Neuronic Convolution Model_Li-Yun Fu

Li-Yun Fu $^{\rm 1}$

¹China University of Petroleum (East China)

October 30, 2023

Abstract

How to represent spatiotemporal information in an artificial neuron model has been a problem of longstanding interest in artificial intelligence. After a brief review of recent advances, Caianiello's neuronic convolutional model is extended in this paper for spatiotemporal information representation. The kernel functions that correspond to the convolutional neuron's receptive field profile can be described by neural wavelets. The convolutional neuron-based multilayer network and its back propagation algorithm are developed to perform spatiotemporal pattern processing. The results provide a natural framework for the discussion of spatiotemporal information representation in an artificial neural network

Neuronic Convolution Model for Spatiotemporal Information Representation and Processing

Li-Yun Fu

Australia Commonwealth Scientific and Industrial Research Organisation (CSIRO) PO Box 1130, Bentley, WA6102, Australia, lfu@dpr.csiro.au

Yanda Li

Tsinghua University Department of Automation, Beijing, China, daulyd@mail.tsinghua.edu.cn

Abstract

How to represent spatiotemporal information in an artificial neuron model has been a problem of longstanding interest in artificial intelligence. After a brief review of recent advances, Caianiello's neuronic convolutional model is extended in this paper for spatiotemporal information representation. The kernel functions that correspond to the convolutional neuron's receptive field profile can be described by neural wavelets. The convolutional neuron-based multilayer network and its back propagation algorithm are developed to perform spatiotemporal pattern processing. The results provide a natural framework for the discussion of spatiotemporal information representation in an artificial neural network.

1 Introduction

A wide variety of patterns from many applications, such as image processing, speech recognition, system identification etc., are described by certain spatiotemporal frequencies. The conventional MP neuron model is a "snap shot representation" where a dynamic pattern is transferred to a static pattern for processing and much spatiotemporal information implied in an input signal is missing due to the dot product of the input with weight function in the MP neuron. Some real-time neuron models have been developed from the MP model by two strategies, one with constant weights and another with time-varying weights. In the first strategy, an additive short-term memory (STM) model is obtained by adding a positive state feedback term to the MP model. STM has been extensively studied in neural modelings [1-2] and applied successfully in artificial neural networks. However, there are some computational problems [3] associated with this method which restrict computational flexibility in the temporal domain.

The second strategy incorporates explicit delays in the MP model, such as Caianiello's neuronic equation [4] which is defined as

$$o_i(t) = f(\sum_{j=1}^N \int_0^t w_{ij}(\tau) o_j(t-\tau) d\tau - \Theta_i(t)), \qquad (1)$$

where the neuron's input, output, and threshold are represented by $o_j(t)$, $o_i(t)$, and $\theta_i(t)$, respectively, f is the neuron activation function, and $w_{ij}(t)$ is the timevarying connection weight with which the firing of the *j*th neuron affects the *i*th neuron after the t time-units. There is ample biological support for the substitution of constant weights w_{ij} by time-varying weights $w_{ij}(\tau)$. The neuron equation (1) represents a neuron with its spatial integration of inputs being a dot-product operation similar to the MP model, but with its temporal integration of inputs being a convolution. Noted that the integral in equation (1) is the Riemann convolution over 0 to t rather than the conventional convolution over $-\infty$ to $+\infty$.

In general, the flexible Caianiello's time-convolution equation needs strong simplifications of the weight kernel for engineering applications [5-7] [3], otherwise the increased dimensionality of the time-varying weights with time destroys the performance of the net. The solution to this problem introduces perceptual aperture problem. The input data to a neural net will be convolution-stacked over a given range called perceptual aperture that is related to the region of the receptive field sensitivity profile. Based on the investigation of visual system, it is a relatively fixed parameter independent of the length of the input signal. That is, the time-varying weights in the Caianiello model should be short convolutional operators with their length unchanged during the net training. The Caianiello model has been modified with short convolutional operators to construct artificial neural nets for engineering applications [8-9]. Here, we extend this approach for both space- and time-varying information processing. We first develop a neural wavelet representation to describe the amplitudephase characteristics of time- and space-varying weights.

Presented at the International Joint INNS-IEEE Conference on Neural Networks, Washington DC, USA, August 2001. 6 pages' Expanded Abstracts.

