NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

**IDENTIFICATION OF
PUSH-TO-TALK TRANSMITTERS
USING WAVELETS**

by

Yalçın Payal

December, 1995

Thesis Advisors : Ralph Hippenstiel
Monique P. Fargues

Approved for public release; distribution is unlimited.

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE December 1995	3. REPORT TYPE AND DATES COVERED Master's Thesis

4. TITLE AND SUBTITLE IDENTIFICATION OF PUSH-TO-TALK TRANSMITTERS USING WAVELETS	5. FUNDING NUMBERS
6. AUTHOR(S) Yalçın Payal	
7. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey CA 93943-5000	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES
The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.	12b. DISTRIBUTION CODE

13. ABSTRACT *(maximum 200 words)*
The main objective of this study is to find a wavelet-based, feature extracting algorithm for push-to-talk transmitter identification. A distance-measure algorithm is introduced to classify signals belonging to one of four transmitters. The signals are first preprocessed to put them into a form suitable for wavelet analysis. The preprocessing scheme includes taking the envelopes and differentials. Median filtering is also applied to the outputs of the wavelet transform. The distance algorithm uses local extrema of the wavelet coefficients, and computes the distance between the local extrema of a template and the processed signals. A small distance implies high similarity . A signal from each transmitter is selected as a template. A small distance measure indicates that the signal belongs to the transmitter from which the template originated. The distance algorithm can classify correctly the four different signal sets provided for the research. Even at lower signal-to-noise levels, good identification is achieved.

14. SUBJECT TERMS Wavelet Transform,Local extrema, Euclidean Distance Measure			15. NUMBER OF PAGES 84
			16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

Approved for public release; distribution is unlimited.

IDENTIFICATION OF PUSH-TO-TALK TRANSMITTERS USING WAVELETS

Yalçın Payal
Lieutenant Junior Grade, Turkish Navy
B.S., Turkish Naval Academy, 1989

Submitted in partial fulfillment
of the requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
December 1995

Author: Yalçın Payal
Yalçın Payal

Approved by: Ralph Hippenstiel
Ralph Hippenstiel, Thesis Co-Advisor

Monique P. Fargues, Co-Advisor

Herschel H. Loomis, Jr.
Herschel H. Loomis, Jr., Chairman
Department of Electrical and Computer Engineering

ABSTRACT

The main objective of this study is to find a wavelet-based, feature extracting algorithm for push-to-talk transmitter identification. A distance-measure algorithm is introduced to classify signals belonging to one of four transmitters. The signals are first preprocessed to put them into a form suitable for wavelet analysis. The preprocessing scheme includes taking envelopes and differentials. Median filtering is also applied to the envelopes and the differentials to denoise the data. The preprocessed data takes on a pulse-like shape, which is suitable for wavelet processing. The distance algorithm is applied to the outputs (scales) of the wavelet transform. The distance algorithm uses local extrema of the wavelet coefficients, and computes the distance between the local extrema of a template and the processed signals. A small distance implies high similarity. A signal from each transmitter is selected as a template. The distance measure is computed between any signal of interest and the reference templates. The signals are identified to belong to one of four transmitters according to the distance measure. A small distance measure indicates that the signal belongs to the transmitter from which the template originated. The distance algorithm can classify correctly the four different signal sets provided for the research and, even at lower signal-to-noise levels, good identification is achieved.

TABLE OF CONTENTS

I. INTRODUCTION

Signals produced by different transmitters have different characteristics. The transient response, obtained when the transmitter is turned on or off, is basically a build-up from off-state to on-state, or vice versa. These responses are unique to the specific transmitter and differ even among units of the same make and model. This observation applies only to the turn-on response. Therefore, a technique to extract the uniqueness of the turn-on transients will help to identify the transmitter.

Time-frequency analysis of stationary signals is a well-known subject. The Fourier Transform method is quite efficient for this kind of analysis. However, when the signal of interest is non-stationary, the Fourier Transform method is not appropriate, because it uses as a basis function the complex exponential that expands over infinite time. That is, it lacks the time-evolution of frequencies that is very important for the analysis of transients of short duration.

A sliding time-window was introduced by Gabor (1946) to gain time information from the Fourier Transform method. This modified Fourier Transform method is called the Short-Time Fourier Transform (STFT), and uses a modulated window as a basis function. The STFT looks through the window, and processes just a small portion of the signal and, by sliding the window, processes another small portion of the signal successively. Thus, once a window has been chosen, time-frequency resolution is fixed. The problem with this method is that the signal must be stationary within these small portions, or else the method will have limitations in reflecting the time-evolution of frequencies within the window.

Another technique for transient signal analysis is the Wavelet Transform (WT), which seems more efficient in terms of time evolution of frequencies than the STFT. Basis functions of the WT, unlike the complex exponential of the Fourier Transform, do not have infinite duration. They are nonzero for only a short duration. Accordingly, this "compact support" makes the WT localized, not only in frequency but in time as well. Moreover, wavelets provide the flexibility to choose the particular wavelet function that is appropriate for a specific application. This is possible since there are a large number of compactly supported wavelets that can be used as orthogonal basis functions.

The purpose of this thesis is to investigate the use of the Wavelet Analysis for the classification of transient signals of push-to-talk transmitters, and to find an algorithm to identify transmitters. Chapter II gives a brief introduction to wavelet analysis and median filtering. Chapter III explains the algorithms to identify the push-to-talk transmitters, and Chapter IV shows the results and presents the recommendations for future studies.

II. SIGNAL DECOMPOSITION AND MEDIAN FILTERING

A. CONCEPT OF A BASIS

A linear independent set S of vectors $\{\mathbf{v}_k\}$ forms a basis of the space $\mathbf{V}$ if every element $\mathbf{x} \in \mathbf{V}$ can be represented in the form of a series

$$\mathbf{x} = \sum_k c_k \mathbf{v}_k \, , \tag{2.1}$$

where c_k's are the coefficients.

A set S of vectors are said to be linearly independent if and only if there are numbers $c_1, c_2, ...c_n, ...$, all of which are zeros, such that

$$c_1\mathbf{v}_1 + c_2\mathbf{v}_2 + ... + c_n\mathbf{v}_n + ... = 0 \, .$$

B. INNER PRODUCT SPACES

An inner product space is a vector space $\mathbf{V}$ with an inner product defined on it. If $\mathbf{x} = (x_1, x_2, ..., x_n)$ and $\mathbf{y} = (y_1, y_2, ..., y_n)$ are vectors in $\mathbf{V}$, then the inner product of $\mathbf{x}$ and $\mathbf{y}$ are defined by

$$\langle \mathbf{x}, \mathbf{y} \rangle = \mathbf{x}^T.\mathbf{y}^* = x_1.y_1^* + x_2.y_2^* + ... + x_n.y_n^* \, . \tag{2.2}$$

Thus, an inner product of $\mathbf{x}$ and $\mathbf{y}$ is a mapping from $\mathbf{x}.\mathbf{y}$, cartesian product space, into the scalar product field.

The inner product has the following properties:
Let $\mathbf{x}$, $\mathbf{y}$, $\mathbf{z}$ be vectors in $\mathbf{V}$, and c any real-valued scalar, then

1. $\langle (\mathbf{x} + \mathbf{y}), \mathbf{z} \rangle = \langle \mathbf{x},\mathbf{z} \rangle + \langle \mathbf{y},\mathbf{z} \rangle$.
2. $\langle \mathbf{x},\mathbf{y} \rangle = \langle \mathbf{y},\mathbf{x} \rangle$.
3. $\langle c.\mathbf{x},\mathbf{y} \rangle = c.\langle \mathbf{y},\mathbf{x} \rangle$.
4. $\langle \mathbf{x},\mathbf{x} \rangle \geq 0$ with equality if and only if $\mathbf{x} = 0$.

Functions like $f(x)$, on the interval $0 \leq x \leq X$, can be thought of as a vector with a continuum of components along the whole interval. In this case, the summation can be replaced by integration (Burrus and Gopinath, 1993) to find the length of the vector:

$$\| f \|^2 = \int_0^X | f(x) |^2 \, dx \tag{2.3}$$

Inner products of two functions $f(x)$ and $g(x)$ also can be produced in the same manner:

$$\langle f(x), g(x) \rangle = f^T . g^* = \int_0^X f(x) . g^*(x) \, dx \, ,$$

where $f(x)$ and $g(x)$ are defined between $0 \leq x \leq X$.

C. ORTHONORMAL BASES

Any two vectors are called orthogonal if they are perpendicular to each other. A basis $V_1, V_2, ... V_n$ is called orthonormal if all the elements of $V_1, V_2, ..., V_n$ are orthogonal to each other, and $|| V_i || = 1, i = 1, 2, ..., n$ (Strang, 1976).

$V_1, V_2, ..., V_n$ is said to form an orthonormal basis set if the inner product is

$$\langle \ V_i, V_j \ \rangle = \begin{cases} 0 & \text{when } i \neq j \\ 1 & \text{when } i = j \, . \end{cases}$$

D. FOURIER TRANSFORM

The Fourier Transform of a square integrable function $f(t)$ is defined by

$$F(\omega) = \int_{-\infty}^{\infty} f(t) \, e^{-j \omega t} = \langle f(t), \, e^{j \omega t} \rangle \, , \tag{2.4}$$

which is called the Fourier Analysis Formula. The inverse Fourier Transform is given by

$$f(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(\omega) \, e^{j \omega t} d\omega \, , \tag{2.5}$$

which is the Synthesis Formula.

E. SAMPLING

The Sampling Theorem allows us to convert the analog signal to a discrete form for discrete signal processing. The main idea is to take samples from a signal at such a rate that the sequence of samples uniquely defines the original analog signal.

Let $f_T(t)$ denote the signal obtained by weighting the periodic sequence of delta functions spaced T seconds apart by the sequence of numbers $\{f_T(t)\}$; then, the sampled signal is

$$f_T(t) = \sum_{n=-\infty}^{\infty} f(nT)\,\delta(t-nT). \tag{2.6}$$

F. DISCRETE FOURIER TRANSFORM

The Discrete Fourier Transform of a square summable function $f(n)$, $n = 0, 1, 2, ...N$, is defined by

$$F(k) = \sum_{n=0}^{N-1} f(n)\,e^{-j\frac{2\pi}{N}kn}. \tag{2.7}$$

Its inverse is given by

$$f(n) = \frac{1}{N}\sum_{k=0}^{N-1} F(k)\,e^{j\frac{2\pi}{N}kn}. \tag{2.8}$$

G. SHORT TIME FOURIER TRANSFORM (STFT)

1. Introduction

It can be seen from Equation (2.4) that the Fourier Transform of a signal does not reflect the time-evolution of the frequencies (Vetterli and Kovacevic, 1995). Because it does not show how the frequencies vary with time, it is not possible to extract the transient anomalies in the signal, which is spread over all frequencies.

Gabor (1946) proposed a time-frequency localization technique, which is known as the Short Time Fourier Transform (STFT). In this transform, the signal first is divided into short consecutive segments, and then the Fourier Transform of each segment is computed. The problem with this transform is the poor time resolution if the transient phenomena is shorter than the time window.

2. The Continuous STFT

In order to obtain the STFT of the signal, the standard Fourier Transform is modified by inserting a time window $g(t)$ in Equation (2.4) as follows:

$$F(\omega, \tau) = \int_{-\infty}^{\infty} f(t)\, g(t - \tau)\, e^{-j\omega t}\, dt. \tag{2.9}$$

Equation (2.9) is called the Short Time Fourier Transform (STFT). When the window $g(t)$ is Gaussian, the STFT is called the Gabor Transform.

3. The Discrete STFT

We can obtain the discrete STFT by sampling the continuous STFT, Equation (2.9), on a uniform grid (Akansu and Haddad, 1992) such that

$$\omega = k\omega_0, \quad \tau = n\tau_0 .$$

Thus, we obtain the discrete STFT

$$\begin{aligned} F(k, n) &= \int_{-\infty}^{\infty} f(t)\, g(t - n\tau_0)\, e^{-jk\omega_0 t}\, dt \\ &= \langle f(t), e^{jk\omega_0 t}\, g^*(t - n\tau_0)\rangle . \end{aligned} \tag{2.10}$$

This is the inner product of the signal $f(t)$ and the modulated window $g(t)$. The inner product yields large values when $f(t)$ has sinusoids at or near the frequency $k\omega_0$. The inner product is small at the resolution cell $k\omega_0, n\tau_0$ when $f(t)$ has sinusoids at locations other than $k\omega_0$.

The inner product $F(k, n)$ is calculated for each window location $n\tau_0$. Then the window is shifted to the next location, $(n+1)\,\tau_0$, and $F(k, n)$ is calculated for that resolution cell, $k\omega_0$, $(n+1)\,\tau_0$. This procedure generates a two-parameter family $F(k, n)$, which can be plotted on a time-frequency grid as shown in Figure 2.1 (Akansu and Haddad, 1992).

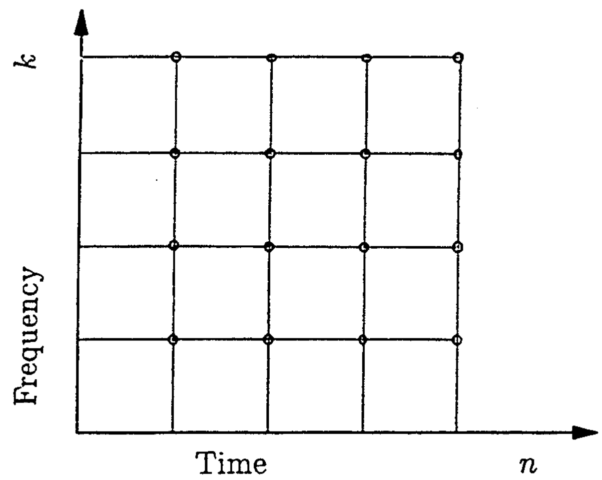


Figure 2.1. Time-frequency grid for the discrete STFT

The STFT, Equation (2.9) can also be interpreted as the convolution of $\tilde{g}(\mathrm{t}) \triangleq g(-\mathrm{t})$ with the modulated signal $e^{-j\omega t} f(t)$ such that

$$F(\omega, t) = \tilde{g}(t) * e^{-j\omega t} f(t), \tag{2.11}$$

where $\tilde{g}(t)$ is the time reversal of the window $g(t)$ and "$*$" denotes the convolution operation. The convolution can be shown schematically as in Figure 2.2.

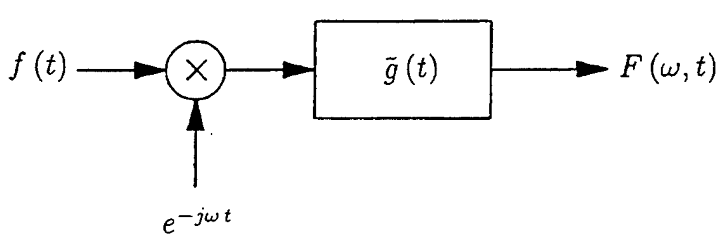


Figure 2.2. Convolution to obtain STFT

If ω_0 in Figure 2.2 is discretized, the modulated filter bank is obtained. If the output of Figure 2.2 is also sampled at $t = n\tau_0$, the corresponding discrete STFT is obtained, as shown in Figure 2.3.

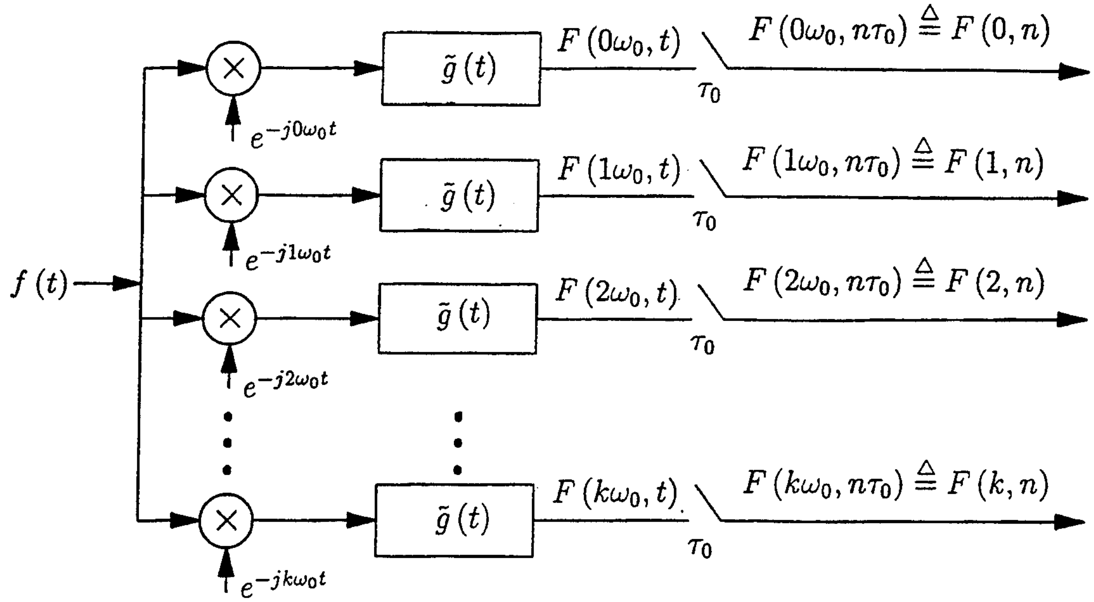


Figure 2.3. The discrete STFT as the modulated filter bank

H. WAVELET TRANSFORM (WT)

The Wavelet Transform (WT) is founded on a set of specific basis functions, which are called "little waves" or wavelets (Young, 1993). They include short duration/high frequency and long duration/low frequency functions. Each element in the wavelet set is constructed from the same function, which is called the "analyzing wavelet" or the "mother wavelet."

1. Continuous Wavelet Transform (CWT)

The Wavelet Transform of a function $f(t)$, with respect to the mother wavelet $\psi(t)$ is given by

$$W_f(a,b) = \frac{1}{\sqrt{a}} \int_{-\infty}^{\infty} f(t)\, \psi^*(\frac{t-b}{a})\, dt, \quad a \in R^+, \quad b \in R \qquad (2.12)$$
$$= \langle f(t), \frac{1}{\sqrt{a}}\, \psi(\frac{t-b}{a}) \rangle,$$

where superscript "*" denotes a complex conjugate, and a and b are scaling and translation parameters, respectively. The normalization term $\frac{1}{\sqrt{a}}$ assures that the energy of the scaled mother wavelet is the same as that of the mother wavelet.

There are three conditions for a function to be a mother wavelet. It must be oscillatory, it must decay to zero, and it must integrate to zero (Young, 1993).

As can be seen from the Equation (2.12), the wavelet family incorporates shifted and scaled versions of the mother wavelet, $\psi(t)$. If we define the wavelet family as $\{\psi_{ab}(t)\}_{a,b}$, where

$$\psi_{ab}(t) = \frac{1}{\sqrt{a}}\,\psi\left(\frac{t-b}{a}\right), \tag{2.13}$$

then this enables us to rewrite Equation (2.12) as

$$\begin{aligned} W_f(a,b) &= \int_{-\infty}^{\infty} f(t)\,\psi_{ab}^{*}(t)\,dt \\ &= \langle f(t),\, \psi_{ab}(t)\rangle\,. \end{aligned} \tag{2.14}$$

The Continuous Wavelet Transform is invertible, provided that the mother wavelet $\psi(t)$ satisfies the admissibility condition (Young, 1993). A wavelet is admissible if the admissibility constant, C_ψ, satisfies the following inequality

$$C_\psi = \int_0^{\infty} \frac{|\,\Psi(\omega)\,|^2}{\omega}\,d\omega < \infty\,, \tag{2.15}$$

where $\Psi(\omega)$ is the Fourier Transform of the wavelet $\psi(t)$.

If the admissibility condition is met for the wavelet function, the inverse CWT exists, and is given by

$$f(t) = \frac{1}{C_\psi}\int_{-\infty}^{\infty}\int_0^{\infty} \frac{1}{a^2}\,W_f(a,b)\,\psi_{ab}(t)\,da db\,. \tag{2.16}$$

2. Discrete Wavelet Transform (DWT)

The Discrete Wavelet Transform (DWT) can be obtained by sampling a, b in Equation (2.13) at

$$a = a_0^{-j}\,,\; b = k b_0 a_0^{-j}. \tag{2.17}$$

Then the wavelet family takes the form

$$\psi_{jk}(t) = a_0^{j/2}\ \psi\left(a_0^j t\ -\ kb_0\right), \tag{2.18}$$

where $j, k \in \mathbf{Z}$. If this wavelet family is complete in $L^2(\mathbf{R})$ for some choice of $\psi(t)$, a, and b, then the set $\{\psi_{jk}\}$ are called frames (Vetterli and Kovacevic, 1995). Now $f(t)$ in $L^2(\mathbf{R})$ can be expressed as

$$f(t) = \sum_j \sum_k d_{jk}\ \psi_{jk}(t), \tag{2.19}$$

where the wavelet coefficient d_{jk} is the inner product

$$\begin{aligned} d_{jk} &= \langle f(t),\ \psi_{jk}(t)\rangle \\ &= a_0^{j/2}\int_{-\infty}^{\infty} f(t)\,\psi^*\left(a_0^j t - kb_0\right). \end{aligned} \tag{2.20}$$

Since the sets are frames not bases, they do not satisfy the Parseval's Theorem. Also the expansion, Equation (2.19) is not a unique use of frames.

A family of wavelet functions, $\{\psi_{jk}\}$ is said to be a frame if there exists $0 < A \leq B < \infty$, such that, for all $f(t)$ in $L^2(\mathbf{R})$,

$$A \parallel f(t) \parallel^2 \leq \sum_j \sum_k \parallel \langle \psi_{jk}, f(t)\rangle \parallel^2 \leq B \parallel f(t) \parallel^2, \tag{2.21}$$

where A and B are frame bounds (Vetterli and Kovacevic, 1995).

When the two frame bounds are equal, the frame is called a tight frame. If ψ_{jk} is a tight frame, with a frame bound $A = 1$, and if $\parallel \psi_{jk} \parallel^2 = 1$ for all j, k, then ψ_{jk} forms an orthonormal basis (Vetterli and Kovacevic, 1995).

