image Processing. **9**(4), 597–608 (2000)
117. Liu, J., Moulin, P.: Information-theoretic analysis of interscale and intrascale dependencies between image wavelet coefficients. IEEE Trans. Image Processing. **10**(11), 1647–1658 (2000)
118. Guo, H., Odegard, J.E., Lang, M., Gopinath, R.A., Selesnick, I., Burrus, C.S.: Speckle reduction via wavelet shrinkage with application to SAR based ATD/R. Technical Report CML TR94-02, CML, Rice University, Houston (1994)
119. Coifman, R.R., Donoho, D.L.: Translation-invariant de-noising. A. Antoniadis & G. Oppenheim (eds), Lecture Notes in Statistics. Springer-Verlag. vol. 103, pp 125-150 (1995)
120. Misiti, M., Misiti, Y., Oppenheim, G., Poggi, J.M.: Wavelet Toolbox, for use with MATLAB®, User's guide, R2015b (2015) http://www.mathworks.com/help/pdf_doc/wavelet/wavelet_ug.pdf
121. Burrus, C.S., Gopinath, R.A., Guo, H.: Introduction to Wavelets and Wavelet Transforms: A Primer. Prentice Hall, New Jersey (1998)
122. Hubbard, B.B.: The World According to Wavelets: The Story of a Mathematical Technique in the Making. A. K. Peter Wellesley Eds., Massachusetts (1996)
123. Grossman, A., Morlet, J.: Decomposition of Hardy Functions into Square Integrable Wavelets of Constant Shape. SIAM J. App Math. vol. 15, 723-736 (1984)
124. Valens, C.: A really friendly guide to wavelets (2004) http://perso.wanadoo.fr/polyvalens/ clemens/wavelets/wavelets.html
125. Kaiser, G.: A Friendly Guide To Wavelets. Boston: Birkhauser (1994)
126. Walker, J.S.: A Primer on Wavelets and their Scientific Applications. Chapman & Hall/CRC, New York (1999)
127. Stollnitz, E.J., DeRose, T.D., Salesin, D.H.: Wavelets for Computer Graphics: Theory and Applications. Morgan Kaufmann Publishers, San Francisco (1996)
128. Shen, J., Strang, G.: The zeros of the Daubechies polynomials. Proc. American Mathematical Society (1996)
129. Yu, R., Allen, A.R., Watson, J.: An optimal wavelet thresholding for speckle noise reduction. Summer School on Wavelets: Papers, Publisher: Silesian Technical University, Gliwice, Poland. 77-81 (1996)
130. Gao, H.Y., Bruce, A.G.: WaveShrink with firm shrinkage. Statistica Sinica. 7, 855-874 (1997)
131. Gagnon, L., Smaili, F.D.: Speckle noise reduction of airborne SAR images with Symmetric Daubechies Wavelets. SPIE Proc. #2759. 14-24 (1996).
132. Wikipedia. https://en.wikipedia.org/wiki/Uncertainty_principle
133. Wikipedia. https://en.wikipedia.org/wiki/Root_mean_square
134. Mastriani, M.: Quantum spectral analysis: frequency at time with applications to signal and image processing (2017) <hal-01654125>
135. Oppenheim, A.V., Willsky, A.S., and, Nawab, S.H.: Signals and Systems, Second Edition. Upper Saddle River, NJ: Prentice-Hall, Inc. (1997)
136. Briggs, W.L., and, Van Emden H.: The DFT: An Owner's Manual for the Discrete Fourier Transform, SIAM, Philadelphia (1995)
137. Oppenheim, A.V., and, Schafer, R.W.: Discrete-Time Signal Processing, Third Edition. Upper Saddle River, NJ: Prentice-Hall, Inc. (2010)
138. Oppenheim, A.V., and, Schafer, R.W., and, Buck, J.R.: Discrete-Time Signal Processing, Second Edition. Upper Saddle River, NJ: Prentice-Hall, Inc. (1999)
139. Gonorovski, I.S.: Radio circuits and signals. MIR Publishers, Moscow (1986)
140. NVIDIA® Tesla© 2050 GPU. http://www.nvidia.com/
141. https://en.wikipedia.org/wiki/MP3
142. Miano, J.: Compressed image file formats: JPEG, PNG, GIF, XBM, BMP. ACM Press/Addison-Wesley Publishing Co. New York, NY, USA (1999)
143. Wien, M.: Variable Block-Size Transforms for Hybrid Video Coding, Degree Thesis, Institut für Nachrichtentechnik der Rheinisch-Westfälischen Technischen Hchschule Aachen, February 2004.
144. https://en.wikipedia.org/wiki/Video_codec

145. https://en.wikipedia.org/wiki/VP9
146. https://en.wikipedia.org/wiki/VP9#VP10
147. https://en.wikipedia.org/wiki/H.264/MPEG-4_AVC
148. https://en.wikipedia.org/wiki/High_Efficiency_Video_Coding
149. https://en.wikipedia.org/wiki/Field-programmable_gate_array
150. https://en.wikipedia.org/wiki/Firmware
151. https://en.wikipedia.org/wiki/ARM_architecture
152. Cariolaro, G.: Quantum Communications. Springer International Publishing (2015)
153. Mishra, V.K.: An Introduction to Quantum Communication. Momentum Press (2016)
154. Imre, S., Gyongyosi, L.: Advanced Quantum Communications: An Engineering Approach. Wiley-IEEE Press (2012)
155. NIST: Quantum Computing and Communication. CreateSpace Independent Publishing Platform (2014)