The problem is related to the spatiotemporal properties of the receptive field profile [10-11]. The experimental results in vision research have shown that the main spatiotemporal properties of major types of receptive fields in different levels of vertebrates can be described in terms of a family of extended Gabor functions. The spectrum of the function may be zero-phase for spatial frequency so as to assure topologically correct mappings, and minimum-phase for temporal frequency due to the real-time requirements of visual system. We employ the convolutional neurons with the spatiotemporal wavelet representation with limited wavelet apertures to construct multilayer neural nets. Since the forward- and back-propagation procedures of the net involve spatiotemporal convolution and crosscorrelation respectively, it is possible to efficiently implement these operations using Fourier transforms and the corresponding block updating strategies for neural wavelets. We design examples on several data sets of varying quality to test the performance and ability of the net.

2 Information Representation in Early Visual System

For space- and time-varying signals, the topographic mappings actually require neural networks to process their amplitude-phase information with regard to spatiotemporal frequencies. Much remains unknown about how the brain trains itself to process the information. In the visual system, there are many topographic mappings of visual space onto the surface of the visual cortex. The topographic representations must be related to the cell's information transfer function that describes spatiotemporal properties of the so-called receptive field. The information transfer function when distributed appropriately over spatiotemporal frequencies is able to encode an arbitrary visual image. To our understanding, the representation of information in a single neuron is the key to the problem. There have been several papers studying the representation of space- and time-varying patterns based on spatiotemporal filtering in the visual nervous system [12-15]. These papers describe families of motion-sensitive mechanisms. Some properties of visual image motion are most evident in the Fourier domain. Thus the motion information in the visual field may be described in terms of spatial and temporal frequencies. To use such mechanisms for constructing an artificial neuron model for artificial neural networks, the following three problems must be solved.

The first problem is the operation relationship between inputs and weights of a neuron. It determines the computational capability and complexity of an artificial neuron model. The operations with a dot product for space and a convolution for time yield the Caianiello neuron model in which the temporal information of the input is processed and remained in the output but much spatial information loses; the convolutional operations for both space and time are discussed in this paper. The second problem is about the spatiotemporal properties of the receptive field profile (i.e., the amplitude-phase characteristics of the neural wavelet in this paper). Its significance is based on the fact that the output of a neuron depends not solely on the spatiotemporal frequencies of the input signal but rather on the amplitude-phase characteristics of the neuronic weight function. The investigations in vision researches [16-20] have shown that the main spatiotemporal properties of major types of receptive fields in different levels of vertebrates can be described in terms of a family of extended Gabor functions. The phase spectrum of the function may be zero-phase for spatial frequency so as to assure topologically correct mappings, and minimum-phase for temporal frequency due to the real-time requirements of visual system. The artificial neuron with its weight function being the kind of function can selectively response to the different spatiotemporal frequency components of input signals. The third problem is the aperture problem that is related to the region of the receptive field sensitivity profile. The aperture corresponds to the length of the weight function of a visual neuron, and will be referred to as the length of a neural wavelet in this paper. Based on the investigation of visual system, it is a fixed parameter that is independent of the length of the input signal to the neuron and has different values for neurons with different functions. This property determines local interconnections instead of global interconnections among neurons in a neural network.

3 General Neuron Model and Neural Wavelet

As a visual signal is distributed over space and time, any visual neuron must collect together and appropriately combine information at different points in the image at different times. The spatiotemporal integration can be described as: at each point in space and time, the signal is weighted by some coefficient and these values are added together. The weight function specifying these coefficients completely characterizes the neuron. Consequently, a generalized neuron model can be defined as

$$o(\mathbf{r},t) = f(\iint w(\mathbf{r},\mathbf{r}',t)s(\mathbf{r}',t)d\mathbf{r}'dt - \theta(\mathbf{r},t)), \qquad (2)$$

where $\mathbf{r} = (x, y, z)$, and $\mathbf{r}' = (x', y', z')$. In the case of a visual neuron, the weight function $w(\mathbf{r}, \mathbf{r}', t)$ is the neuronic detection and integration operator to the input signal $s(\mathbf{r}', t)$, its spatial components correspond to what is ordinarily called the receptive field sensitivity profile. On the other hand, the weight function $w(\mathbf{r}, \mathbf{r}', t)$ is also the point spread function at \mathbf{r} where the neuron is located. The Fourier transform of the weight function consists of two parts: modulation transfer function and phase transfer function. According to the generalized neuron model of equation (2), certain spectrum structure of the weight and

threshold functions will endow the neuron with the ability to process information.