Since ψ_{jk} is a tight frame, Equation (2.21) becomes

$$\sum_j \sum_k \parallel \langle \psi_{jk}, f(t)\rangle \parallel^2 = A \parallel f(t) \parallel^2. \tag{2.22}$$

The reconstruction formula can be derived from the Equations (2.19), (2.20), and (2.22) as

$$f(t) = \frac{1}{A}\sum_j \sum_k \langle \psi_{jk}, f(t)\rangle \psi_{jk}. \tag{2.23}$$

When $A = 1$, Equation (2.23) is equivalent to Equation (2.19).

Sampling a and b in CWT, Equation (2.13), as stated in Equations (2.17) and (2.18), makes the translation steps, $b = kb_0a_0^{-j}$, proportional to the time scaling, $a = a_0^{-j}$. Accordingly, large scale values correspond to good time resolution, whereas small scale values correspond to poor time resolution.

The uncertainty principle states that, for a given transform pair, $f(t) \leftrightarrow F(\omega)$, the inequality

$$\sigma_t\sigma_\omega \geq \frac{1}{2} \tag{2.24}$$

holds, where σ_t and σ_ω are defined as follows (Akansu and Haddad, 1992):

$$\sigma_t^2 = \frac{\int t^2 \mid f(t) \mid^2 dt}{\int \mid f(t) \mid^2 dt}, \; \sigma_\omega^2 = \frac{\int \omega^2 \mid F(\omega) \mid^2 d\omega}{\int \mid F(\omega) \mid^2 d\omega},$$

where the integration equation limits $\pm\infty$ are suppressed.

As seen in Equation (2.24), there is a lower bound for the product $\sigma_t \sigma_\omega$; thus, an enhancement in one domain results in degradation in the other domain. Figure 2.4 shows the time-frequency tiling for the WT.

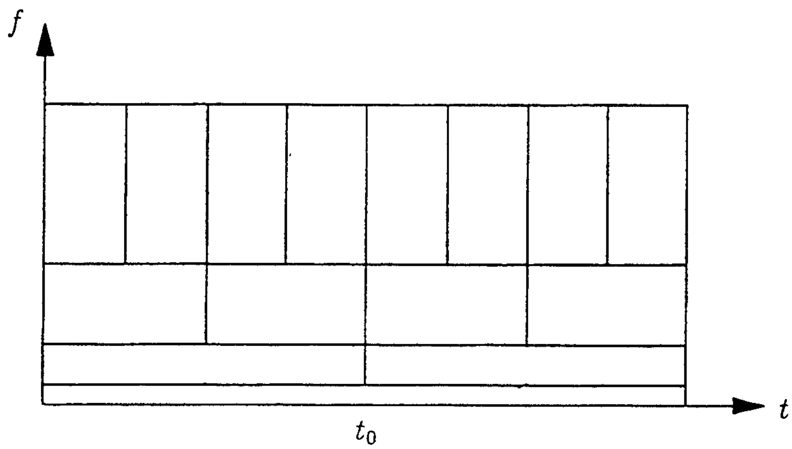


Figure 2.4. Time-frequency tiling for the WT

3. Multiresolution Analysis

a. Multiresolution Spaces

A multiresolution analysis consists of a sequence of closed subspaces

$$...\mathbf{V}_{-2} \subset \mathbf{V}_{-1} \subset \mathbf{V}_0 \subset \mathbf{V}_1 \subset \mathbf{V}_2... \tag{2.25}$$

such that it has the following properties (Vetterli and Kovacevic, 1995):

(a) Completeness:

$$\cap \mathbf{V}_j = \{0\}, \quad \cup \mathbf{V}_j = \mathbf{L}^2(\mathbf{R}) \quad \mathrm{j} \in \mathbf{Z} \tag{2.26}$$

(b) Scaling Property:

$$f(t) \in \mathbf{V}_j \leftrightarrow f(2t) \in \mathbf{V}_{j+1} \tag{2.27}$$

$$\text{for any function} \qquad f \in \mathbf{L}^2(\mathbf{R})$$

(c) Existence of Basis:

There exists a scaling function $\phi \in \mathbf{V}_j$, such that

$$\{\phi_{jk}(t)\} = 2^{j/2}\, \phi(2^j t - k) \;\; \forall\, j \in \mathbf{Z} \tag{2.28}$$

is an orthonormal basis for $\mathbf{V}_j$.

Equation (2.25) is illustrated via a Venn diagram in Figure 2.5. The basic idea in multiresolution analysis is successive approximations. As can be seen from Figure 2.5, a signal in $\mathbf{L}^2$ $(\mathbf{R})$ can be approximated by a coarse representation $\mathbf{V}_j$, which is spanned by ϕ_{jk}. The difference between the original signal and its approximation is the detail. This allows the vector space $\mathbf{V}_j$ to be divided into two parts such as

$$\begin{aligned} &\mathbf{V}_j = \mathbf{V}_{j-1} \oplus \mathbf{W}_{j-1} \\ &\mathbf{V}_{j-1} \perp \mathbf{W}_{j-1}, \end{aligned} \tag{2.29}$$

where $\mathbf{W}_{j-1}$ is defined as the orthogonal complement of $\mathbf{V}_{j-1}$ in $\mathbf{V}_j$, and it provides the detail information to go from subspace $\mathbf{V}_{j-1}$ to $\mathbf{V}_j$. The symbol "$\oplus$" stands for direct sum, and it means that each element of $\mathbf{V}_j$ can be written as the sum of the elements of $\mathbf{W}_{j-1}$ and the elements of $\mathbf{V}_{j-1}$.

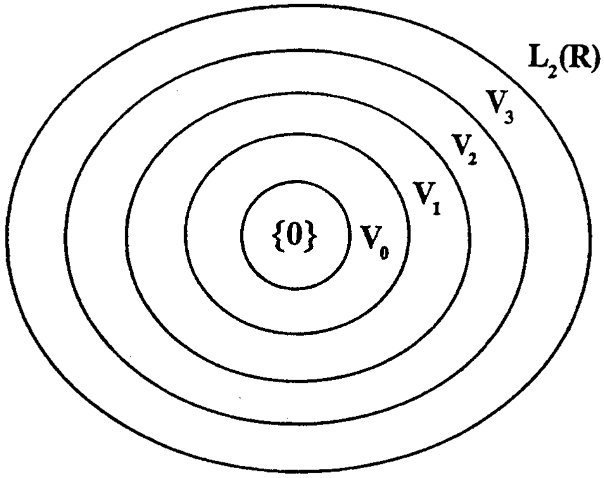


Figure 2.5. Equation 2.25 illustrated via a Venn diagram

$\mathbf{L}^2$ ($\mathbf{R}$) also can be represented by the direct sum as

$$L^2(\mathrm{R}) = \mathrm{V}_{j_0} \oplus \sum_{j \geq j_0} W_j\,, \tag{2.30}$$

as illustrated in Figure 2.6.

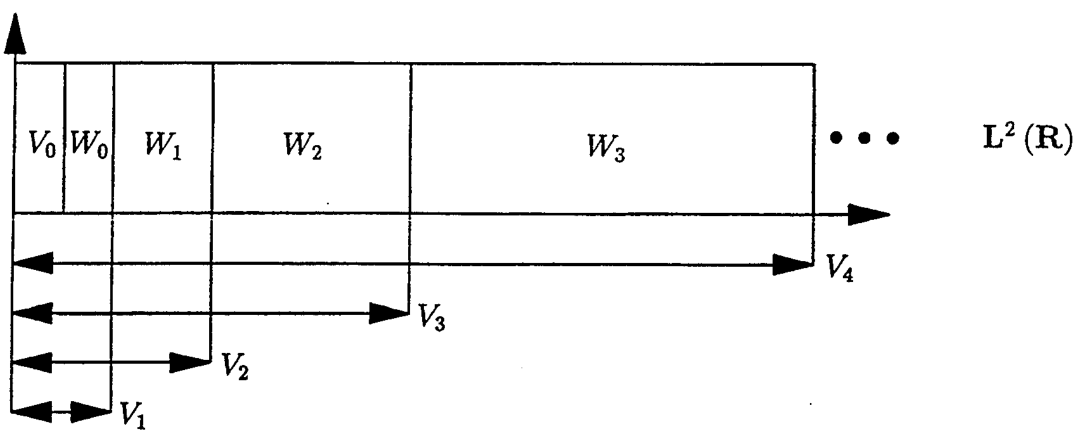


Figure 2.6. Multiresolution representation of $\mathbf{L}^2$ ($\mathbf{R}$)

b. Scaling and Wavelet Functions

Equation (2.28) gives the scaling function, ϕ_{jk}, which spans the coarse approximation space $\mathbf{V}_j$, and Equation (2.18) gives the wavelet function, ψ_{jk}, which spans the difference information space $\mathbf{W}_j$. Together, they can represent any signal in $\mathbf{V}_{j+1}$. It is also possible to interpret $\mathbf{V}_j$ and $\mathbf{W}_j$ as lowpass and highpass portions of the space $\mathbf{V}_{j+1}$.

Equations (2.18) and (2.28) imply that both ψ_{jk} and ϕ_{jk} are sets of translates of dilated functions of ψ_j and ϕ_j, respectively. In Equation (2.28), j is a scale factor. When $j > 0, \phi_{jk}$ narrows, $\mathbf{V}_j$ gets wider accordingly, which results in finer detail. If $j < 0, \phi_{jk}$ stretches, $\mathbf{V}_j$ gets narrower, and ends up with coarser resolution.

Let $\phi \subset \mathbf{V}_0$ be a scaling function. The translates of the dilated $\phi(t)$ span $\mathbf{V}_0$. Thus, the scaling function $\phi(t)$ can be represented as a linear combination of translations of $\phi(2t)$ as

$$\phi(t) = \sum_k h_0(k)\, \phi(2t - k),\ k \in \mathbf{Z}, \tag{2.31}$$

where the $h_0(k)$'s form a sequence of real or complex numbers called scaling function coefficients.

Let $\psi(t)$ be an element of subspace $\mathbf{W}_0$. The subspace $\mathbf{W}_0$ itself is a subspace of $\mathbf{V}_1$ spanned by $\phi(2t)$. Therefore, $\psi(t)$ can be written as a linear combination of translates of $\phi(2t)$:

$$\psi(t) = \sum_k h_1(k)\, \phi(2t - k),\ k \in \mathbf{Z}, \tag{2.32}$$

where the $h_1(k)$'s are wavelet coefficients.

It should be noted that since $\phi(t)$ and $\psi(t)$ span two orthogonal spaces, $h_0(k)$ and $h_1(k)$ also should be orthogonal. Hence, orthogonality of $\mathbf{V}_0$ and $\mathbf{W}_0$ can be achieved by requiring that

$$\langle h_0(k),\, h_1(k)\rangle = 0\,. \tag{2.33}$$

This allows the wavelet coefficients, $h_1(k)$, to be related to the scaling coefficients, $h_0(k)$, such as

$$h_1(k) = (-1)^k\, h_0(N - 1 - k)\,, \tag{2.34}$$

assuming N is the even number of coefficients. Then, the inner product in Equation (2.33) becomes

$$\begin{aligned}\langle h_0(k), h_1(k)\rangle &= \sum_{k=0}^{N-1} h_0(k)h_1(k) \\ &= \sum_{k=0}^{N-1} h_0(k)(-1)^k h_0(N-1-k) \\ &= 0\,.\end{aligned} \tag{2.35}$$

Therefore, any function $f(t)$ in $\mathbf{L}^2(\mathbf{R})$ can be written as

$$f(t) = \sum_{k=-\infty}^{\infty} a_k \phi_k(t) + \sum_{j=1}^{\infty} \sum_{k=-\infty}^{\infty} d_{jk}\,\psi_{jk}(t)\,, \tag{2.36}$$

where a_k and d_{jk} are the Discrete Wavelet Transform Coefficients. They are defined as

$$\begin{aligned} a_k &= \langle f(t), \phi_k(t)\rangle = \int_{-\infty}^{\infty} f(t)\,\phi_k(t)\,dt \\ d_{jk} &= \langle f(t), \psi_{jk}(t)\rangle = \int_{-\infty}^{\infty} f(t)\,\psi_{jk}(t)\,dt\,, \end{aligned} \tag{2.37}$$

where $\phi_k(t)$ and $\psi_{jk}(t)$ are real functions.

Equation (2.36) states that any function in $\mathbf{L}^2(\mathbf{R})$ can be written as a linear combination of the scaling function at a fixed scale plus a linear combination of wavelet functions at higher scales.

The scaling coefficients $h_0(k)$ should be chosen carefully to ensure that the resulting scaling function has compact support. For that reason, they must satisfy some necessary conditions given below (Newland, 1992). The development of these conditions are provided in Appendix A.

(a) The sum of the scaling coefficients should be equal to two.

$$\sum_k h_0(k) = 2\,. \tag{2.38}$$

(b) If a solution to Equation (2.32) exists, $\int \phi(t)\,dt = 0$, and $\int |\phi(t)|^2 dt = 1$, and an integer number of translations of $\phi(t)$ are orthogonal to each other, as defined by

$$\langle \phi(t), \phi_k(t)\rangle = \int \phi(t)\,\phi^*(t-k)\,dt = \delta(k) = \begin{cases} 1; & k=0 \\ 0; & \text{else} \end{cases}$$

then

$$\sum_n h_0(n)\, h_0(n - 2k) = 2\delta(k)\,. \tag{2.39}$$

(c) The sum of the squares of the coefficients is equal to two. Let $k = 0$; Equation (2.39) then becomes

$$\sum_{n=0}^{N-1} |\, h_0(n)\,|^2 = 2\,. \tag{2.40}$$

(d) The individual sums of the odd and even terms of $h_0(k)$ are equal to one.

$$S_1 + S_2 = 2\,,$$

where N is even and

$$S_1 = \sum_{k=0}^{N/2-1} h_0(2k)\,, \tag{2.41}$$

and

$$S_2 = \sum_{k=0}^{N/2-1} h_0(2k + 1)\,. \tag{2.42}$$

c. Frequency Domain

The DFT of $h_0(k)$ is defined as

$$H(w) = \sum_{k=0}^{N-1} h_0(k) e^{-j\omega k}\,. \tag{2.43}$$

Recall that Equation (2.31) can be interpreted as a convolution between $h_0(k)$ and $\phi(2t)$. In this case, $h_0(k)$ can be viewed as an FIR digital filter impulse response and, accordingly, $H(\omega)$ is the filter's frequency response.

If zero is substituted in frequency ω in Equation (2.43), the d.c. response of the FIR filter is obtained as:

$$H(0) = \sum_k h_0(k). \tag{2.44}$$

The right hand side of Equation (2.44) should be equal to two as the result of the necessary condition stated in Equation (2.38). Thus, the necessary condition, Equation (2.38), is equivalent to requiring the FIR filter's frequency response at d.c. to be two.

If the integer translates of ϕ are orthogonal, then the constraint defined in Equation (2.38) is true, if and only if

$$|H(w)|^2 + |H(w + \pi)|^2 = 4. \tag{2.45}$$

If equation (2.44) is viewed as an FIR filter, $h_0(k)$ is called a quadrature mirror filter impulse response (QMF) (Burrus and Gopinath, 1993).

d. Filter Banks and Discrete Wavelet Transform

Multiresolution analysis can be implemented by using a technique called Multiresolution Pyramid Decomposition or Mallat's algorithm (Vetterli and Kovacevic, 1995).

Let f be a real function in $\mathbf{V}_{j+1}$. Since $\{2^{\frac{j+1}{2}}\,\phi(2^{j+1}t - k)\}$ spans $\mathbf{V}_{j+1}$, f can be represented as

$$f(t) = \sum_k a_{(j+1)k} \;\; 2^{\frac{j+1}{2}}\phi(2^{j+1}t - k) = \sum_k a_{(j+1)k} \;\; \phi_{(j+1)k}(t), \tag{2.46}$$

where

$$a_{(j+1)k} = \langle f(t), \phi_{(j+1)k}(t)\rangle = \int f(t) \;\; 2^{\frac{j+1}{2}}\,\phi(2^{j+1}t - k)dt. \tag{2.47}$$

Next, since $\mathbf{V}_{j+1} = \mathbf{V}_j \oplus \mathbf{W}_j$, f can be expressed as the sum of two functions, one lying in $\mathbf{V}_j$ and the other in the orthogonal complement $\mathbf{W}_j$. Hence,

$$f(t) = \sum_k a_{jk}\phi_{jk}(t) + \sum_k d_{jk}\psi_{jk}(t), \tag{2.48}$$

where the DWT coefficients a_{jk} and d_{jk} are given by

$$\begin{aligned} a_{jk} &= \langle \phi_{jk}, f(t) \rangle \\ d_{jk} &= \langle \psi_{jk}, f(t) \rangle . \end{aligned} \tag{2.49}$$

As mentioned earlier, the first term in Equation (2.48) is the coarse approximation, which is the lowpass portion of $f(t)$, whereas the second term contains the detail information, which is the highpass portion of $f(t)$.

From Equations (2.31) and (2.32), we have

$$\phi(t) = \sum_k h_0(k)\,\phi(2t - k),$$
$$\psi(t) = \sum_k h_1(k)\,\phi(2t - k)$$

$$\begin{aligned} \phi(2^j t - n) &= \sum_k h_0(k) \;\; \phi(2(2^j t - n) - k) \\ &= \sum_k h_0(k) \;\; \phi(2^{j+1} t - 2n - k) \end{aligned} \tag{2.50}$$

Let $m = 2n + k$ or $k = m - 2n$, which leads to

$$\phi(2^j t - n) = \sum_m h_0(m - 2n)\phi(2^{j+1} t - m). \tag{2.51}$$

Now, to obtain the DWT coefficients a_{jk} and d_{jk}, Equation (2.50) is substituted into Equation (2.49). Assuming all the values are real,

$$\begin{aligned} a_{jk} &= \langle \phi_{jk}, f(t) \rangle \\ &= \int_{-\infty}^{\infty} f(t) \;\; 2^{j/2} \;\; \phi(2^j t - k) \\ &= \int_{-\infty}^{\infty} f(t) \;\; 2^{j/2} \;\; \sum_m h_0(m - 2k) \;\; \phi(2^{j+1} t - m) \\ &= \sum_m h_0(m - 2k) \;\; 2^{-1/2} \int_{-\infty}^{\infty} f(t) \;\; 2^{\frac{j+1}{2}} \;\; \phi(2^{j+1} t - m) \\ &= 2^{-1/2} \sum_m h_0(m - 2k) \;\; a_{(j+1)m} . \end{aligned} \tag{2.52}$$

$$d_{jk} = 2^{-1/2} \sum_m h_1 (m - 2k) \; a_{(j+1)m} \, . \tag{2.53}$$

Equations (2.52) and (2.53) are convolutions followed by subsampling. These operations are illustrated in Figure 2.7, where $h_0(k)$ and $h_1(k)$ are given by

$$\begin{aligned} \tilde{h}_0(k) &= h_0(-k) \\ \tilde{h}_1(k) &= h_1(-k) \, . \end{aligned} \tag{2.54}$$

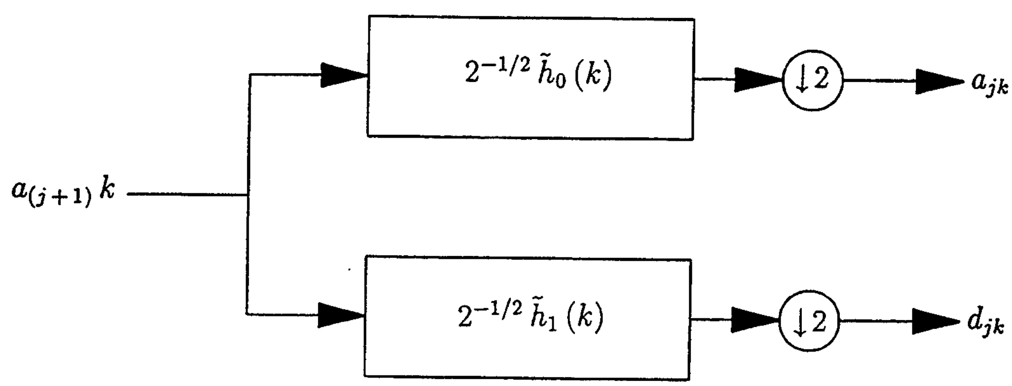


Figure 2.7. One stage of multiresolution signal decomposition.

Therefore, the Discrete Wavelet Transform Coefficients at scale j are obtained by convolving the coefficients at scale $j + 1$ with $h_0(-k)$ and $h_1(-k)$, and then decimating to produce the expansion coefficients at scale j. Figure 2.8 shows the implementation of Equations (2.52) and (2.53) for three levels. The notation LP stands for lowpass FIR filter whose weights are $h_0(-k)$, and HP stands for the highpass FIR filter whose weights are $h_1(-k)$.

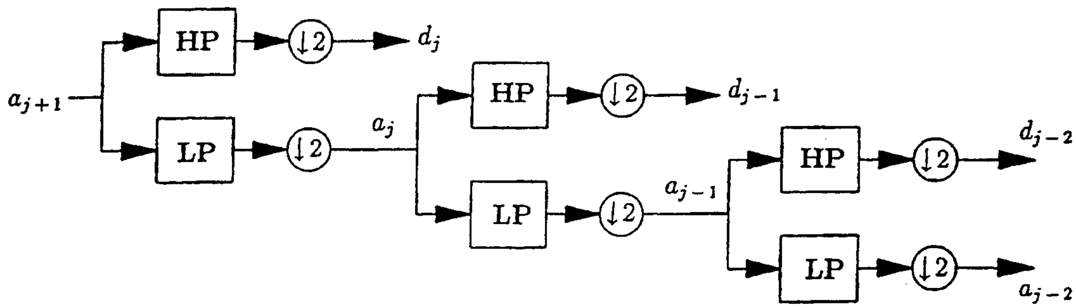


Figure 2.8. Three levels of multiresolution analysis

I. MEDIAN FILTERS

A median filter consists of a sliding window. The center value in the window is replaced by the median of the values within the window. The median of n observations x_i, $i = 1, 2, ..., n$ is given by

$$\operatorname{med}(x_i) = \begin{cases} x_{(v+1)} & , n = 2v+1 \\ \frac{1}{2}\left(x_{(v)} + x_{(v+1)}\right) & , n = 2v \end{cases} , \tag{2.55}$$

where $x_{(i)}$ denotes the center value of n observations (Pitas and Venetsanopoulos, 1990).

A median filter of size $n = 2v + 1$ is defined by the following input-output relation:

$$y_i = \operatorname{med} \ (x_{i-v}, ..., x_i, ..., x_{i+v}) \quad i \in \mathbf{Z},$$

where x_i is the input sequence and y_i is the output sequence. An example of median filtering of size $n = 3$ is shown in Figure 2.9. The input takes on three values: 0, 1, 2. As can be seen in Figure 2.9, the output sequence is 3-valued and does not contain isolated jumps at $n = 2$ and $n = 8$.

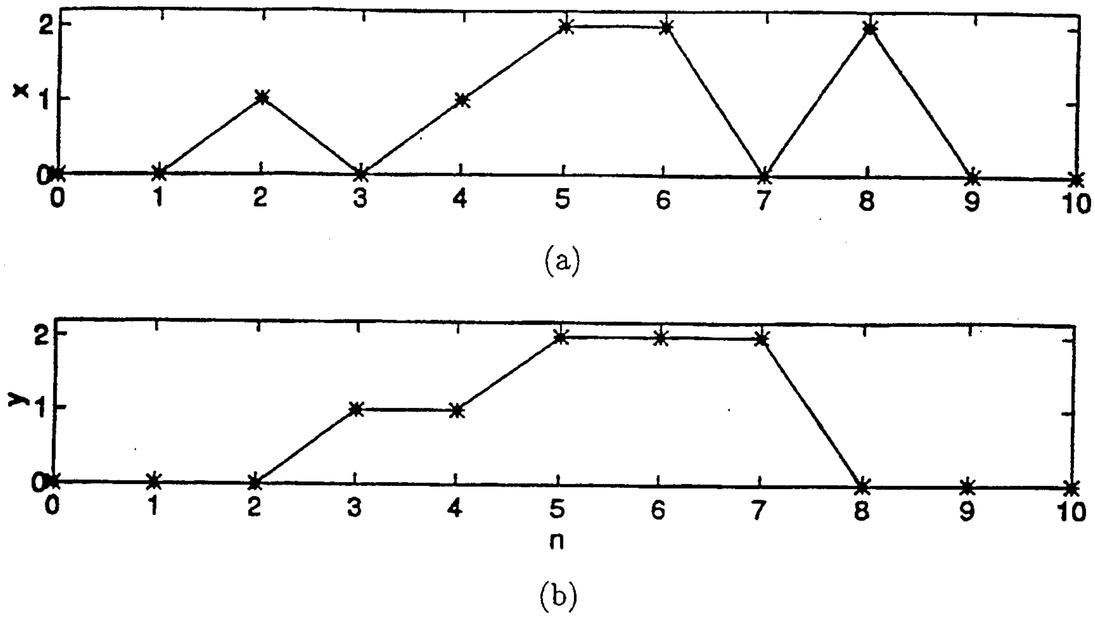


Figure 2.9. (a) Input x to median filter. (b) Output y from median filter.

Median filtering is a nonlinear operation. The main advantage of this kind of nonlinear filtering is its computational simplicity and robustness. Furthermore, it has good edge properties. As will be seen in Chapter III, data processing, wavelet analysis, and signal identification are used to separate the signals according to the transmitter from which they originate. The data processing uses median filtering and extracts the envelope. The wavelet analysis uses a Daubechies wavelet in the Mallat algorithm. The signal identification uses a distance criteria based on selected wavelet scales and wavelet coefficients.

III. WAVELET SIGNAL PROCESSING

A. INTRODUCTION

Mallat's Pyramid Algorithm was used to obtain the Discrete Wavelet Transform (DWT). All programs related to preprocessing and wavelet processing, as well as to signal identification are written in Matlab, a product of Mathworks, Inc. (1992), and are given in Appendix D. The signals used in this analysis do not lend themselves directly to wavelet analysis. Therefore, the processing phase is divided into two parts: a preprocessing phase where the signals are exposed to linear and nonlinear operations, and an identification phase, which will be explained in subsequent sections.

The signals are turn-off/turn-on transients from push-to-talk radios. Nine transmissions from four different transmitters were used in the identification part of this thesis.

B. PREPROCESSING PHASE

1. Turn-On Signals from Push-To-Talk Radios

The signals used in the analysis were collected and recorded by the Naval Security Group Activity, Charleston, SC. Nine turn-on/off samples of each of four transmitters were recorded. All the radios were Motorola models. Each radio is identified by its model name and number and is tabulated below:

Radio:	Model:
Transmitter 1 (Tr1)	Maxtrac
Transmitter 2 (Tr2)	Saber
Transmitter 3 (Tr3)	HT440
Transmitter 4 (Tr4)	Saber

Tr2 and Tr4 are different radios of the same model.

The carrier frequency of the radios was 138.525 MHz. The signals were filtered at 1 Mhz Bandwidth (BW), and then digitized with a sampling frequency of 5 Mhz and a center frequency of 1.075 Mhz.

The recordings from each transmitter are numbered from one to nine. Figure 3.1 shows the first turn-on signal of the four transmitters. It can be noticed that the recordings differ from each other in duration as well as in the type of waveform produced during the transition part. The turn-off response was also examined, but did not lead to any identification clues and was, therefore, not included in the results section.

The information useful for the identification of the transmitter lies in the envelope, which is the low frequency component of the signals. Wavelet Transforms are not useful in analyzing low frequency, but are well-suited for short duration phenomena. Thus, we need to transform the data into a form suitable for wavelet analysis. A preprocessing scheme is applied to the signals to transform them into a usable form, and to improve the signal-to-noise ratio (SNR), as presented in the next section. It should also be noted that the SNR's of the recordings are not known, and a denoising process would definitely enhance the identification performance.

2. Preprocessing Scheme

There are four steps in the preprocessing phase: taking the envelope, median filtering, differentiating, and a final median filtering. Prior to taking the envelopes, the d.c. terms were removed. 100-point median filters were used for filtering the envelope, whereas a 50-point median filter was used after the differentiation. The sizes of the median filters were determined experimentally. The main reason for using median filtering was to avoid broadening the signal, which usually is caused by convolutional operations.

The preprocessed signals for the first recordings from the four transmitters are shown in Figure 3.2. All of the four final pulse-shaped signal waveforms seem to be appropriate for WT analysis. The preprocessing scheme is illustrated step-by-step in Appendix C for the first signal of each transmitter. The final preprocessed pulse-shaped signals for all transmitters are included in Appendix C.

C. IDENTIFICATION PHASE

1. Reduced Set Representation

One of the main drawbacks of the wavelet transform is the shift variance, since the wavelet coefficients of a signal and a shifted replica can be totally different. Therefore, a Euclidean distance measure (introduced by Aware, Inc., 1992) is used as the technique for the classification of signals.

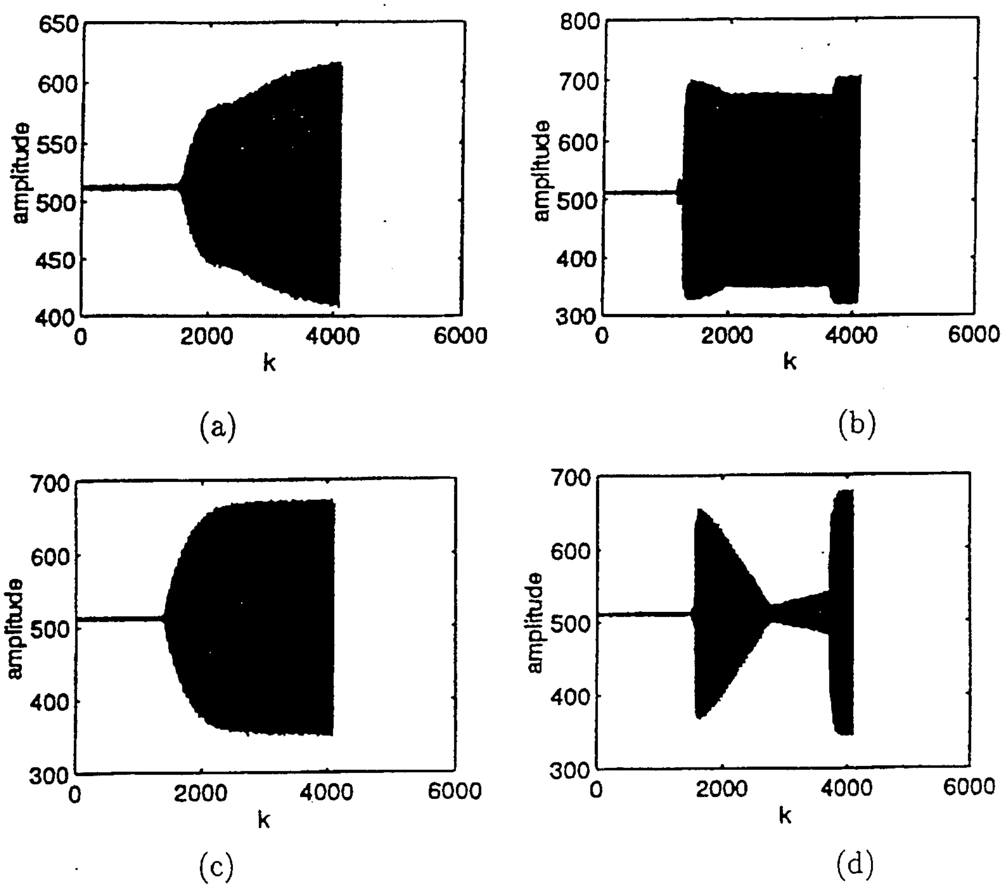


Figure 3.1 Signal samples from four different transmitters

(a) Signal 1 from Transmitter 1 (b) Signal 1 from Transmitter 2
(c) Signal 1 from Transmitter 3 (c) Signal 1 from Transmitter 4

Mallat and Zhong (1989) demonstrated that the maxima extracted from the modulus of the wavelet coefficients can be used to reconstruct the input signal. That is, the maxima of the modulus of the wavelet coefficients contain approximately the same amount of information as the original signal. Consequently, signal analysis can be performed based on the wavelet extremals, which form a reduced signal representation.

Thus, wavelet coefficients at each scale have been replaced by their extremal values. Hence, the reduced set is only nonzero where the scale has an extremal, and is equal to the original value at these locations. Wavelet scale coefficients, which are not extremals, are set to zero.

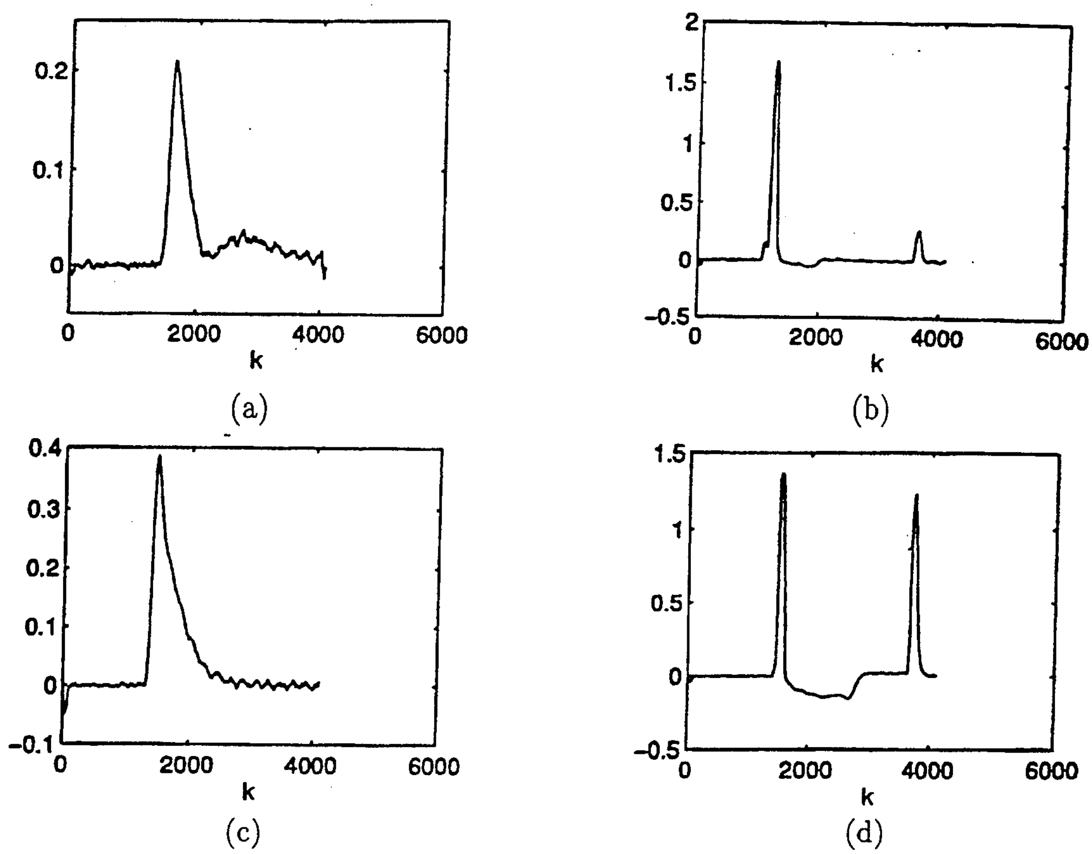


Figure 3.2 Preprocessed signal samples from four different transmitters

(a) Signal 1 from Transmitter 1 (b) Signal 1 from Transmitter 2
(c) Signal 1 from Transmitter 3 (c) Signal 1 from Transmitter 4

2. Ranking/Matching Algorithm

The first step in computing a pairwise distance measure is to rank the peaks of the sets to be compared. That is, the peaks at each scale are ranked by their amplitudes. The second step is to match the ranked peaks to form pairs. The peak with the highest rank in one set is matched to the peak with the highest rank in the other set. The next in rank is matched to that of the other set, and so on. To illustrate the ranking and matching scheme, let us consider $f(k)$ and $g(k)$, as shown in Figure 3.3. We first rank the peaks by amplitude. The amplitudes of the signals in ascending order are -2, -1, 1, 2, 3, 5 for $f(k)$, and -1, 0, 0.5, 1, 2.5, 4 for $g(k)$. Then, we match the peaks by taking corresponding ranked peaks and forming pairs. The peak with the highest rank in $f(k)$ is 5; the peak with the highest rank in

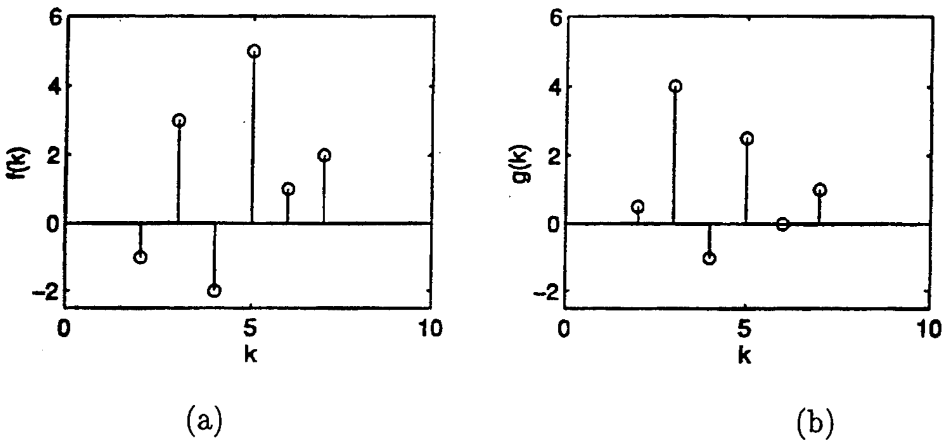


Figure 3.3 Signals to be matched:
(a) maxima of signal $f(k)$ (b) maxima of signal $g(k)$

$f(5) = 5$	$f(3) = 3$	$f(7) = 2$	$f(6) = 1$	$f(1) = -1$	$f(4) = -2$
$g(3) = 4$	$g(5) = 2.5$	$g(7) = 1$	$g(2) = 0.5$	$g(6) = 0$	$g(4) = -1$

Table 3.1 Matched pairs of signals $f(k)$ and $g(k)$
(columns show matched pairs, ranked in decreasing order from left to right)

$g(k)$ is 4. We match these peaks with highest ranks first. Then, we match the next in rank, which are 3 from $f(k)$ and 2.5 from $g(k)$, and so on. The ranked and matched pairs are tabulated in Table 3.1. The matching scheme does not require the number of elements in both sets to be the same. When a peak in one set does not have a corresponding peak in the other, we insert a zero into the set that has a smaller number of peaks, and form the pairs by matching the remaining peaks and the zeros.

3. Distance Measure

The third step is to compute a distance measure for the matched pairs. The distance assigned to the pair is the sum over the Euclidean distances in each scale. Thus,

$$d(a^j, b^j) = \sum_{(k,l)\text{ are locations of the matched peaks}} [W^j_{k,l}(a^j_k - b^j_l)^2]^{\frac{1}{2}} \tag{3.1}$$

where a^j and b^j are the wavelet maxima at scale j, and a_k^j and b_l^j are the values a^j, b^j at temporal locations k and l, respectively, and $\mathbf{W}_{k,l}^j$ is the weighting factor at scale j for the relative distances between the corresponding coordinates of matched peaks. The weighting factor, $\mathbf{W}_{k,l}^j$, is defined as

$$\mathbf{W}_{k,l}^j = | n_k^j - n_l^j | \tag{3.2}$$

where n_k^j and n_l^j are the coordinates of a_k^j and b_l^j, respectively.

The similarity between two sets is described in terms of the sum of the Euclidean distances of amplitudes weighted by the square root of the relative distances between the corresponding coordinates. Basically, a small value $| n_k^j - n_l^j |$ and $| a_k^j - b_l^j |$ imply a high degree of similarity between a^j and b^j.

As can be seen from Equation (3.1), this method provides no penalty if the peaks happen to be at the same location, or the amplitudes of the peaks are equal to each other. The penalty weight, $\mathbf{W}_{k,l}^j$, is determined by the separation of the peaks. Large separation between the matched peaks corresponds to large penalty factors. The distance measure also depends on the amplitude difference of the matched peaks. The distance measure is directly proportional to the difference in amplitude. It should be noted that even identical scale outputs with offset in time (due to signal delay) will have a nonzero weighting factor.

4. Modification to Weighting Factor

We now introduce a lineup procedure and a modified weighting term to offset the shortcoming of the technique, described in Section III.C.3. This is the most robust method for obtaining reliable distance measures, and is used to obtain the experimental results.

There are four different sets of signals. Each set belongs to a different transmitter. The method defined above was used to determine the transmitting source of the signal. For this purpose, a template for each transmitter set is needed to measure the similarity with any signal of interest. The objective is to apply the distance measure algorithm to identify the transmitters.

Matching peaks by ranking preserves the order of the peaks when the signals to be compared are very similar (i.e., identical), which then causes the weighting factor in Equation (3.2) to be a constant denoted by $\mathbf{W}$. For example, if we shift $f(k)$ in

Figure 3.3 four units to the right and compare it to $f(k)$, the weighting factor in $\mathbf{W}$ in Equation (3.2) is 4 for all matched peaks. Since the amplitudes of the matched peaks are the same, the distance measure defined in Equation (3.1) is zero, in spite of the nonzero weighting factor $\mathbf{W}$. This is not the case when we add noise to one or both of the signals, because it will cause amplitude difference between matched peaks. Even though the signals are similar, a nonzero distance measure will be computed. Furthermore, this distance measure will be dominated by the constant weighting factor $\mathbf{W}$. We need to reduce the weighting factor in such cases by eliminating the offset between the signals. To eliminate this problem, the maximum peaks of the two sets are aligned by removing the time shift offset between the signals.

The first signal of the four sets was chosen as a template. To improve the accuracy of the algorithm, every signal was aligned with each template before applying the distance measure, Equation (3.1). Hence, the relative distance between the maximum matched peaks of the signal and templates is zero. Accordingly, the weighting factor, $\mathbf{W}^j_{k,l}$, will be zero for the maximum peaks for all templates. Applying zero weight for the distance with the corresponding (i.e., proper) template makes sense, but not for the other templates. Therefore, the weighting factor is modified to be

$$\mathbf{W}^j_{k,l} = \begin{cases} \mid n^j_k - n^j_l \mid & k \neq l \\ 1 & k = l\,. \end{cases} \tag{3.3}$$

Now, a small distance measure implies more accurately that the signal belongs to the set from which the template came.

5. Application of Distance Measure Algorithm

The fourth, eighth, and sixteenth order Daubechies wavelet functions were used to compute the WT of the preprocessed signals. The distance measure algorithm was applied to measure the similarity between the wavelet coefficients of the signal to be identified and the templates. From experimental considerations, the eighth order Daubechies (Dau-8) wavelet functions gave satisfactory results, whereas Dau-4 and Dau-16 did not. The Dau-8 wavelet function, which was used as the mother wavelet, is shown in Figure 3.4.

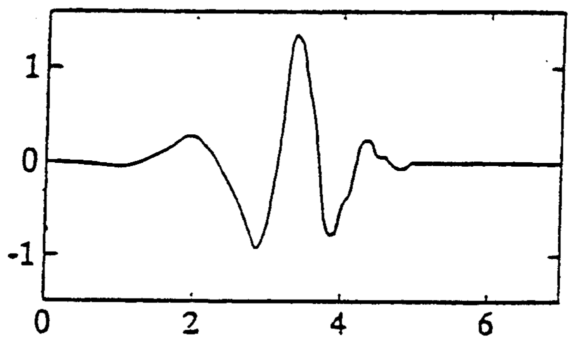


Figure 3.4 Eighth order Daubechies wavelet function

a. Transmitter 1

The distance values at Scale 11 between the first four signals from Transmitter 1 and the templates are tabulated and shown in Table 3.2. As expected, the distance value with Template 1 is significantly less than that with the other templates, since Template 1 is the template for the signals from Transmitter 1, and small distance values imply high similarity.