Based on immense experimental results in vision researches, the main spatiotemporal properties of major types of receptive fields in different levels of vertebrates may be described in terms of a family of extended Gabor functions, that is, the optimal weight function in equation (2) for a visual neuron is a set of Gabor basis functions which can provide a complete and exact representation of an arbitrary spatiotemporal signal. An example of a 3-D Gabor function in the complex form can be expressed as

 $G(x, y, z, t) = A \exp(2\pi i (f_x x + f_y y + f_z z + f_t t))$

$$\times \exp(-\frac{x^2}{\sigma_x^2} - \frac{y^2}{\sigma_y^2} - \frac{z^2}{\sigma_z^2} - \frac{t^2}{\sigma_t^2}) \qquad (3)$$

where A is the maximum amplitude; (f_x, f_y, f_z) represents spatial frequency, f_t temporal frequency; $(\sigma_x, \sigma_y, \sigma_z)$ and σ_t represent spatial and temporal deviations respectively. The shapes of the cosine- and sine-phase versions of the 1-D Gabor function are pictured in Figure 1. Apparently, the form of the Gabor function is a waveform with the central frequency (f_x, f_y, f_z, f_t) , the damping factor $\exp(-x^2/\sigma_x^2 - y^2/\sigma_y^2 - z^2/\sigma_z^2 - t^2/\sigma_t^2)$, and the frequency band width $(1/\sigma_x, 1/\sigma_y, 1/\sigma_z, 1/\sigma_z)$. The waveform is referred to as the neural wavelet. It can describe various properties of receptive fields if the parameters in equation (3) are taken properly. Note that the side lobes of the neural wavelet in Figure 1 attenuate rapidly far from the central frequency. Therefore, the valid length of the weight function (or neural wavelet) should be a limited dimensional set, not depending on the length of a input signal. Undoubtedly, a set of such neurons defined as equation (3), when their neural wavelets are distributed appropriately over spatiotemporal frequencies, are able to encode an arbitrary visual image with topologically correct mapping.



Figure 1: Examples of the one-dimensional Gabor function. Solid curve is the cosine-phase (or even-symmetric) version, and dashed curve is the sine-phase (or odd-symmetric) version.

4 Neuronic Spatiotemporal Convolution Model

From the viewpoint of engineering applications, each neuron in the brain is a filter with the kernel function being a particular spatiotemporal spectrum. It is actually a scanning operator in the 4-D domain to complete spatiotemporal integration of unceasing inputs from the synapses of other neurons. For some vertebrate's primary visual system with spatiotemporal invariance, the generalized neuron model of equation (2) can be simplified as:

$$o(\mathbf{r},t) = f(s(\mathbf{r},t) * w(\mathbf{r},t) - \theta(\mathbf{r},t))$$

= $f(\int_0^{L_r} \int_0^{L_r} s(\mathbf{r}',\tau) w(\mathbf{r}-\mathbf{r}',t-\tau) d\mathbf{r}' d\tau - \theta(\mathbf{r},t)),$ (4)

where * is a spatiotemporal convolution symbol of operation; $o(\mathbf{r}, t)$ is the output of the neuron located at \mathbf{r} at the time t; $s(\mathbf{r}', \tau)$ is the input to the neuron at **r** from the neuron at \mathbf{r}' at the time τ ; $w(\mathbf{r} - \mathbf{r}', t - \tau)$ is the neural wavelet of the neuron at \mathbf{r} , with which the firing of the neuron at **r**' affects the neuron at **r** at the time $t - \tau$; L_r is the space length of the neural wavelet that consists of three lengths along the x-, y- and z-axis respectively, and also denotes the number of the neuron's synapses connected to other neurons; L_t is the time length of the neural wavelet. According to the shape of the Gabor function, L_r and L_t are limited and small, not changing with the length of the input signal. Because of the convolutional interaction among the neurons, each neuron radially connects to other neurons in a neural network, This can be designed as a symmetrical local connection in the network.

The neuron's filtering mechanism, intrinsically, is that its neural wavelets cross-correlate with the inputs from other neurons, and large correlation coefficients denote a good match between the input information and the neuron's filtering property. The neurons with similar spatiotemporal spectrum gather to complete the same task using what are known as population codes. The adaptive changes of the neural wavelet in equation (4) can provide the neuron with immense learning power. For engineering applications, the form of the neural wavelet in equation (3) can be simplified as

$$w(\mathbf{r}_0, \mathbf{r}') = [1 - 2(\pi f_0 r)^2] \exp(-(\pi f_0 r)^2), \quad (5)$$

where the artificial neuron is located at \mathbf{r}_0 ; $r = |\mathbf{r}' - \mathbf{r}_0|$ and

 f_0 is the center frequency. Obviously, the neural wavelet of equation (5) is a zero-phase wavelet, and its spectrum is determined by two independent parameters: the center frequency and the wavelet length. To train a convolution neuron-based network to perform some task, for the first step, one must set each neuron with an initial wavelet with certain special spectrum, which is different from the MP neuron-based network where the initial weights of each

neuron are set with random series. Maybe, such initial wavelets can be viewed as the background issue in biological neurons. Then the wavelet of each neuron is adjusted in such a way that the error between the desired output and the actual output is reduced. Thus, the information is loaded on the background issue. It is worth emphasizing, the center frequency of neural wavelets should be related to the center frequencies of the input and desired output signals, that is, the proper choice of the parameter can speed up the convergence of the network.

5 Convolution Neuron-Based Multilayer Network

A 3-D multilayer neural network can be built using the neurons described in equation (4) and the back propagation algorithm [21]. Each neuron in the network fans out to connect to other neurons. For the convenience of deriving equations, equation (4) is rewritten as

$$o_k(\mathbf{r},t) = f(o_{k-1}(\mathbf{r},t) * w_k(\mathbf{r},t) - \theta_k(\mathbf{r},t)), \quad (6)$$

where subscripts k and k-1 denote the kth and k-1th layers of the network respectively. $w_k(\mathbf{r},t)$ and $\theta_k(\mathbf{r},t)$ are, respectively, the neural wavelet and threshold function of the neuron located at \mathbf{r} in the kth layer. Obviously, the input to the network is a 4-D signal matrix since each neuron receives a time signal sequence each time during training. Let

$$E = \frac{1}{2} \iint e_k^2(\mathbf{r}, t) d\mathbf{r} dt$$
$$= \frac{1}{2} \iint [d(\mathbf{r}, t) - o_k(\mathbf{r}, t)]^2 d\mathbf{r} dt, \quad (7)$$

be the error measure on an input/output pattern in the form of a matrix. To implement a gradient descent in E, the neural wavelets are updated according to

$$\Delta w_{k}(\mathbf{r},t) = \eta(t) \iint \delta_{k}(\mathbf{r}',\tau) o_{k-1}(\mathbf{r}'-\mathbf{r},\tau-t) d\mathbf{r}' d\tau$$
$$= \eta(t) \delta_{k}(\mathbf{r},t) \otimes o_{k-1}(\mathbf{r},t), \qquad (8)$$

where \otimes is a spatiotemporal correlation operation symbol and the backprop error $\delta_k(\mathbf{r}, t)$ is computed for two cases. For an output neuron of the network, we have

$$\delta_k(\mathbf{r},t) = e_k(\mathbf{r},t) f'(net_k(\mathbf{r},t) - \theta_k(\mathbf{r},t)).$$
(9)

If the *k*th layer is not an output layer of the network we use the chain rule to yield

$$\delta_{k}(\mathbf{r},t) = f'(net_{k}(\mathbf{r},t) - \theta_{k}(\mathbf{r},t))[\delta_{k+1}(\mathbf{r},t) \otimes w_{k+1}(\mathbf{r},t)].$$
(10)

Equations (9) and (10) give a recursive procedure for computing the δ 's for all neurons of the network, which are then used to compute the changes for the neural wavelet according to equation (8). Since the forward- and back-propagation procedures of the network involve spatiotemporal convolution and cross-correlation operations respectively, it is possible to efficiently implement these

operations using FFTs and corresponding block updating strategies for neural wavelets.