Scale 11	Template 1	Template 2	Template 3	Template 4
Signal 1	0	44.3073	14.7737	107.8438
Signal 2	3.0527	47.2375	16.3406	103.8587
Signal 3	2.6195	40.5894	13.797	99.7881
Signal 4	1.0806	42.9072	15.4847	105.308

Table 3.2 Distance measures at Scale 11 for Signals 1 – 4 of Transmitter 1

As can be seen from Table 3.2, the distance values, d, between signals from Transmitter 1 and Template 1 are well separated from those of the other templates. Zero value in distance indicates perfect match, and occurs between Signal 1 and Template 1. This is expected since Signal 1 was chosen as a template for the set.

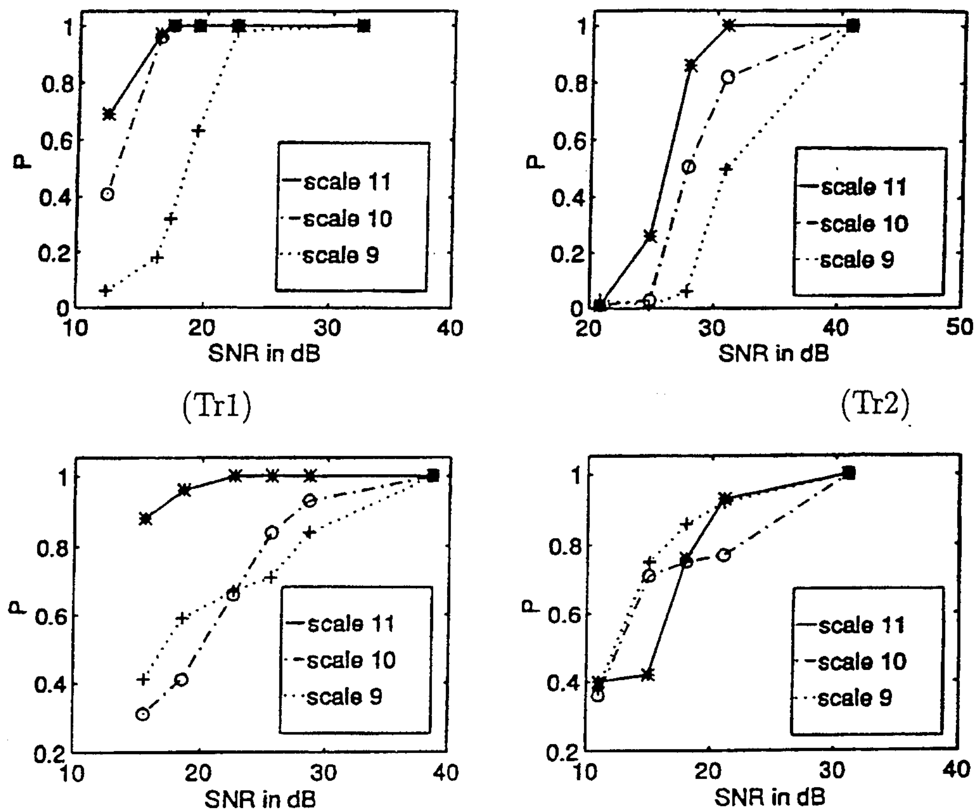
A thresholding technique can be introduced to automate the identification procedure by defining a threshold level for each template. For example, one can compare the maximum value of Set 1 with the minimum values of other templates. Determining the threshold levels is not addressed in this study; however, comparing the minimum and maximum values of the template outputs may be quite useful to evaluate the performance of the identification scheme. The maximum and minimum values quoted in the remainder of Chapter 3 are obtained from Appendix B. At Scale 11 with Template 1, the maximum distance value is 5.3971, whereas the nearest minimum distance value of the other templates is 12.52. This minimum occurs with Template 3, which indicates that Transmitter 1 and Transmitter 3 have a somewhat similar output (i.e., turn-on behavior). The ratio of the minimum Template 3 output to the maximum Template 1 output is 2.319, which shows how well the Template 1 results are separated from the other template results. This separation ratio is 2.381 and 1.66 for Scale 10 and Scale 9, respectively. Lower scales were not useful in terms of the similarity measurement, and were not included.

b. Transmitters 2, 3, and 4

Similarly, the application of the distance measure to the other three sets resulted in the following separation ratios. They show how well the signals of a particular set are separated from the other sets under worst case conditions. They are 3.56, 2.76, and 1.88 for signals from Transmitter 2 at Scales 11, 10, and 9, respectively. The following separation ratios were obtained at Scales 11, 10, and 9: 4.64, 1.98, and 3.73 for signals from Transmitter 3, and 1.75, 1.4, 1.24 for signals from Transmitter 4. Signal versus template output plots for all signal sets are given in Appendix C. For example, one can obtain minima, maxima, and standard deviation of the distance measures very easily. It should be noted that, for Transmitters 1, 2, and 3, nine signals were used while, for Transmitter 4, only six signals were usable.

c. Probability of Identification

Signal-to-Noise Ratios (SNRs) for all signals were estimated. As seen in Figure (3.1), the signal waveforms can be split into three parts. The off region is where there is no signal; the transition region is where there is a build-up from off to on state; and the on region is where the signal is at steady state. We can assume that the off region consists of noise only, allowing the computation of the noise power. Noise and signal coexist in the on region. The signal power can be computed by finding the power in the on region and subtracting the noise power. Hence, we can compute the SNR's for all sets. An average SNR value of 32.38 dB, 40.87 dB, 38.58 dB, and 31 dB was computed for Set 1, Set 2, Set 3, and Set 4, respectively.



(Tr1) (Tr2)

(Tr3) (Tr4)

Figure 3.5 Probability of identification (P_i) of the signals

(a) P_i of signals from Transmitter 1 (b) P_i of signals from Transmitter 2

(c) P_i of signals from Transmitter 3 (c) P_i of signals from Transmitter 4

Gaussian white noise was added to all sets to decrease the SNR levels. We repeated the experiment eleven times for different SNR values, and obtained the probability of identification versus SNR plots for the signals. Results are shown in Figure 3.5. Note that, when the signal of interest is from Transmitter 1 and its SNR is higher than 16 dB, then its identification probability is high, and the distance-based algorithm is reliable. The minimum SNR values required for a reliable classification at Scale 11 are 30, 23, and 30 dB for signals from Transmitter 2, 3, and 4, respectively.

d. Attempts to Reduce the Processing Load

Using a reduced data set (i.e., taking the local extremal in distance computation) reduced the computational complexity. In addition, two approaches were attempted to obtain acceptable performance while minimizing the computational effort. In the first approach, the signal was decimated by using a Chebychev -2 filter after median filtering of the envelope. This reduced the number of points and, therefore, the number of computations. The distance algorithm using the decimated data did not work well enough to classify the signals and was not pursued further. In the second approach, instead of matching all the maxima of the signal of interest and the template, only the highest four peaks from each were matched, and a distance measure was computed. Even the original data did not permit successful classification; hence, noisy versions of the signal were not tried.

Matching of the peaks of the signal and template includes ranking. Therefore, it is highly affected by noise. To minimize the noise effect in the matching of peaks, point-by-point matching was used instead of ranking, but the results were not satisfactory. In summary, distance measures that use all extremals and match them by ranking are the most effective scheme in classifying the four signal sets.

IV. CONCLUSION

A. CLASSIFICATION OF PUSH-TO-TALK COMMUNICATION SIGNALS

The main objective of this study was to determine a wavelet-based algorithm designed to extract features to identify push-to-talk transmitters. Robustness, compact signal representation capability, and low computational complexity are the main advantages of the wavelet analysis, which was used for feature extraction.

Push-to-talk communication recordings provided for this research have a common feature: They all include a transition from the off-to-on state, as well as the on-to-off state. The turn-on transition phase is unique for each transmitter, and can be used for classification purposes. The recordings differ from each other by the waveform in the transition region and the duration. The feature for the classification of the signals is contained in the signal envelopes. The turn-on part of the envelope is a transient and, hence, a wideband signal.

The push-to-talk recordings were not in appropriate forms for WT analysis. Thus, the recordings were preprocessed to become usable for WT processing. Differentiation of the envelope of the signals was used to transform the data into pulse-shaped transients. Median filtering was applied to both the envelope and the differential to improve the signal-to-noise ratio. Median filtering was preferred since it does not broaden the transient features of the signal.

A distance algorithm was introduced in this work. It was based on a Euclidean distance measure between the wavelet coefficients of two data set on a given scale. Decisions about the origin of the signal was made according to the distance measures between the signals and the templates, where each template represented a different transmitter. A small distance value implied that the signal belonged to the same set as that particular template. In its current form, the classification assumes that any signal of interest belongs to one of the four sets.

The distance algorithm was applied to four different signal sets. The first recording in any of the sets was designated to be the template. Instead of using all the wavelet coefficients, just the local extrema were used. Using only edge points by taking the local extrema reduced the computational complexity in the algorithm.

The distance measure between the signal and the templates is the sum of all Euclidean distances between matched peaks of the signal and the templates. It also includes a penalty factor due to the relative square root distance between the matched ranked pairs and their difference in amplitude. Matching signal peaks to the template is performed by ranking. Before matching the pairs, the maximum peak of the signal is aligned with the maximum of the templates. This reduced the penalty weight for like signals that are not aligned in time.

The WT of the signals was computed using Daubechies-4 (Dau-4), Daubechies-8 (Dau-8), and Daubechies-16 (Dau-16) as mother wavelets. Experimentally, it was found that Dau-8 worked best with this data set.

Decimating the signal envelope before taking the differential reduces the number of points and, hence, decreases the number of computations. The distance algorithm was applied to the decimated sequences, but the results did not allow classification of the signals.

We also considered using only the four highest wavelet coefficients at a given scale in the distance computation to reduce the computational complexity. However, simulations showed that the information cited was not sufficient. The distance algorithm described in Section III, using all the extremas, matching the peaks by ranking, and permitting a penalty according to Equation (3.3) allowed robust identification of the signal sets.

B. RECOMMENDATION FOR FUTURE STUDIES

The distance algorithm introduced in Chapter III gave promising results in classifying the four signal sets provided for this research. Three important issues have not been addressed in this research: the template selection, a threshold technique, and incorporation of information from other scales. In this work, templates were chosen arbitrarily from the sets, and the distance algorithm was applied only to the four signal sets. When the signals to be identified are from these sets, the algorithm was capable of classifying the signals. However, if the signals do not belong to these sets, the algorithm will compute a distance to each of the templates, which can lead to misclassification. A thresholding technique and robust template selection should be the subject of further study.

Also, no attempt was made to combine information from several scales. Identification was obtained by using just one scale. Typically, Scale 11 worked best. If the distance information from Scales 9 and 10 could be used, a more robust identification performance should be realized.

APPENDIX A. DEVELOPMENT OF NECESSARY CONDITIONS

NECESSARY CONDITION 1:

The condition given by Equation (2.38) can be derived by integrating both sides of Equation (2.31) (Burrus and Gopinath, 1993).

$$\begin{aligned} \int \phi(t)dt &= \int \sum_k h_0(k)\phi(2t-k)dt \\ &= \sum_k h_0(k) \int \phi(2t-k)dt \,. \end{aligned} \tag{A.1}$$

Let $m = 2t - k$, so $dt = \frac{1}{2}dm$. Substituting into Equation (A.1) gives

$$\begin{aligned} \int \phi(t)dt &= \frac{1}{2}\sum_k h_0(k) \int \phi(m)dm, \\ \frac{\int \phi(t)dt}{\int \phi(m)dm} &= \frac{1}{2}\sum_k h_0(k). \end{aligned} \tag{A.2}$$

The left side of Equation (A.2) is equal to one, and so

$$\frac{1}{2}\sum_k h_0(k) = 1,$$

or

$$\sum_k h_0(k) = 2\,. \tag{A.3}$$

NECESSARY CONDITION 2

ϕ_k forms an orthonormal set for $\mathbf{L}^2(\mathbf{R})$. Therefore, the inner product of $\phi(t)$ with integer translate of itself is

$$\langle \phi(t), \phi_k(t) \rangle = \int \phi(t)\phi(t-k) = \delta(k). \tag{A.4}$$

Substituting Equation (2.31) into (A.4) gives

$$\int \sum_n \sum_m h_0(n) h_0(m) \phi(2t-n)\phi(2(t-k)-m)dt = \delta(k),$$

$$\sum_n \sum_m h_0(n) h_0(m) \int \phi(2t-n)\phi(2(t-k)-m)dt = \delta(k).$$

Let $2t = \tau$, so $dt = \frac{d\tau}{2}$ and

$$\sum_n \sum_m h_0(n) h_0(m) \frac{1}{2} \int \phi(\tau - n) \phi(\tau - 2k - m) d\tau = \delta(k).$$

Since the integral is equal to $\delta(n - m - 2k)$ from (A.4), then

$$\frac{1}{2} \sum_n \sum_m h_0(n) h_0(m) \delta(n - m - 2k) = \delta(k). \tag{A.5}$$

Let $n = m + 2k$; then Equation (A.5) becomes

$$\sum_n h_0(n) h_0(n - 2k) = 2\delta(k). \tag{A.6}$$

NECESSARY CONDITION 3

Let $k = 0$; then Equation (A.6) becomes

$$\begin{aligned} \sum_n h_0(n) h_0(n) &= 2, \\ \sum_n | h_0(n) |^2 &= 2. \end{aligned} \tag{A.7}$$

NECESSARY CONDITION 4

Reproducing Equation (A.6)

$$\sum_n h_0(n) h_0(n - 2k) = 2\delta(k).$$

Summing over k of both sides gives

$$\sum_k \sum_n h_0(n) h_0(n - 2k) = 2. \tag{A.8}$$

Breaking Equation (A.8) into even and odd ordered terms, we obtain

$$\sum_k \left[\sum_m h_0(2m) h_0(2k + 2m) + \sum_m h_0(2k + 2m + 1) h_0(2m + 1) \right] = 2 \tag{A.9}$$

Rearranging terms results in

$$\sum_m \left[\sum_k h_0(2k + 2m) \right] h_0(2m) + \sum_m \left[\sum_k h_0(2k + 2m + 1) \right] h_0(2m + 1) = 2. \tag{A.10}$$

Substituting Equations (2.41) and (2.42) into Equation (A.10), gives

$$S_1 \sum_m h_0\left(2m\right) + S_2 \sum_m h_0\left(2m+1\right) = 2. \tag{A.11}$$

Applying the same equations again into Equation (A.11) gives

$$S_1^2 + S_2^2 = 2. \tag{A.12}$$

Equations 2.41 and 2.42 state that

$$S_1 + S_2 = 2. \tag{A.13}$$

Solving for S_1 and S_2 in Equations (A.11) and (A.12), gives

$$S_1 = \sum_n h_0\left(2n\right) = 1, \tag{A.14}$$

$$S_2 = \sum_n h_0\left(2n+1\right) = 1. \tag{A.15}$$

APPENDIX B. DISTANCE MEASURES

A. SIGNALS FROM TRANSMITTER 1 ARE THE INPUTS

TEMPLATE 1

	SCALE 6	SCALE 7	SCALE 8	SCALE 9	SCALE 10	SCALE 11
SIGNAL 1:	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
SIGNAL 2:	0.0154	0.0189	0.1321	0.1166	0.5972	3.0527
SIGNAL 3:	0.0119	0.0357	0.0512	0.1026	0.6823	2.6195
SIGNAL 4:	0.0119	0.0090	0.0518	0.1234	0.3941	1.0806
SIGNAL 5:	0.0096	0.0596	0.0811	0.1272	1.1857	4.2015
SIGNAL 6:	0.0187	0.0144	0.1260	0.1907	0.5007	2.1553
SIGNAL 7:	0.0088	0.0156	0.0644	0.2489	0.5075	3.4189
SIGNAL 8:	0.0184	0.0353	0.0335	0.1883	0.5085	2.7106
SIGNAL 9:	0.0108	0.0349	0.1020	0.4137	1.0445	5.3966

TEMPLATE 2

	SCALE 6	SCALE 7	SCALE 8	SCALE 9	SCALE 10	SCALE 11
SIGNAL 1:	0.2835	0.2481	0.5950	2.5081	9.6966	44.3073
SIGNAL 2:	0.3811	0.2786	0.6446	2.4572	8.9884	47.2375
SIGNAL 3:	0.3828	0.3385	0.4982	2.3074	8.6315	40.5894
SIGNAL 4:	0.2978	0.2723	0.5886	2.3877	8.8995	42.9072
SIGNAL 5:	0.3043	0.3478	0.4959	2.3277	8.1022	39.6375
SIGNAL 6:	0.3253	0.2469	0.8060	2.3600	9.1666	43.6621
SIGNAL 7:	0.2867	0.2623	0.5220	2.0454	9.1751	39.6036
SIGNAL 8:	0.3772	0.3061	0.5358	2.1604	8.8586	40.6839
SIGNAL 9:	0.3756	0.3014	0.5133	1.9071	8.6736	39.2300

TEMPLATE 3

	SCALE 6	SCALE 7	SCALE 8	SCALE 9	SCALE 10	SCALE 11
SIGNAL 1:	0.0293	0.0478	0.1806	0.7911	3.2748	14.7737
SIGNAL 2:	0.0195	0.0516	0.2315	0.7833	3.4095	16.3406
SIGNAL 3:	0.0292	0.0873	0.1219	0.7441	3.3865	13.7970
SIGNAL 4:	0.0147	0.0461	0.1442	0.7797	3.1756	15.4847
SIGNAL 5:	0.0247	0.1011	0.1338	0.7385	2.8243	13.1612
SIGNAL 6:	0.0143	0.0448	0.1829	0.6879	3.2091	13.4287
SIGNAL 7:	0.0215	0.0571	0.1285	0.8024	3.2493	13.7654
SIGNAL 8:	0.0454	0.0721	0.1630	0.7261	3.3426	14.3787
SIGNAL 9:	0.0333	0.0683	0.1201	0.7172	3.0344	12.5220

TEMPLATE 4

	SCALE 6	SCALE 7	SCALE 8	SCALE 9	SCALE 10	SCALE 11
SIGNAL 1:	0.3094	0.5771	1.2600	4.7861	23.4442	107.8438
SIGNAL 2:	0.3395	0.5918	1.3712	4.9459	21.3511	103.8587
SIGNAL 3:	0.3230	0.6083	1.2067	4.6558	21.4529	99.7881
SIGNAL 4:	0.3675	0.5963	1.2665	4.8759	22.1737	105.3080
SIGNAL 4:	0.3766	0.6255	1.1815	4.7152	20.7366	101.1619
SIGNAL 6:	0.3035	0.5860	1.3969	4.9315	22.9453	110.3185
SIGNAL 7:	0.3733	0.6054	1.1867	4.3094	22.0092	95.2878
SIGNAL 8:	0.2758	0.5979	1.2183	4.4715	21.8996	99.7182
SIGNAL 9:	0.3427	0.6121	1.1942	4.3671	21.8617	101.6282

B. SIGNALS FROM TRANSMITTER 2 ARE THE INPUTS

TEMPLATE 1

	SCALE 6	SCALE 7	SCALE 8	SCALE 9	SCALE 10	SCALE 11
SIGNAL 1:	0.2835	0.2481	0.5950	2.5081	9.6966	44.3073
SIGNAL 2:	0.1008	0.2111	0.5449	2.4071	9.9080	42.9127
SIGNAL 3:	0.2231	0.2929	0.6255	2.2173	9.5509	45.2351
SIGNAL 4:	0.1587	0.3089	0.7071	2.4259	10.2745	45.9598
SIGNAL 5:	0.2397	0.3638	0.6821	2.6701	10.6592	50.7797
SIGNAL 6:	0.1881	0.2625	0.6220	2.6232	10.7310	49.5033
SIGNAL 7:	0.1906	0.3823	0.6591	2.3317	10.0235	50.3606
SIGNAL 8:	0.3259	0.2937	0.7425	2.5765	11.2478	49.7514
SIGNAL 9:	0.3105	0.3787	0.6678	2.5760	11.1999	52.8772

TEMPLATE 2

	SCALE 6	SCALE 7	SCALE 8	SCALE 9	SCALE 10	SCALE 11
SIGNAL 1:	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
SIGNAL 2:	0.1504	0.1834	0.2414	1.1648	2.9675	8.3316
SIGNAL 3:	0.0926	0.1289	0.2083	0.3845	2.0463	11.7426
SIGNAL 4:	0.1542	0.1445	0.2714	0.2385	2.9824	10.2347
SIGNAL 5:	0.1100	0.1513	0.1427	0.2236	1.4235	12.0260
SIGNAL 6:	0.0919	0.0599	0.0953	0.2324	3.0988	3.5685
SIGNAL 7:	0.1120	0.0864	0.2697	1.0350	0.4110	6.5288
SIGNAL 8:	0.0429	0.0602	0.3798	0.2762	3.4554	2.7844
SIGNAL 9:	0.0585	0.1117	0.2090	1.1790	3.0197	8.5858

TEMPLATE 3

	SCALE 6	SCALE 7	SCALE 8	SCALE 9	SCALE 10	SCALE 11
SIGNAL 1:	0.3185	0.2155	0.5388	2.5937	10.7275	52.0090
SIGNAL 2:	0.1653	0.2228	0.4821	2.8090	10.7830	46.1492
SIGNAL 3:	0.3053	0.3015	0.6483	2.4348	10.0656	48.1584
SIGNAL 4:	0.2162	0.3174	0.7455	2.4423	10.9963	48.9938
SIGNAL 5:	0.3320	0.3800	0.6067	2.6860	11.4619	55.0565
SIGNAL 6:	0.2723	0.2287	0.6431	2.5235	11.5436	55.4853
SIGNAL 7:	0.2483	0.3425	0.7296	2.5858	10.9103	55.0225
SIGNAL 8:	0.3857	0.2846	0.8665	2.5850	12.4298	54.1137
SIGNAL 9:	0.4016	0.3610	0.6024	2.8528	11.8614	57.5092

TEMPLATE 4

	SCALE 6	SCALE 7	SCALE 8	SCALE 9	SCALE 10	SCALE 11
SIGNAL 1:	0.0688	0.3144	0.7227	2.4151	10.7828	46.0266
SIGNAL 2:	0.2709	0.4742	0.7804	2.4630	12.0574	53.2623
SIGNAL 3:	0.1439	0.2997	0.7299	2.3018	12.1275	51.6991
SIGNAL 4:	0.2207	0.3734	0.7884	2.6920	12.1243	49.1001
SIGNAL 5:	0.1785	0.2800	0.6635	2.7456	11.6775	49.5180
SIGNAL 6:	0.1251	0.3715	0.8402	2.9717	12.1831	47.9240
SIGNAL 7:	0.1568	0.2853	0.8519	2.6483	11.8194	52.1264
SIGNAL 8:	0.1382	0.2722	0.8705	2.6883	11.7220	47.9139
SIGNAL 9:	0.1242	0.3436	0.6856	2.4425	12.4913	50.0139