6 Example

The convolutional neuron-based network can be applied to functional approximation, pattern recognition and classification of space- and time-varying signals. The network, through training, searches a space of solutions to find the optimal set of neural wavelets to best compute the mapping function. In this section, a three-layer (one hidden layer) network is designed for a simple two-class classification problem to illustrate the capacity of the network. Then computational properties of the network are investigated. In general, the network can be extended for a general multi-class classification problem.

One of the two-class signal patterns is the low-frequency signal whose center frequency is lower than a value of about 35 Hz and its target signal is a cosine-phase sequence in time with the center frequency being 35 Hz; another is the high-frequency signal whose center frequency is higher than a value of about 35 Hz and its target signal is a sine-phase sequence in time with the center frequency also being 35 Hz. A total of 8 such training and testing pattern samples (representing 200 ms, sampling rate=2 ms) with different center frequencies are shown in Figure 2, where Figure 2a shows the inputs and Figure 2b shows the corresponding outputs from the trained network. Among them, four pattern samples (Curves 1, 3, 5, and 7) with the center frequencies 20 Hz, 30 Hz, 40 Hz, and 50 Hz respectively (the former two signals representing the low-frequency class and the latter two being the high-frequency class) are used to simultaneously train the classifier, and the other are testing pattern samples. No learning takes place during testing. These results illustrate the performance of the network.

In the above example, if we set several initial neural wavelets with different function types or different frequent spectrums, what will happen to convergence of the algorithm and evolution of neural wavelets? We have carried out many simulations with different neural wavelets or different training sets, which show that the algorithm always converges but with little difference in the speed of convergence. Choosing the center frequency of the neural wavelet close to that of the input signal can improve the speed of convergence.

It is interesting to discuss the selection of the initial neural wavelet (viewed as the neuron's background issue) affects its evolution results (viewed as the neuron's information loaded). The simulations are divided in 4 tests to illustrate the relationship between the neuron's encoding mode and its initial neural wavelet with different function type or frequent spectrum. In these tests we use exactly the same network (the three-layer) and the same training sets. 4 tests are performed and the evolution wavelets after 30 iterations are constructed in Figures 3a~3d, where the upper of each figure shows the initial neural wavelets with Curve 1 for the hidden layer and Curve 2 for the output layer, and the lower correspondingly shows their evolution results. Obviously, the information loaded is expressed as the high-frequency variations of neural wavelets. Figures 3a and 3c have the initial wavelets with the same center frequency and phase but different wavelet function types, however, we look at the nearly similar evolution wavelets after 30 iterations. Figures 3a and 3b have the same function type and phase but different center frequencies, but the evolution wavelets are different. Similarly, in Figures 3c and 3d their initial wavelets with the same amplitude spectrum but different phase spectrum evolve into different results. These tests show that the evolution of a neural wavelet is not dependent on its functional type, but mainly on its amplitude and phase spectrums. This means the network uses frequent-spectrum codes.

7 Conclusions

In this study, we discuss the neuronic convolution model for spatiotemporal information representation. It is actually a 4-D convolutional scanning operator to perform the spatiotemporal integration of unceasing inputs from other neurons. The neural wavelets with a particular amplitudephase spectrum in terms of spatiotemporal frequencies provide the convolution neuron with an information processing ability rather than only a logic unit. The length of the neural wavelet is independent of the length of the input signal. It is biologically fixed to make the connections among neurons local instead of global. The performance of the convolution neuron-based network is tested on several data sets of varying quality. These results illustrate the ability of the network to process spatiotemporal patterns.

Acknowledgments

This paper is based on the material with the research carried out by LYF when he worked at Dept. of Engineering Mechanics, Tsinghua University, Beijing.

References

[1] S. Grossberg, "Contour enhancement, short term memory, and constancies in reverberating neural networks", *Studies in Applied Math.*, *52*, pp. 217-257, 1973.

[2] S. Grossberg, "Competition, decision, and consensus", J. Math. Anal. Appl., 66, pp. 470-493, 1978.

[3] B. de Vries and J.C. Principe, "The Gamma model--a new neural model for temporal processing", *Neural Networks*, **5**, pp. 565-576, 1992.

[4] E.R. Caianniello, "Outline of a theory of thought-processes and thinking machines", *J. Theoret. Biol.*, **2**, pp. 204-235, 1961.

[5] K. Lang, A.H. Waibel, and G.E. Hinton, "A time-delay neural network architecture for isolated word recognition", *Neural Networks*, **3**, pp. 23-44, 1990.