C. SIGNALS FROM TRANSMITTER 3 ARE THE INPUTS

TEMPLATE 1

	SCALE 6	SCALE 7	SCALE 8	SCALE 9	SCALE 10	SCALE 11
SIGNAL 1:	0.0293	0.0478	0.1806	0.7911	3.2748	14.7737
SIGNAL 2:	0.0258	0.0486	0.1886	0.7386	3.4439	16.0413
SIGNAL 3:	0.0258	0.0488	0.1760	0.7199	2.9349	14.0412
SIGNAL 4:	0.0243	0.0380	0.2365	0.8262	3.1571	15.9305
SIGNAL 5:	0.0351	0.0413	0.2032	0.7480	3.3249	15.7214
SIGNAL 6:	0.0181	0.0522	0.1437	0.6904	3.3154	15.6880
SIGNAL 7:	0.0204	0.0407	0.1965	0.6858	3.0757	14.9835
SIGNAL 8:	0.0211	0.0487	0.1396	0.7145	3.1253	14.8495
SIGNAL 9:	0.0132	0.0385	0.1556	0.6820	2.6568	13.9337

TEMPLATE 2

	SCALE 6	SCALE 7	SCALE 8	SCALE 9	SCALE 10	SCALE 11
SIGNAL 1:	0.3185	0.2155	0.5388	2.5937	10.7275	52.0090
SIGNAL 2:	0.2770	0.2549	0.6212	2.6437	10.7470	58.6234
SIGNAL 3:	0.3123	0.2733	0.5398	2.9106	8.9611	54.0690
SIGNAL 4:	0.3022	0.2595	0.6690	2.7903	9.6474	53.2208
SIGNAL 5:	0.2388	0.2737	0.5984	2.6087	10.4443	58.9315
SIGNAL 6:	0.2715	0.2588	0.5428	2.5480	9.9815	52.8461
SIGNAL 7:	0.2536	0.2180	0.5879	2.3883	10.0475	54.4567
SIGNAL 8:	0.2530	0.2886	0.5516	2.6977	10.0750	50.2555
SIGNAL 9:	0.2634	0.2249	0.4667	2.5582	7.2510	43.6765

TEMPLATE 3

	SCALE 6	SCALE 7	SCALE 8	SCALE 9	SCALE 10	SCALE 11
SIGNAL 1:	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
SIGNAL 2:	0.0145	0.0182	0.1264	0.1059	0.2600	2.5576
SIGNAL 3:	0.0140	0.0574	0.0679	0.1825	0.5453	1.9164
SIGNAL 4:	0.0087	0.0243	0.1422	0.0815	0.2531	0.5718
SIGNAL 5:	0.0202	0.0205	0.1187	0.0708	0.2221	1.8509
SIGNAL 6:	0.0173	0.0460	0.0522	0.1326	0.5117	1.8301
SIGNAL 7:	0.0080	0.0164	0.2044	0.1456	0.2368	0.6173
SIGNAL 8:	0.0202	0.0689	0.0991	0.1083	0.3890	1.7340
SIGNAL 9:	0.0165	0.0355	0.0718	0.1247	1.3389	2.9981

TEMPLATE 4

	SCALE 6	SCALE 7	SCALE 8	SCALE 9	SCALE 10	SCALE 11
SIGNAL 1:	0.3651	0.5208	1.1902	5.0321	23.6918	115.5311
SIGNAL 2:	0.3238	0.5603	1.3472	5.0435	23.8257	122.6571
SIGNAL 3:	0.3244	0.5725	1.2440	5.3595	22.0900	119.5437
SIGNAL 4:	0.3540	0.5550	1.3640	5.2345	22.9700	116.3537
SIGNAL 5:	0.2868	0.5780	1.3037	5.0934	23.5456	124.5405
SIGNAL 6:	0.3084	0.5596	1.2133	4.8809	23.0578	117.4324
SIGNAL 7:	0.2942	0.5274	1.3532	4.6602	23.2007	118.0311
SIGNAL 8:	0.2827	0.5685	1.2326	5.0956	23.2144	113.1257
SIGNAL 9:	0.2940	0.5262	1.0909	4.9186	19.2677	104.3332

D. SIGNALS FROM TRANSMITTER 4 ARE THE INPUTS

TEMPLATE 1

	SCALE 6	SCALE 7	SCALE 8	SCALE 9	SCALE 10	SCALE 11
SIGNAL 1:	0.3094	0.5771	1.2600	4.7861	23.4442	107.8438
SIGNAL 2:	0.3736	0.9836	3.0273	9.2057	38.3416	164.0387
SIGNAL 3:	0.3331	0.6218	1.3112	5.3833	24.3415	113.3645
SIGNAL 4:	0.3133	0.5338	1.3146	5.2073	24.1267	112.2255
SIGNAL 5:	0.4374	0.5831	1.1027	5.0784	23.9667	107.1038
SIGNAL 6:	0.3432	0.5722	1.2066	5.4055	23.0626	110.6970
SIGNAL 7:	0.2560	0.4075	0.8567	3.8320	18.1956	83.7372
SIGNAL 8:	0.2761	0.4673	1.3469	4.5214	28.6522	133.7564
SIGNAL 9:	0.2699	0.6246	1.7105	7.5262	30.5762	136.1710

TEMPLATE 2

	SCALE 6	SCALE 7	SCALE 8	SCALE 9	SCALE 10	SCALE 11
SIGNAL 1:	0.0688	0.3144	0.7227	2.4151	10.7828	46.0266
SIGNAL 2:	0.2104	0.6746	1.6970	4.8096	18.7294	72.9213
SIGNAL 3:	0.1649	0.2812	0.8120	2.6968	11.2046	50.4846
SIGNAL 4:	0.1370	0.2327	0.8829	2.6756	11.2253	57.7131
SIGNAL 5:	0.1847	0.3027	0.5789	2.4825	10.9737	45.7909
SIGNAL 6:	0.1039	0.2557	0.6717	2.8612	10.1952	53.3557
SIGNAL 7:	0.1637	0.1727	0.5083	2.1826	6.2022	38.8847
SIGNAL 8:	0.1866	0.2580	0.8640	2.9625	12.0382	54.5502
SIGNAL 9:	0.1472	0.3361	0.9322	4.4756	14.4394	55.8276

TEMPLATE 3

	SCALE 6	SCALE 7	SCALE 8	SCALE 9	SCALE 10	SCALE 11
SIGNAL 1:	0.3651	0.5208	1.1902	5.0321	23.6918	115.5311
SIGNAL 2:	0.4188	0.8757	2.3272	8.7571	35.0247	158.7921
SIGNAL 3:	0.2974	0.5532	1.0881	5.5421	24.5287	120.8973
SIGNAL 4:	0.3393	0.4954	1.1246	5.4472	24.5602	122.9673
SIGNAL 5:	0.4295	0.5159	1.0414	5.1931	24.3348	114.7444
SIGNAL 6:	0.3967	0.5028	1.0955	5.6244	23.1653	119.4670
SIGNAL 7:	0.2582	0.3225	0.7155	4.1114	18.9958	94.1898
SIGNAL 8:	0.2952	0.3749	1.1948	5.2218	25.6827	132.6012
SIGNAL 9:	0.2697	0.5467	1.3014	6.8545	27.5281	133.7723

TEMPLATE 4

	SCALE 6	SCALE 7	SCALE 8	SCALE 9	SCALE 10	SCALE 11
SIGNAL 1:	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
SIGNAL 2:	0.1732	0.5275	1.2713	3.4912	9.3186	30.3546
SIGNAL 3:	0.0544	0.0466	0.3896	0.4135	2.4614	6.5196
SIGNAL 4:	0.0610	0.0694	0.4249	0.3959	2.6097	14.8425
SIGNAL 5:	0.0986	0.0655	0.1514	0.4633	2.2296	6.8575
SIGNAL 6:	0.0528	0.0506	0.0990	0.5903	3.7932	11.8557
SIGNAL 7:	0.0814	0.1619	0.4059	1.7605	4.4264	22.1434
SIGNAL 8:	0.0935	0.3151	0.6468	2.8861	13.5405	47.6043
SIGNAL 9:	0.0776	0.3200	0.9673	3.8654	13.8608	49.1905

E. SIGNALS FROM TRANSMITTER 1 ARE THE INPUTS: 10 DB SNR REDUCTION

TEMPLATE 1

	SCALE 6	SCALE 7	SCALE 8	SCALE 9	SCALE 10	SCALE 11
SIGNAL 1:	0.0375	0.1043	0.2759	0.7683	1.3094	3.3502
SIGNAL 2:	0.0468	0.1085	0.3193	0.7442	2.0640	3.6190
SIGNAL 3:	0.0431	0.0800	0.2972	0.6296	1.3008	3.5047
SIGNAL 4:	0.0504	0.1064	0.3262	0.7624	1.7088	3.4416
SIGNAL 5:	0.0421	0.0706	0.3415	0.6067	1.5736	5.5176
SIGNAL 6:	0.0508	0.0743	0.3502	0.8021	1.5439	5.0661
SIGNAL 7:	0.0291	0.1163	0.3047	0.6125	1.2903	3.8559
SIGNAL 8:	0.0377	0.1302	0.2676	0.5524	1.3807	3.7544
SIGNAL 9:	0.0337	0.1012	0.2747	0.5609	1.5242	5.8662

TEMPLATE 2

	SCALE 6	SCALE 7	SCALE 8	SCALE 9	SCALE 10	SCALE 11
SIGNAL 1:	0.3553	0.2728	0.6642	2.8099	10.4138	47.5912
SIGNAL 2:	0.2227	0.2168	0.6260	2.7765	11.2037	48.0249
SIGNAL 3:	0.3561	0.2650	0.5283	2.4908	9.4202	43.7669
SIGNAL 4:	0.2451	0.2573	0.7072	2.5680	10.7271	46.0365
SIGNAL 5:	0.2911	0.2100	0.5792	2.2985	9.2627	42.8591
SIGNAL 6:	0.2959	0.2452	0.6188	2.8187	10.3953	49.7207
SIGNAL 7:	0.2792	0.3019	0.5697	2.5446	8.9549	42.6443
SIGNAL 8:	0.3114	0.2584	0.5708	2.5522	9.9918	43.5053
SIGNAL 9:	0.3268	0.2492	0.5431	2.2212	9.0768	42.7932

TEMPLATE 3

	SCALE 6	SCALE 7	SCALE 8	SCALE 9	SCALE 10	SCALE 11
SIGNAL 1:	0.0359	0.1303	0.3973	1.0658	3.8188	15.6111
SIGNAL 2:	0.0389	0.0850	0.3844	1.0643	4.3243	16.0790
SIGNAL 3:	0.0430	0.0984	0.3287	1.0871	4.0226	16.3021
SIGNAL 4:	0.0373	0.1119	0.3942	1.0760	3.9323	16.3871
SIGNAL 5:	0.0346	0.0814	0.3848	1.1143	4.0585	15.6347
SIGNAL 6:	0.0390	0.0843	0.4338	0.9025	3.6642	14.2155
SIGNAL 7:	0.0231	0.1120	0.3312	1.1282	4.3698	17.4460
SIGNAL 8:	0.0353	0.1325	0.3284	1.0011	4.0881	16.2695
SIGNAL 9:	0.0315	0.1139	0.3366	1.0887	3.4706	13.8132

TEMPLATE 4

	SCALE 6	SCALE 7	SCALE 8	SCALE 9	SCALE 10	SCALE 11
SIGNAL 1:	0.3328	0.5612	1.3761	4.9135	22.4741	107.4480
SIGNAL 2:	0.2562	0.4936	1.2892	4.8175	22.7738	102.1320
SIGNAL 3:	0.3450	0.5801	1.1919	4.5966	20.6796	100.6919
SIGNAL 4:	0.3263	0.5714	1.3486	4.7544	23.1659	106.4092
SIGNAL 5:	0.3633	0.5636	1.2499	4.3789	20.7572	97.1743
SIGNAL 6:	0.3042	0.5860	1.3593	5.1565	22.9280	115.2724
SIGNAL 7:	0.3338	0.4590	1.2041	4.6390	20.0266	96.3665
SIGNAL 8:	0.3610	0.5003	1.2423	4.3976	21.1563	98.0142
SIGNAL 9:	0.3608	0.5617	1.2187	4.3478	21.0854	100.2342

F. SIGNALS FROM TRANSMITTER 2 ARE INPUTS: 10 DB SNR REDUCTION

TEMPLATE 1

	SCALE 6	SCALE 7	SCALE 8	SCALE 9	SCALE 10	SCALE 11
SIGNAL 1:	0.2998	0.2975	0.6936	2.5272	9.2397	41.5986
SIGNAL 2:	0.1149	0.2315	0.4739	2.1895	9.2329	42.4683
SIGNAL 3:	0.2483	0.3285	0.6071	2.1489	9.2771	44.6032
SIGNAL 4:	0.1683	0.2955	0.6568	2.4697	9.9201	46.1952
SIGNAL 5:	0.2497	0.3499	0.6493	2.6333	10.4630	46.5821
SIGNAL 6:	0.1699	0.3207	0.6972	2.3188	9.9021	43.4716
SIGNAL 7:	0.2107	0.4434	0.8268	2.5254	10.3927	45.1681
SIGNAL 8:	0.3312	0.3403	0.8018	2.6353	10.2630	47.7992
SIGNAL 9:	0.3012	0.4205	0.7371	2.3549	10.5009	47.2822

TEMPLATE 2

	SCALE 6	SCALE 7	SCALE 8	SCALE 9	SCALE 10	SCALE 11
SIGNAL 1:	0.0353	0.1029	0.3632	1.0898	3.2442	9.5561
SIGNAL 2:	0.1267	0.1354	0.2513	0.9447	2.8265	8.8378
SIGNAL 3:	0.0796	0.1694	0.4690	0.6193	3.2228	9.6555
SIGNAL 4:	0.1472	0.1665	0.4090	0.8105	7.2229	14.2345
SIGNAL 5:	0.0873	0.1667	0.2662	0.9486	2.9736	13.1501
SIGNAL 6:	0.1078	0.1217	0.4735	1.5780	5.9286	21.2568
SIGNAL 7:	0.1093	0.1754	0.6404	1.3555	8.2270	18.1313
SIGNAL 8:	0.0543	0.1232	0.5491	1.1825	6.1173	8.1226
SIGNAL 9:	0.0705	0.1886	0.4846	2.8504	9.8706	19.6026

TEMPLATE 3

	SCALE 6	SCALE 7	SCALE 8	SCALE 9	SCALE 10	SCALE 11
SIGNAL 1:	0.3378	0.2623	0.6906	2.3864	10.3773	48.5725
SIGNAL 2:	0.1748	0.2254	0.4875	2.3375	9.7415	46.8498
SIGNAL 3:	0.3212	0.3164	0.7229	2.2456	9.7959	48.5703
SIGNAL 4:	0.2274	0.3074	0.7264	2.4591	10.1376	49.7111
SIGNAL 5:	0.3371	0.3617	0.6446	2.5659	11.1484	51.7127
SIGNAL 6:	0.2661	0.2662	0.7978	2.5988	10.7795	48.9604
SIGNAL 7:	0.2570	0.3376	0.9129	2.2908	10.6121	47.4262
SIGNAL 8:	0.3939	0.3071	0.9439	2.4221	10.1326	54.1949
SIGNAL 9:	0.3870	0.3634	0.8302	2.3635	10.7260	49.2730

TEMPLATE 4

	SCALE 6	SCALE 7	SCALE 8	SCALE 9	SCALE 10	SCALE 11
SIGNAL 1:	0.1007	0.2607	0.8255	2.5868	9.9048	44.4863
SIGNAL 2:	0.2222	0.2770	0.6216	2.0512	10.6348	48.9550
SIGNAL 3:	0.1201	0.3323	0.8504	2.2053	10.3880	45.7483
SIGNAL 4:	0.2068	0.3380	0.8050	2.4836	11.3920	47.6053
SIGNAL 5:	0.1445	0.2756	0.6615	2.7278	10.2407	51.0403
SIGNAL 6:	0.1370	0.3112	0.9905	2.0032	11.8533	53.8810
SIGNAL 7:	0.1369	0.3029	0.9008	2.7150	11.7296	53.8566
SIGNAL 8:	0.1439	0.2392	0.8706	3.2170	12.0501	46.7004
SIGNAL 9:	0.1384	0.3070	0.9407	3.0078	12.4255	52.4257

G. SIGNALS FROM TRANSMITTER 3 ARE INPUTS: 10 DB SNR REDUCTION

TEMPLATE 1

	SCALE 6	SCALE 7	SCALE 8	SCALE 9	SCALE 10	SCALE 11
SIGNAL 1:	0.0478	0.1274	0.5144	1.1785	4.0000	18.1846
SIGNAL 2:	0.0365	0.1234	0.4115	1.2813	4.1553	18.5676
SIGNAL 3:	0.0684	0.1219	0.3577	1.1853	3.8279	16.0824
SIGNAL 4:	0.0523	0.1170	0.4138	1.1140	4.3548	18.4488
SIGNAL 5:	0.0348	0.1336	0.5074	1.2930	3.9845	17.9401
SIGNAL 6:	0.0550	0.1275	0.4415	1.1653	3.9053	18.0756
SIGNAL 7:	0.0478	0.1284	0.4557	1.1081	3.6589	17.1620
SIGNAL 8:	0.0531	0.0784	0.3782	1.1528	4.0583	17.6531
SIGNAL 9:	0.0320	0.1208	0.3447	1.0956	3.6144	16.3529

TEMPLATE 2

	SCALE 6	SCALE 7	SCALE 8	SCALE 9	SCALE 10	SCALE 11
SIGNAL 1:	0.2735	0.2305	0.8375	3.2786	12.6610	59.1120
SIGNAL 2:	0.2489	0.2555	0.7085	3.4069	13.2948	64.9324
SIGNAL 3:	0.2403	0.2307	0.6814	3.2335	13.1578	56.7576
SIGNAL 4:	0.2543	0.2354	0.6180	3.1848	12.5049	61.7671
SIGNAL 5:	0.2546	0.1906	0.8137	3.4720	12.2753	62.2135
SIGNAL 6:	0.2131	0.1717	0.8292	3.7610	11.4564	59.3982
SIGNAL 7:	0.2271	0.2378	0.7183	2.9783	12.5854	62.4342
SIGNAL 8:	0.2246	0.2336	0.7436	3.1300	12.7692	62.2046
SIGNAL 9:	0.2461	0.2098	0.6158	2.7632	10.8813	49.2295

TEMPLATE 3

	SCALE 6	SCALE 7	SCALE 8	SCALE 9	SCALE 10	SCALE 11
SIGNAL 1:	0.0203	0.0815	0.6030	0.9418	2.5238	6.4598
SIGNAL 2:	0.0201	0.0950	0.4735	1.0146	3.0909	9.5653
SIGNAL 3:	0.0465	0.0833	0.3634	0.9566	2.7794	6.4284
SIGNAL 4:	0.0362	0.0954	0.4588	0.7535	2.6418	7.8841
SIGNAL 5:	0.0253	0.0951	0.5927	1.0052	2.3614	7.6033
SIGNAL 6:	0.0485	0.0933	0.5202	1.0480	1.7231	5.9440
SIGNAL 7:	0.0328	0.0992	0.5659	0.7845	2.2809	8.0012
SIGNAL 8:	0.0354	0.0650	0.4643	0.8565	2.4123	8.5680
SIGNAL 9:	0.0193	0.0851	0.3710	0.7626	2.3411	5.4612

TEMPLATE 4

	SCALE 6	SCALE 7	SCALE 8	SCALE 9	SCALE 10	SCALE 11
SIGNAL 1:	0.3287	0.5182	1.5587	5.8201	25.7509	122.5166
SIGNAL 2:	0.1893	0.5261	1.4451	5.8022	26.7385	129.8573
SIGNAL 3:	0.2463	0.5399	1.3549	5.8309	25.7889	119.4983
SIGNAL 4:	0.3308	0.5394	1.3779	5.5470	25.6568	127.0563
SIGNAL 5:	0.2908	0.4859	1.6405	6.0450	25.2253	127.3273
SIGNAL 6:	0.2010	0.4756	1.5749	6.3159	24.4335	122.7423
SIGNAL 7:	0.2183	0.5214	1.4519	5.5075	25.3703	126.9125
SIGNAL 8:	0.1728	0.4902	1.4167	5.4070	25.5855	126.2491
SIGNAL 9:	0.2523	0.5030	1.2223	5.0020	23.1458	109.1846

H. SIGNALS FROM TRANSMITTER 4 ARE INPUTS: 10 DB SNR REDUCTION

TEMPLATE 1

	SCALE 6	SCALE 7	SCALE 8	SCALE 9	SCALE 10	SCALE 11
SIGNAL 1:	0.3270	0.5990	1.6784	5.8691	25.0385	118.3022
SIGNAL 2:	0.3248	0.5290	1.7946	5.8942	23.5419	117.8467
SIGNAL 3:	0.3220	0.6044	1.5821	6.1968	25.1165	118.5933
SIGNAL 4:	0.4828	0.4838	1.1432	5.2652	22.8460	112.4434
SIGNAL 5:	0.3500	0.5793	1.2490	5.8876	23.7742	113.3160
SIGNAL 6:	0.2645	0.4183	1.0173	4.2269	18.5720	84.9252

TEMPLATE 2

	SCALE 6	SCALE 7	SCALE 8	SCALE 9	SCALE 10	SCALE 11
SIGNAL 1:	0.0850	0.3153	1.0099	3.3984	13.1845	62.8281
SIGNAL 2:	0.1570	0.2157	0.8610	3.0255	13.4155	53.4214
SIGNAL 3:	0.1514	0.2960	0.8447	3.6034	12.9318	50.1148
SIGNAL 4:	0.2097	0.2221	0.7012	2.9600	10.1786	54.1355
SIGNAL 5:	0.1014	0.3442	0.7740	2.8104	13.7426	49.5709
SIGNAL 6:	0.1545	0.2012	0.7999	2.0320	8.7308	34.0015

TEMPLATE 3

	SCALE 6	SCALE 7	SCALE 8	SCALE 9	SCALE 10	SCALE 11
SIGNAL 1:	0.3780	0.5205	1.5290	5.9429	24.3696	121.4879
SIGNAL 2:	0.3014	0.4644	1.5550	5.8505	22.9977	121.4152
SIGNAL 3:	0.3627	0.5140	1.4117	6.1926	24.7024	123.3364
SIGNAL 4:	0.4693	0.4223	1.0334	5.4432	23.3718	117.9020
SIGNAL 5:	0.3955	0.5023	1.1421	5.9523	24.0663	119.0966
SIGNAL 6:	0.2624	0.3113	0.9258	4.2049	18.3186	87.9338

TEMPLATE 4

	SCALE 6	SCALE 7	SCALE 8	SCALE 9	SCALE 10	SCALE 11
SIGNAL 1:	0.0328	0.0486	0.4642	1.0811	4.1788	23.1301
SIGNAL 2:	0.0725	0.1947	0.2994	0.9492	2.7563	11.3148
SIGNAL 3:	0.0789	0.0741	0.3058	1.0105	4.5781	9.8803
SIGNAL 4:	0.1255	0.2016	0.2156	0.7971	1.8608	17.1363
SIGNAL 5:	0.0563	0.0770	0.2064	1.9411	4.4024	11.8552
SIGNAL 6:	0.0804	0.2145	0.7304	1.9989	7.3588	29.4086

APPENDIX C

PREPROCESSED SIGNALS AND TEMPLATE OUTPUTS

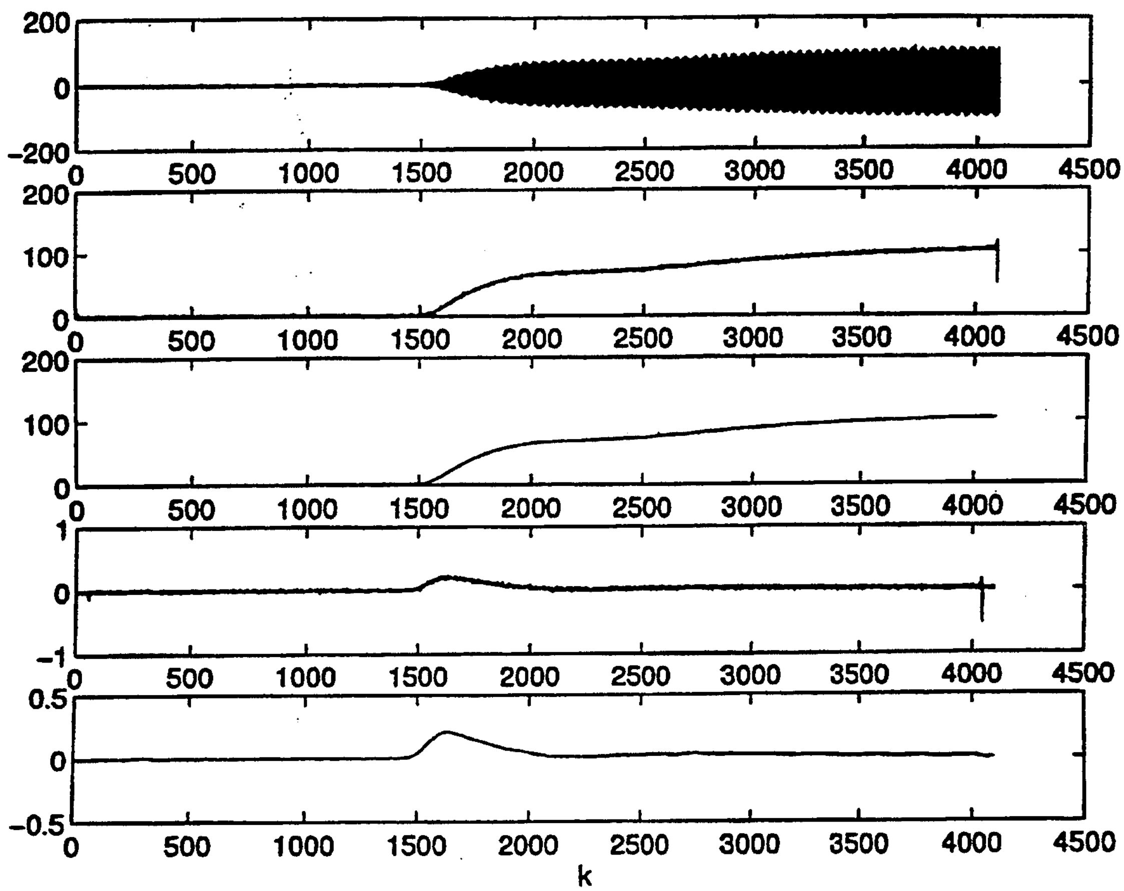


Figure C.1 Preprocessing scheme to obtain an appropriate waveform for Signal 1 of Transmitter 1. From top to bottom: Signal 1 of Transmitter 1 (d.c. removed); Upper envelope of Signal 1; Envelope after median filtering of size 100; Differential of median filtering output; Differential after median filtering of size 50.

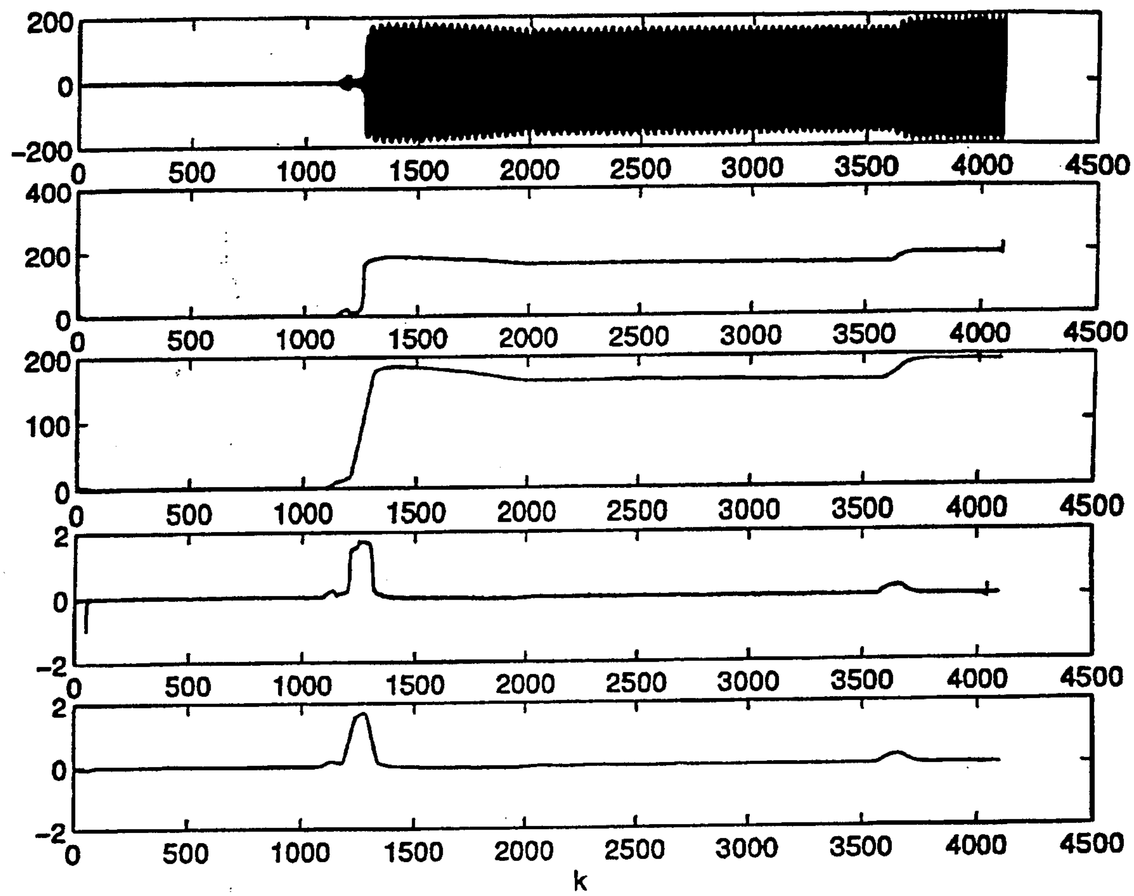


Figure C.2 Preprocessing scheme to obtain appropriate waveform for Signal 1 of Transmitter 2. From top to bottom: Signal 1 of Transmitter 2 (d.c. removed); Upper envelope of Signal 1; Envelope after median filtering of size 100; Differential of median filtering output; Differential after median filtering of size 50.

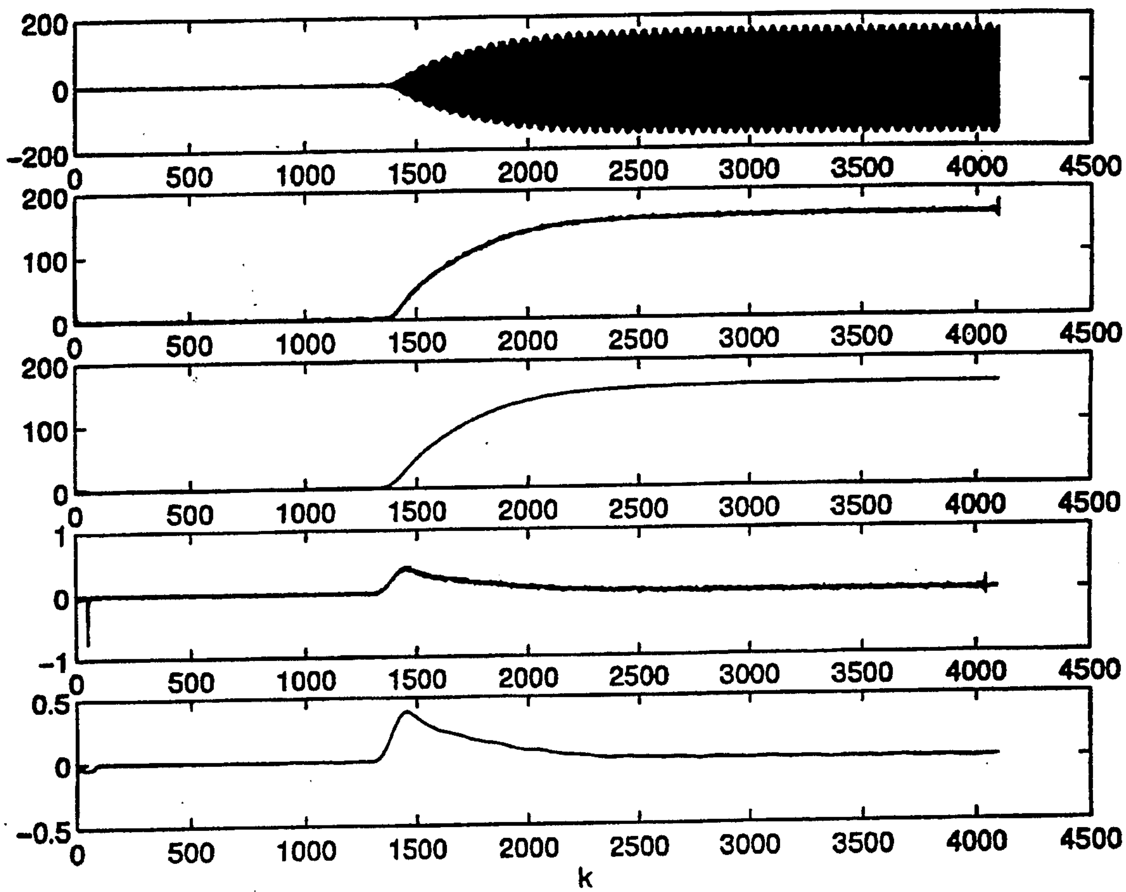


Figure C.3 Preprocessing scheme to obtain appropriate waveform for Signal 1 of Transmitter 3. From top to bottom: Signal 1 of Transmitter 3 (d.c. removed); Upper envelope of Signal 1; Envelope after median filtering of size 100; Differential of median filtering output; Differential after median filtering of size 50.

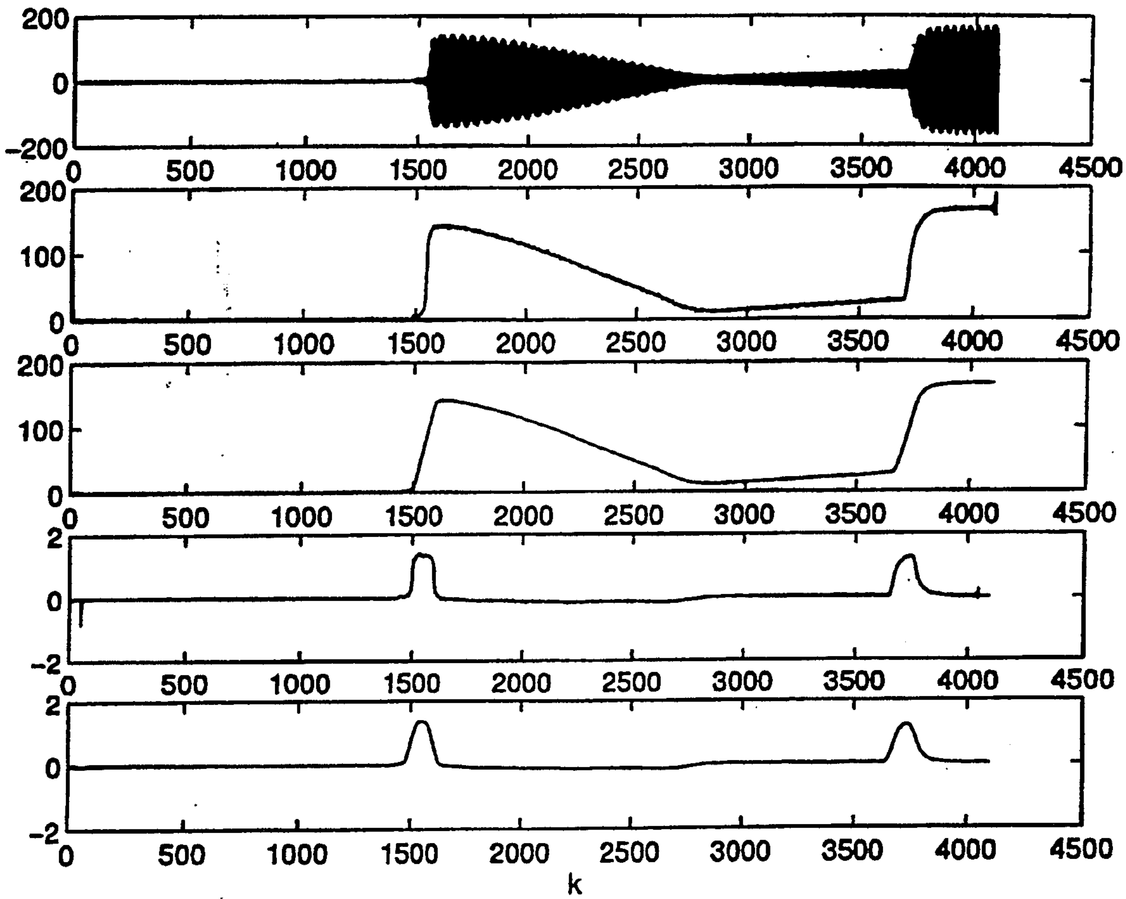


Figure C.4 Preprocessing scheme to obtain appropriate waveform for Signal 1 of Transmitter 4. From top to bottom: Signal 1 of Transmitter 4 (d.c. removed); Upper envelope of Signal 1; Envelope after median filtering of size 100; Differential of median filtering output; Differential after median filtering of size 50.

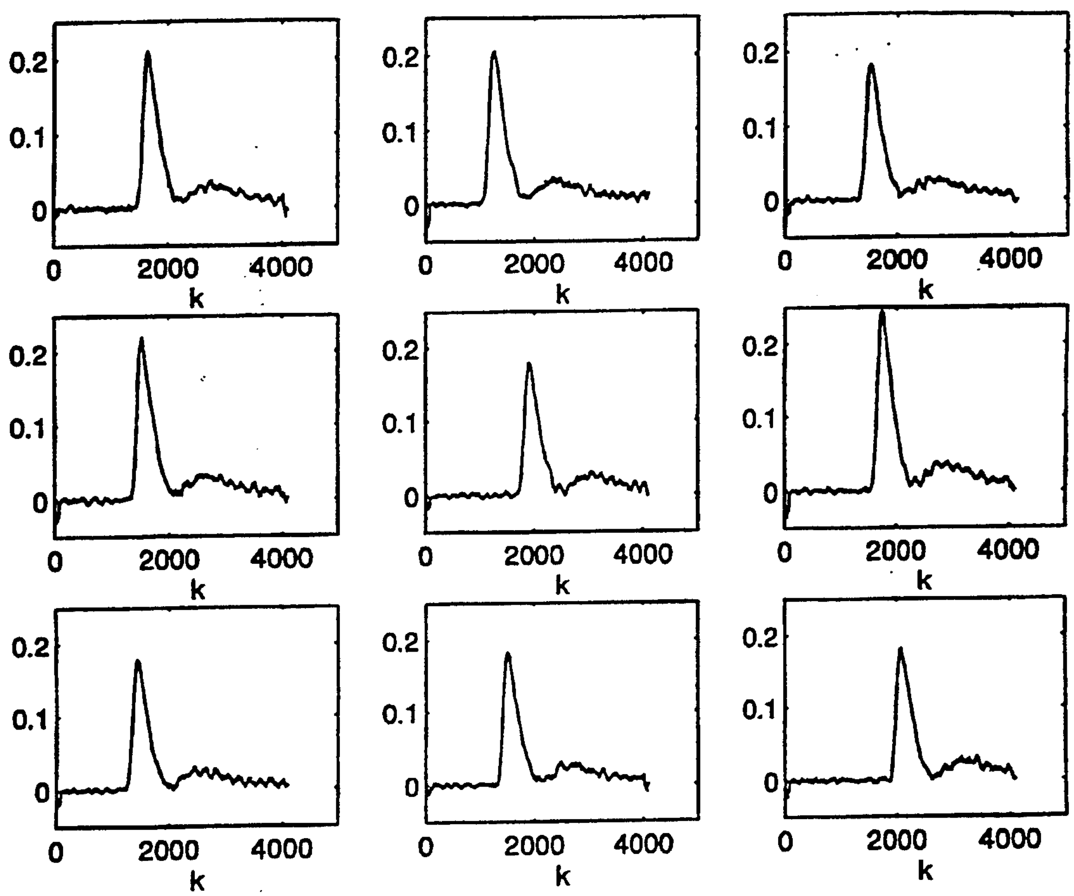


Figure C.5 Preprocessed signals from Transmitter 1.
Top row (left to right): Signal 1; Signal 2; Signal 3
Center row: Signal 4; Signal 5; Signal 6
Bottom row: Signal 7; Signal 8; Signal 9

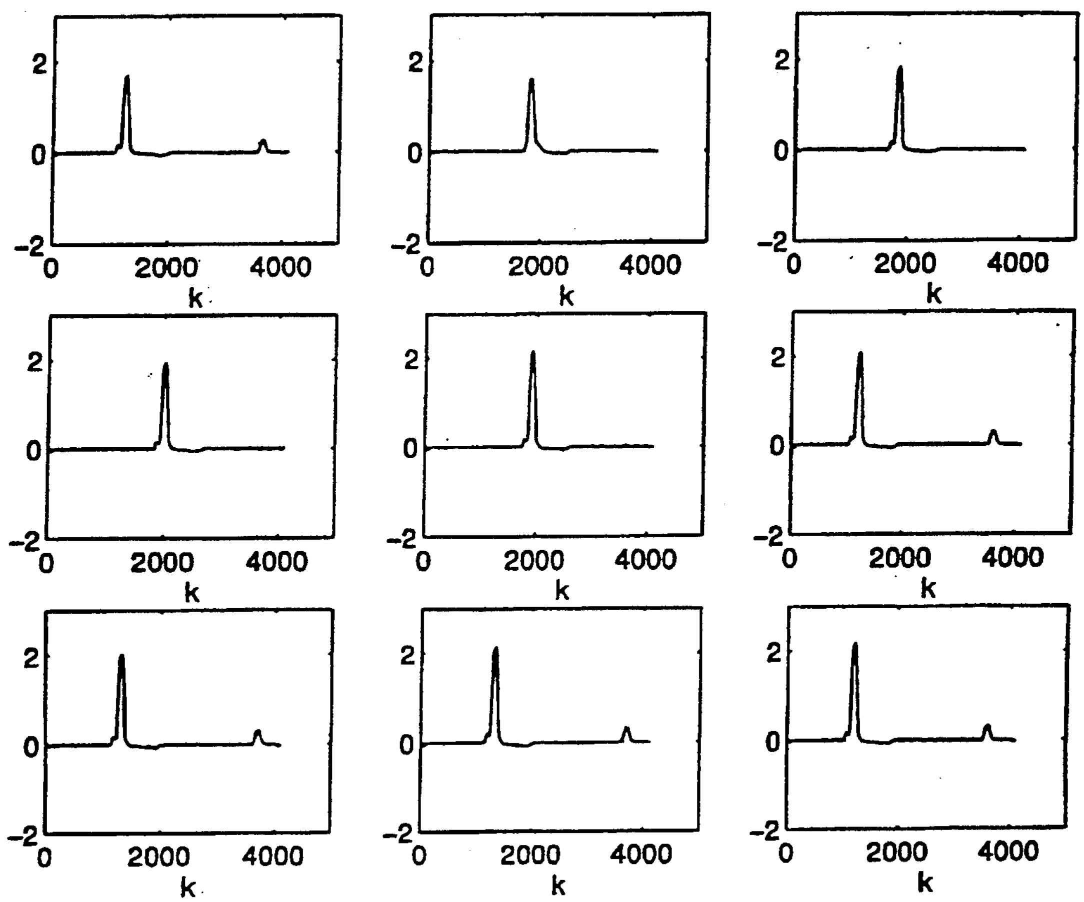


Figure C.6 Preprocessed signals from Transmitter 2.
Top row (left to right): Signal 1; Signal 2; Signal 3
Center row: Signal 4; Signal 5; Signal 6
Bottom row: Signal 7; Signal 8; Signal 9