[6] U. Bodenhausen and A. Waibel, "Learning the architecture of neural networks for speech recognition", *IEEE proceedings of the ICASSP*, **I**, pp. 117-124, 1991.

[7] D.W. Tank and J.J. Hopfield, "Concentrating information in time: Analog neural networks with applications to speech recognition problems", *IEEE 1st International Conference on Neural Networks*, **IV**, pp. 455-468, 1987.

[8] L.Y. Fu and Y. Li, "Generalized nonlinear inverse problem solved using neural network methods", *Internat. Conference on Neural Network and Brain*, Expanded Abstracts, pp. 482-485, Beijing, China, 1998.

[9] L.Y. Fu, "Caianiello neural network method for geophysical inverse problems", *in* M. Poulton, Ed., *Computational Neural Networks for Geophysical Data Processing*, Chapter **12**, Elsevier, 2001.

[10] L.Y. Fu, "A neuron filtering model and its neural network for space- and time-varying signal processing", *Third International Conference on Cognitive and Neural systems*, Boston University, Paper Vision B03, 1999.

[11] L.Y. Fu, "A filtering neural network for space- and timevarying signal processing", submitted to *Neural Networks*, 1999.

[12] A.B. Watson, H.B. Barlow, and J.G. Robson, "What does the eye see best?", *Nature*, **302**, pp. 419-422, 1983.

[13] A.B. Watson and A.J. Ahumada, "Model of human visual motion sensing", *J. Opt. Soc. Am. A*, **2**, pp. 322-342, 1985.

[14] E.H. Adelson and J.R. Bergen, "Spatiotemporal energy models for the perception of motion", *J. Opt. Soc. Am. A*, **2**, pp. 284-299, 1985.

[15] D.J. Heeger, "Model for the extraction of image flow", J. Opt. Soc. Am. A, 4, pp. 1455-1471, 1987.

[16] J.G. Daugman, "Two-dimensional spectral analysis of cortical receptive field profiles", *Vision Res.*, **20**, pp. 847-856, 1980.

[17] J.G. Daugman, "Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters", *J. Opt. Soc. Am. A*, 2, pp. 1160-1169, 1985.
[18] S. Marcelja, "Mathematical description of the responses of simple cortical cells", *J. Opt. Soc. Am.*, 70, pp. 1297-1300, 1980.

[19] M.R. Turner, "Texture discrimination by Gabor functions", *Biol. Cybern.*, **55**, pp. 71-82, 1986.

[20] Y.J. Wang, X.L. Qi, G.Z. Yao, and M. Wang, "Neural wave representation in early vision", *Science in China*, *B*, **36**, pp. 677-684, 1993.

[21] D.E. Rumelhart, G.E. Hinton, and R.J. Williams, "Learning representations by error propagation", *in* D.E. Rumelhart and J.L. McClelland (eds.), *Parallel distributed processing* (pp. 318-362). Cambridge, MA: MIT Press, 1986.



Figure 2: Study of the two-class classification. (a) The input signals with different center frequency f_0 's, among which Curves 1 and 3 represent the low-frequency training sets whose desired output is a cosine function with $f_0=35$ Hz, Curves 5 and 7 represent the high-frequency training sets whose desired output is a sine function with $f_0=35$ Hz, and the other are the testing signals. (b) The corresponding outputs. The center frequencies for Curves 1~8 are equal to 20 Hz, 25 Hz, 30 Hz, 35 Hz, 40 Hz, 45 Hz, 50 Hz, and 55 Hz, respectively.



Figure 3: Study of the evolution characteristics of neural wavelets. The upper of each figure shows initial wavelets with Curve 1 (its center frequency is f_1) for the hidden layer and Curve 2 (its center frequency is f_2) for the output layer, and the lower shows the corresponding evolution results. (a) The cosine-type, zero-phase initial wavelets with f_1 =5Hz and f_2 =10Hz. (b) The cosine-type, zero-phase initial wavelets with f_1 =10Hz and f_2 =15Hz. (c) The Gabor-type, zero-phase initial wavelets with f_1 =5Hz and f_2 =10Hz. (d) The Gabor-type, minimum-phase initial wavelets with f_1 =5Hz and f_2 =10Hz.