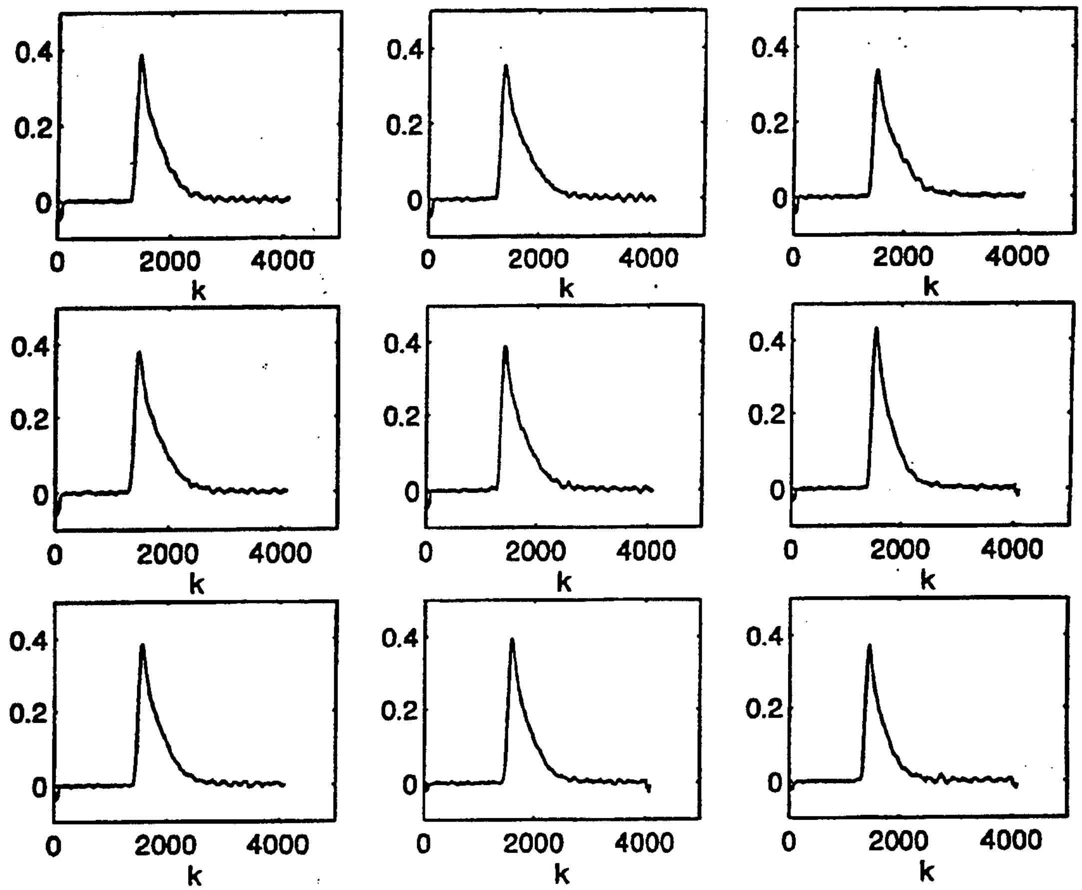


Figure C.7 Preprocessed signals from Transmitter 3.
Top row (left to right): Signal 1; Signal 2; Signal 3
Center row: Signal 4; Signal 5; Signal 6
Bottom row: Signal 7; Signal 8; Signal 9

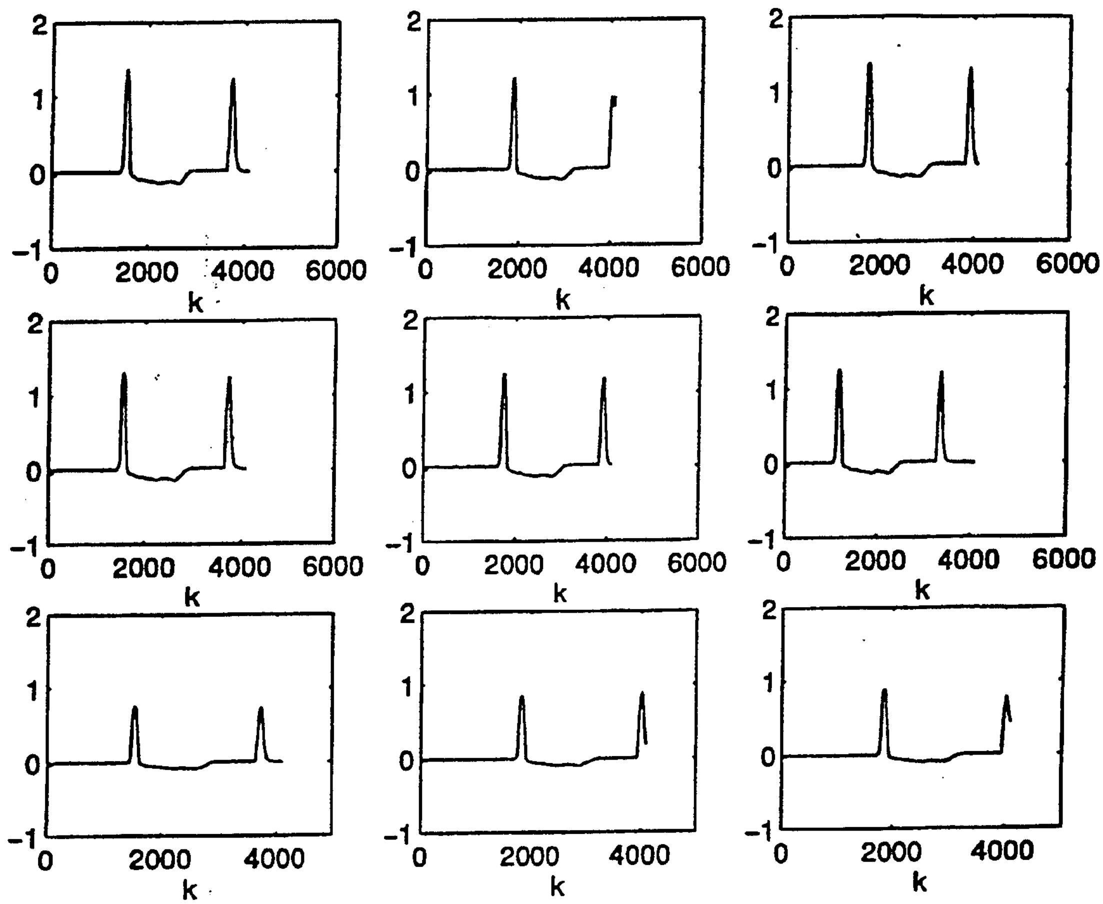


Figure C.8 Preprocessed signals from Transmitter 4.
Top row (left to right): Signal 1; Signal 2; Signal 3
Center row: Signal 4; Signal 5; Signal 6
Bottom row: Signal 7; Signal 8; Signal 9

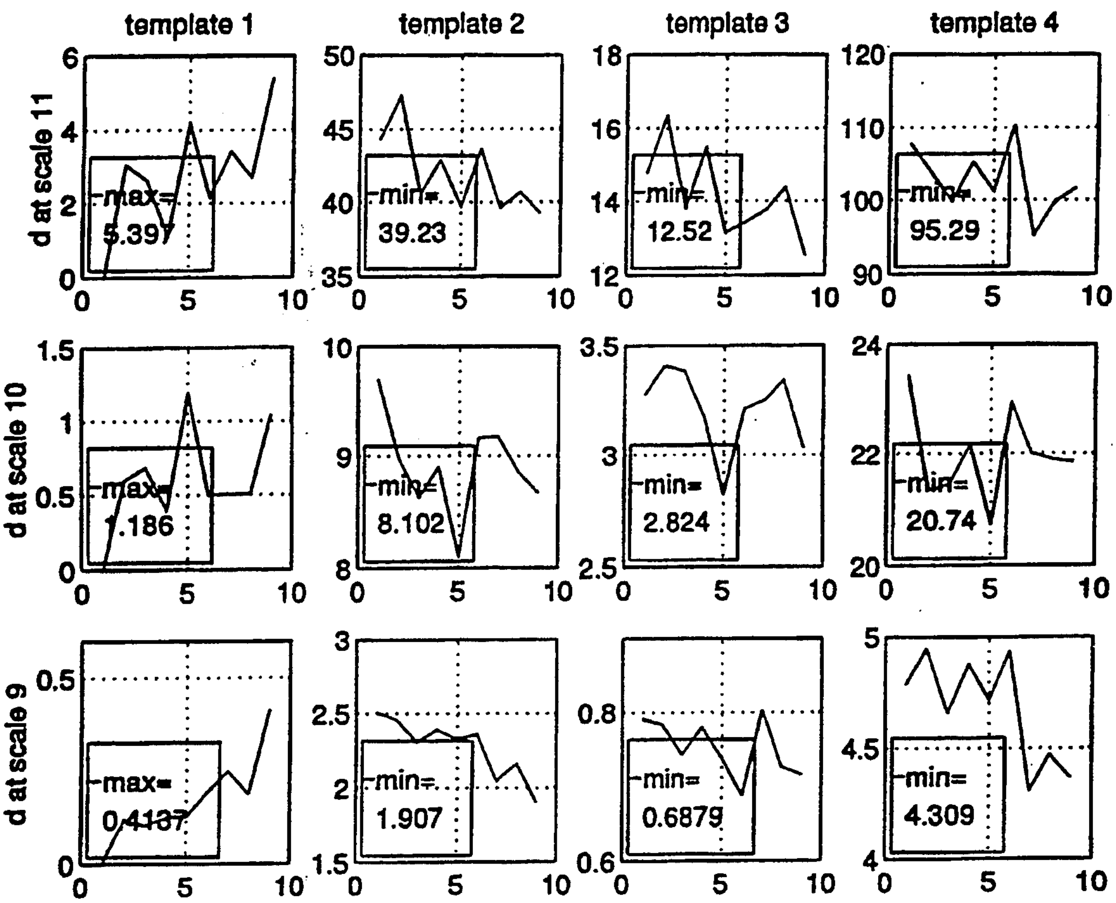


Figure C.9 Distance measures at the output of templates when signals from Transmitter 1 are the inputs. Horizontal axis shows the number of the signals. Each column represents a template. First row is for Scale 11; second row is for Scale 10; third row is for Scale 9.

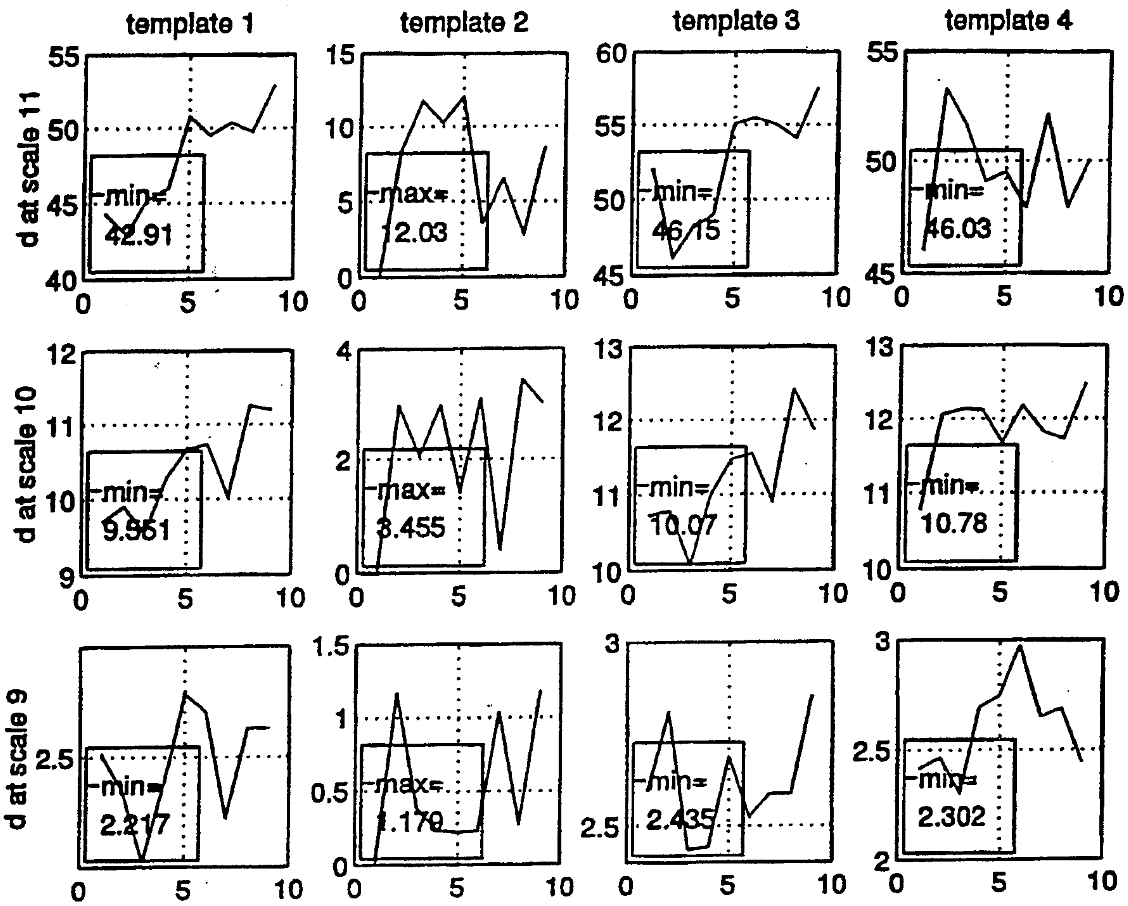


Figure C.10 Distance measures at the output of templates when signals from Transmitter 2 are the inputs. Horizontal axis shows the number of the signals. Each column represents a template. First row is for Scale 11; second row is for Scale 10; third row is for Scale 9.

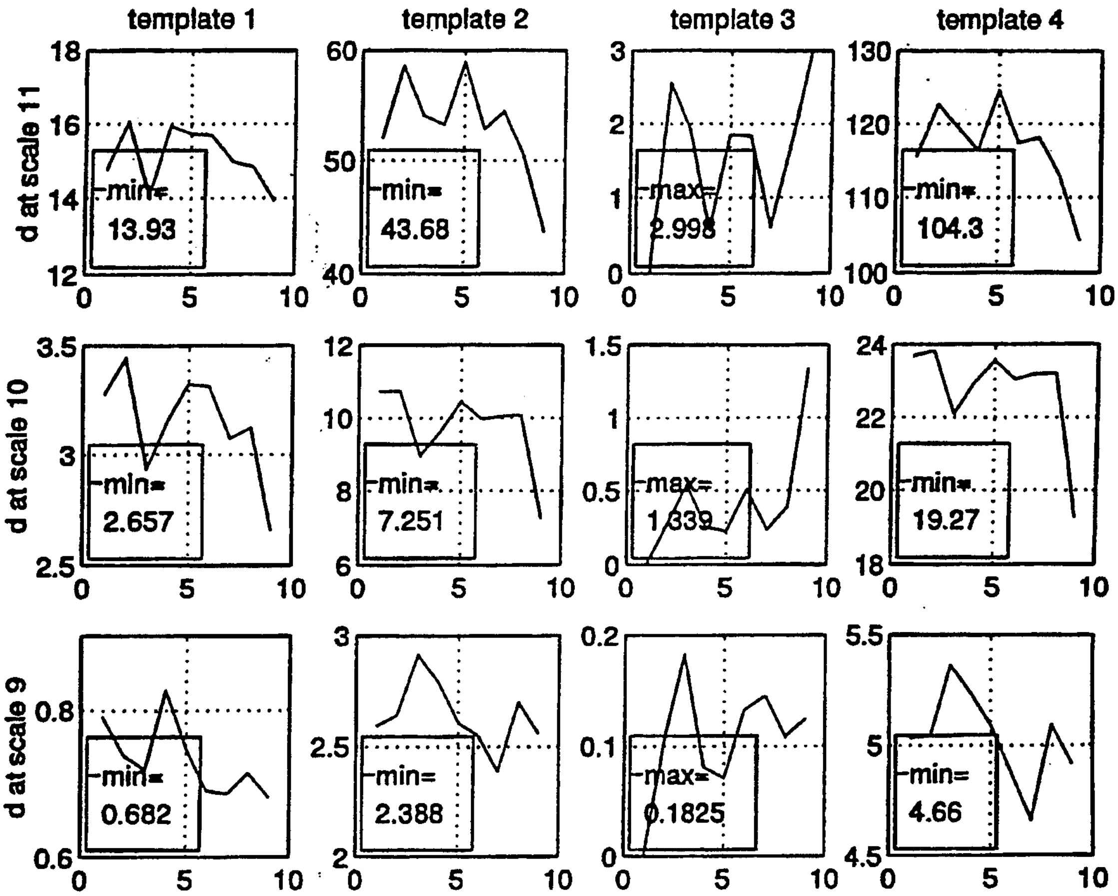


Figure C.11 Distance measures at the output of templates when signals from Transmitter 3 are the inputs. Horizontal axis shows the number of the signals. Each column represents a template. First row is for Scale 11; second row is for Scale 10; third row is for Scale 9.

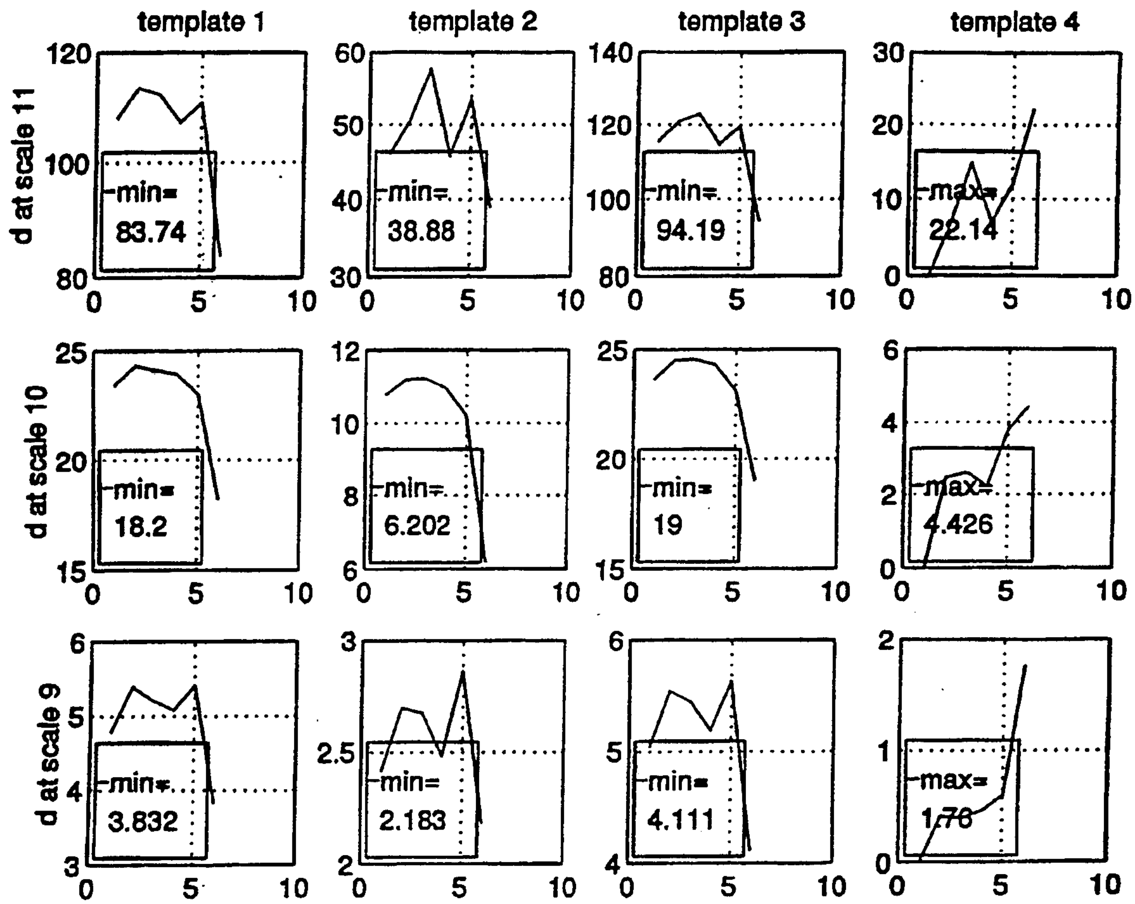


Figure C.12 Distance measures when signals from Transmitter 4 are the inputs. Horizontal axis shows the number of the signals. Each column represents a template. First row is for Scale 11; second row is for Scale 10; third row is for Scale 9.

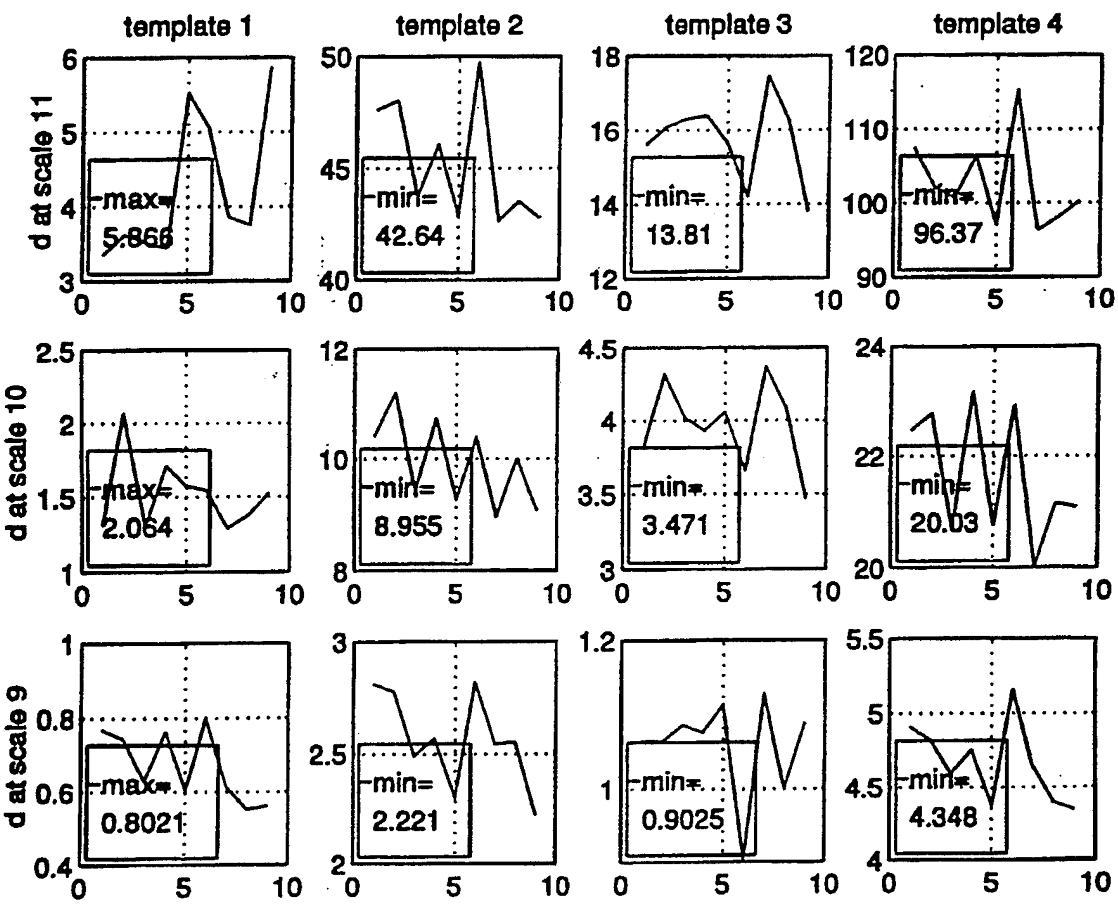


Figure C.13 Distance measures when signals from Transmitter 1 are inputs and their SNR values are 10 dB lower than the original data set. Horizontal axis shows the number of the signals. Each column represents a template. First row is for Scale 11; second row is for Scale 10; third row is for Scale 9.

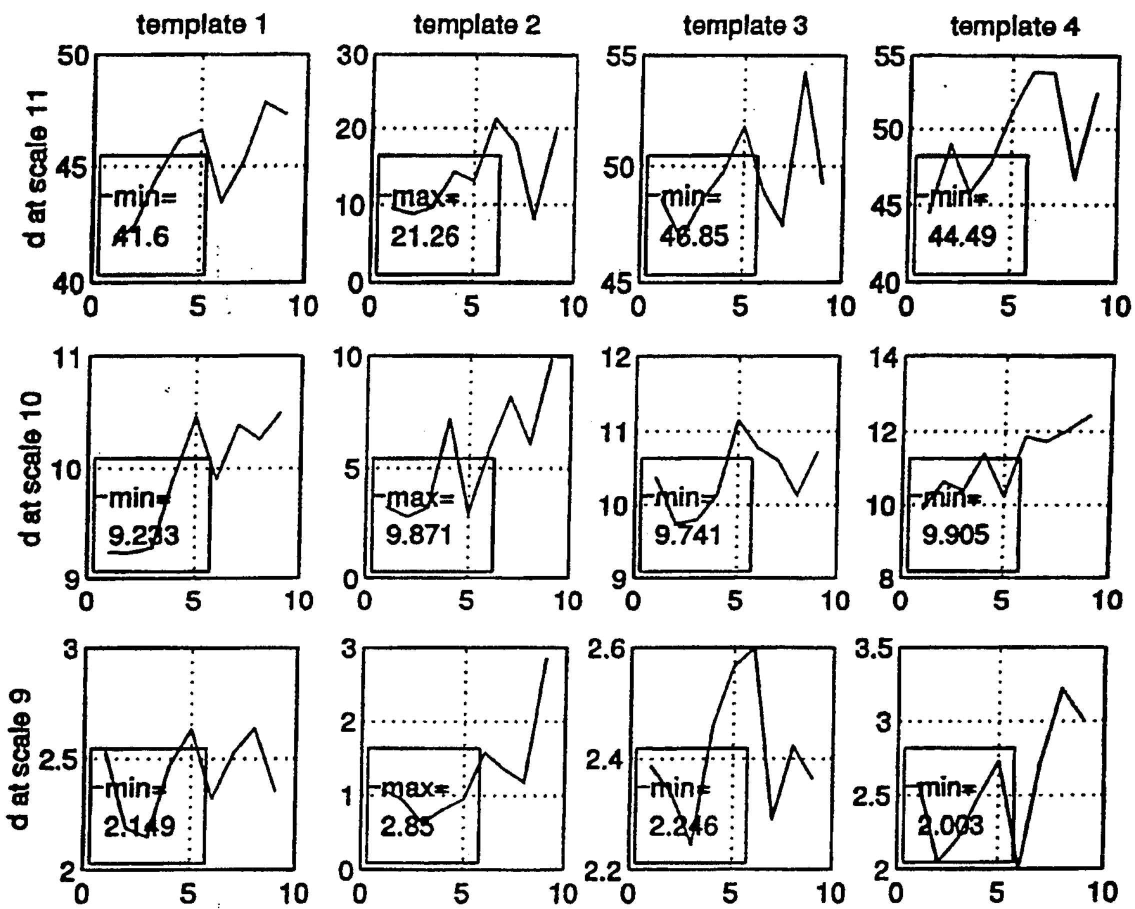


Figure C.14 Distance measures when signals from Transmitter 2 are inputs and their SNR values are 10 dB lower than the original data set. Horizontal axis shows the number of the signals. Each column represents a template. First row is for Scale 11; second row is for Scale 10; third row is for Scale 9.

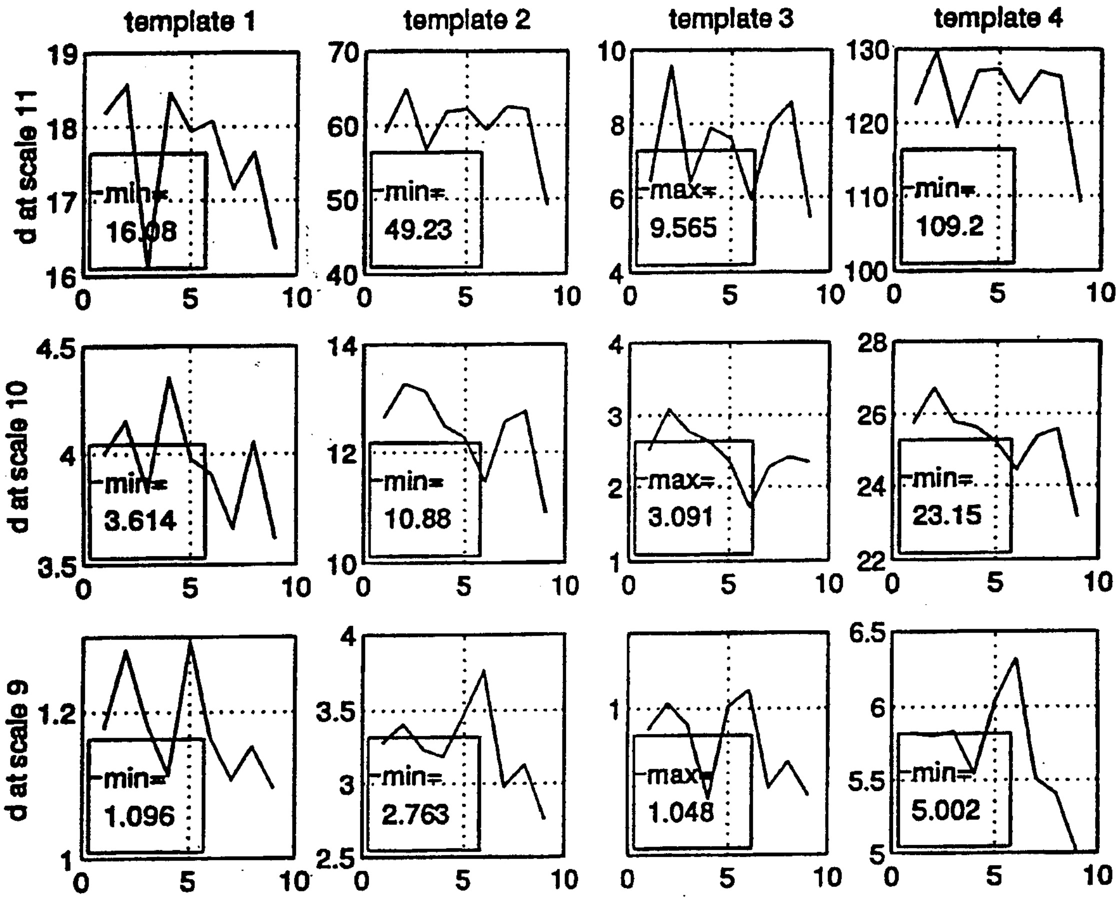


Figure C.15 Distance measures when signals from Transmitter 3 are inputs and their SNR values are 10 dB lower than the original data set. Horizontal axis shows the number of the signals. Each column represents a template. First row is for Scale 11; second row is for Scale 10; third row is for Scale 9.

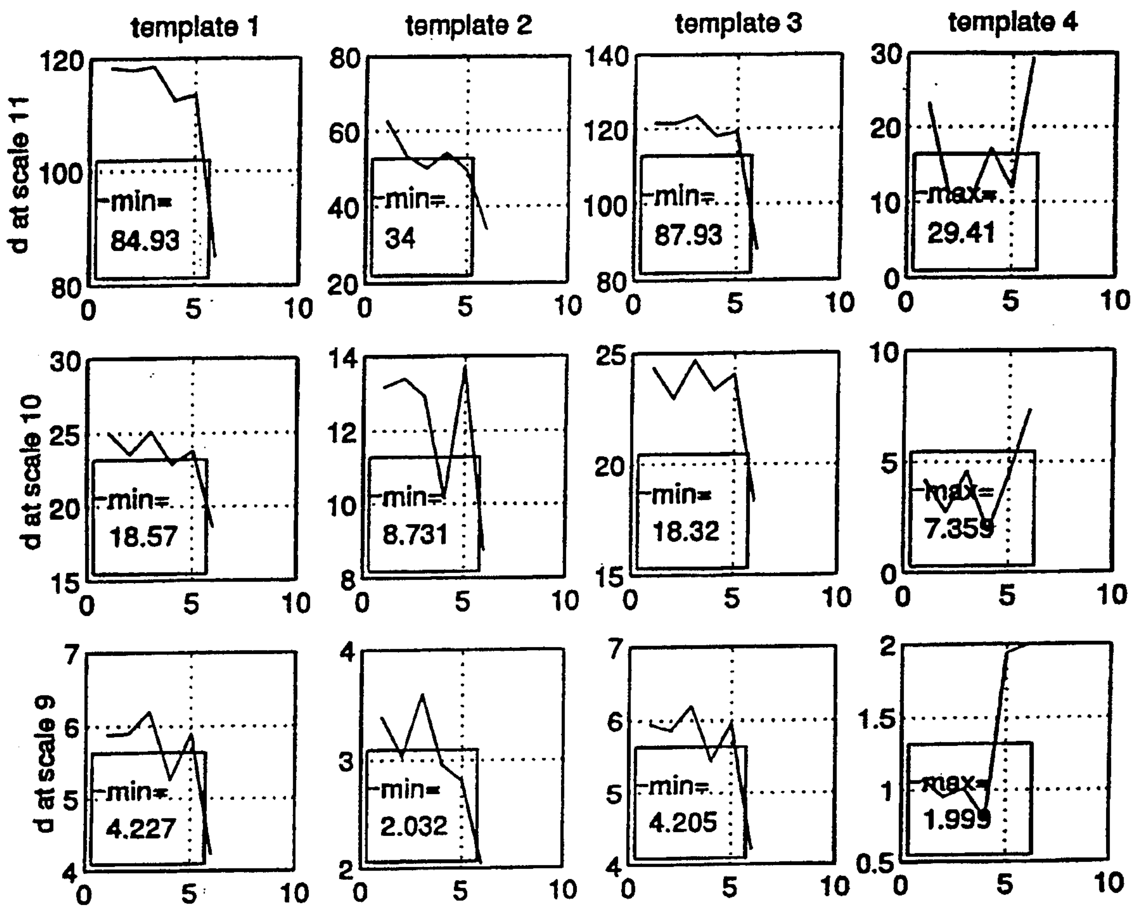


Figure C.16 Distance measures when signals from Transmitter 4 are inputs and their SNR values are 10 dB lower than the original data set. Horizontal axis shows the number of the signals. Each column represents a template. First row is for Scale 11; second row is for Scale 10; third row is for Scale 9.

APPENDIX D

MATLAB CODES

wtsignal.m

Purpose:
To transform the push-to-talk recordings into pulse shape forms which are suitable for wavelet analysis.

Synopsis:
wtsignal

Description:
The program preprocesses the data as mentioned in Chapter 3. The signal vector to be preprocessed is entered as the input.
Created by Yalcin Payal, November, 1995.

```
x=input('enter the signal to be preprocessed = ')
x=x-mean(x);
y=envelope(x);
my=mdsmooth(y,100);
dmy=diff(my);
mdmy=mdsmooth(dmy,50);
end
```

distance.m

Purpose:

To compute distance measure between the signal and the four templates. Signals are 4096 points long.

Synopsis:

distance

Description:

Wavelet transform matrix of the signal and the templates are the inputs to the program. Program is written to work with the wavelet transform matrices produced by the subroutine "mapdn.m" The subroutine "mapdn.m" produces the magnitude square of the wavelet coefficients at each scale, "distance.m" needs the amplitudes of the wavelet coefficients. Thus, a modification to "mapdn.m" is essential prior to run "distance.m." mapdn.m and other functions called by mapdn.m can be found in (Newland, 1992) or in (Pitta, 1995). d1, d2, d3, d4 are the outputs of the program and they are distance measures between the signal and Template 1, Template 2, Template 3, Template 4, respectively at each scale.
Created by Yalcin Payal, November, 1995.

```
clear, clg
s1=input(' enter the WT of the signal to be identified = ');
kk1=input('enter the WT of the template 1= ');
mm1=input('enter the WT of the template 2= ');
tt1=input('enter the WT of the template 3= ');
vv1=input('enter the WT of the template 4= ');
for n=1:11
t=length(kk1(1,:))/(2^n);
col=n+2;
a=kk1(col,1:t:length(kk1(1,:)));
b=mm1(col,1:t:length(mm1(1,:)));
c=tt1(col,1:t:length(tt1(1,:)));
f=vv1(col,1:t:length(vv1(1:)));
s=s1(col,1:t:length(s1(1,:)));
a=localext(a);
```

```
b=localext(b);
c=localext(c);
f=localext(f);
d=localext(s);
[temp1,i]=sort(a);
[temp2,j]=sort(b);
[temp3,k]=sort(c);
[temp4,k]=sort(f);
[x,m]=sort(d);
shift1=m(length(x))-i(length(a));
shift2=m(length(x))-j(length(b));
shift3=m(length(x))-k(length(c));
shift4=m(length(x))-q(length(f));
w1=[abs(m-i-shift1)];
w2=[abs(m-j-shift2)];
w3=[abs(m-k-shift3)];
w4=[abs(m-q-shift4)]; w1(find(w1==0))=ones(1,length(find(w1==0)));
w2(find(w2==0))=ones(1,length(find(w2==0)));
w3(find(w3==0))=ones(1,length(find(w3==0)));
w4(find(w4==0))=ones(1,length(find(w4==0)));
d1(n)=sum(sqrt(w1.*(temp1-x)^2));
d2(n)=sum(sqrt(w2.*(temp2-x)^2));
d3(n)=sum(sqrt(w3.*(temp3-x)^2));
d4(n)=sum(sqrt(w4.*(temp4-x)^2));
end
```

localext.m

Purpose:

To take the local extrema of the signal.

Synopsis:

localext.

Description:

LOCALEXT extracts the local extrema of a vector. LOCALEXT (X) returns a copy of vector X with all non extreme values set to 0.

[Y,K] = LOCALEXT (X) will also return the number of deleted samples of X.

function [ou,k]=localext(sg)

Author: Sergio J. Garcia Galan, e-mail: Uvi_Wave@tsc.uvigo.es

```
l=length(sg);
sg=sg(:)';
ou=zeros(1,l);
ax=abs(sg);
MX = max(ax);
MX=MX+1;
ou(1)=sg(1);
ou(l)=sg(l);
k=0;
for i=2:l-1
        if (sg(i) > sg(i-1)) & (sg(i) > sg(i+1))
                ou(i)=sg(i);
                k=k+1;
        end
        if (sg(i) < sg(i-1)) & (sg(i) < sg(i+1))
                ou(i)=sg(i);
                k=k+1;
        end
end
k=l-k;
```

envelope.m

Purpose:

To take the envelope of the push-to-talk communication signals.

Synopsis:

envelope .

Description:

[Y,M]=ENVELOPE (x) returns the envelope , Y , and modulation index, M, of the AM signal in vector x.

Created by Dennis W. Brown, 1993.
Naval Postgraduate School, Monterey, CA.
May be freely distributed; not for use in commercial applications.
Part of the SPC Toolbox (Brown, 1995).

```
function [y,m] = envelope(x)
y = [ ];
if nargin  = 1,
        error('envelope: One argument required...');
end
if min(size(x))  = 1,
        error('envelope: Input arg "x" must be a 1xN or Nx1 vector.');
end;
x = x(:);
y = abs(hilbert(x));
mmax = max(y);
mmin = min(y);
m = (mmax - mmin) / (mmax + mmin);
```

mdsmooth.m

Purpose:

To filter the envelope of the push-to-talk communication signals and the differential of the envelopes by median filtering.

Synopsis:

mdsmooth.

Description:

[y] = MDSMOOTH(X,L) smooths the input vector X using amedian filter with a rectangular window of "L" samples.

MDSMOOTH is implemented as a mex function on someinstallations.

Created by LT Dennis W. Brown

Naval Postgraduate School, Monterey, CA

May be freely distributed; not for use in commercial applications.

Part of the SPC Toolbox (Brown, 1995).

```
function [y] = mdsmooth(x,L)
y = [];
if nargin  = 2,
        error('mdsmooth: Invalid number of input arguments...');
end;
if min(size(x))  = 1,
        error('mdsmooth: Input arg "x" must be a 1xN or Nx1 vector.');
end;
x = x(:);
Ns = length(x);
y = zeros(Ns,1);
x = [zeros(L/2-1,1); x ; zeros ; zeros(L/2,1)];
for k=1:Ns
        y(k,1) = median(x(k:k+L-1,1));
end
```

LIST OF REFERENCES

Akansu, A.N. and Haddad, R.A., *Multiresolution Signal Decomposition,* Academic Press, Inc., San Diego, CA, 1992.

Aware, Inc., "Wavelet Signal Processing for Transient Feature Extraction," Technical Report, AD 920315, March, 1992.

Brown, D.W., "SPC Toolbox: A Matlab-Based Software Package for Signal Analysis," Master's Thesis, Naval Postgraduate School, 1995.

Burrus, C.S., and Gopinath, R.A., "Introduction to Wavelets and Wavelet Transforms," Tutorial #1, *IEEE ICASSP '93,* Minneapolis, MN, April, 1993.

Carter, P.H. "Unknown Transient Detection using Wavelets," *SPIE Proceedings,* Vol. 2242, pp. 803-814, 1994.

Ehara, N., Sasase, I., and Mori, S., "Weak Radar Signal Detection Based on Wavelet Transform," *IEEE Signal Processing,* Vol. 2, pp. 377-380, 1994.

Gabor, D., "Theory of Communication," *Journal of IEEE,* Vol. 93, pp. 429-457, 1946.

Hippenstiel, R., Wavelet Course Class Notes, Naval Postgraduate School, Monterey, CA, 1995.

Mallat, S.G., and Zhoung, S., "Complete Signal Representation with Multiscale Edges," Technical Report 483, Courant Insitute of Mathematical Sciences, New York University, December, 1989.

Mathworks, MATLAB Reference Guide, August, 1992.

Newland, D.E., "An Introduction to Random Vibrations," *Spectral and Wavelet Analysis,* Longman Scientific & Technical, John Wiley & Sons, Inc., New York, 1992.

Pitas, I., and Venetsanopoulos, A.N., *Nonlinear Digital Filters,* Kluwer Academic Publishers, Boston, 1990.

Pitta, J.R., "Transient Detection Using Wavelets," Master's Thesis, Naval Postgraduate School, Monterey, CA, 1995.

Shensa, M.J., "Discrete Wavelet Transform: Wedding A Trous and Mallat Algorithms," *IEEE Transactions on Signal Processing,* Vol 40, p 2464- 2482, 1992.

Strang, G., *Linear Algebra and Its Applications,* Academic Press, New York, 1976.

Vetterli, M., and Kovacevic, J., *Wavelets and Subband Coding*, Prentice Hall, Inc., Englewood Cliffs, NJ, 1995.

Weiss, L.G., "Wavelets and Wideband Correlation Process," *IEEE Signal Processing,* pp.13-32, January, 1994.

Young, R.K., *Wavelet Theory and its Applications,* Kluwer Academic Publishers, Boston, 1993.

INITIAL DISTRIBUTION LIST

		No.Copies
1.	Defense Technical Information Center 8725 John J. Kingman Rd. , STE 0944 Ft. Belvoir, VA 22060-6218	2
2.	Library Code, 013 Naval Postgraduate School Monterey, CA 93943-5101	2
3.	Chairman, Code EC Department of Electrical and Computer Engineering Naval Postgraduate School Monterey, CA 93943-5121	1
4.	Professor R. Hippenstiel, Code EC/Hi Department of Electrical and Computer Engineering Naval Postgraduate School Monterey, CA 93943-5121	1
5.	Professor M. P. Fargues, Code EC/Fa Department of Electrical and Computer Engineering Naval Postgraduate School Monterey, CA 93943-5121	1
6.	Deniz Kuvvetleri Komutanlığı Personel Daire Başkanlığı Bakanlıklar, Ankara, 06600 TURKEY	2
7.	Yalcin Payal Cumhuriyet cad. No:49 Golcuk-Degirmendere , KOCAELI 41950 TURKEY	2