Automatic Radar Waveform Recognition

Jarmo Lundén, *Student Member, IEEE*, and Visa Koivunen, *Senior Member, IEEE*

***Abstract*—In this paper, a system for automatically recognizing radar waveforms is introduced. This type of techniques are needed in various spectrum management, surveillance and cognitive radio or radar applications. The intercepted radar signal is classified to eight classes based on the pulse compression waveform: linear frequency modulation (LFM), discrete frequency codes (Costas codes), binary phase, and Frank, P1, P2, P3, and P4 polyphase codes. The classification system is a supervised classification system that is based on features extracted from the intercepted radar signal. A large set of potential features are presented. New features based on Wigner and Choi–Williams time-frequency distributions are proposed. The feature set is pruned by discarding redundant features using an information theoretic feature selection algorithm. The performance of the classification system is analyzed using extensive simulations. Simulation results show that the classification system achieves overall correct classification rate of 98% at signal-to-noise ratio (SNR) of 6 dB on data similar to the training data.**

***Index Terms*—Pulse compression, radar, spectrum management, waveform recognition.**

I. Introduction

AGILE SENSING of the electromagnetic spectrum, signal environment and recognition of the employed waveforms are crucial for efficient and effective operation of the future communication and radar systems such as cognitive radios and radars. The ever-increasing number of different communication and radar emitters and waveforms as well as increasing data rate demands in communication systems require efficient and agile utilization of the electromagnetic spectrum including unlicensed spectrum. Automatic radar waveform recognition is very important task in intercept receivers for electronic warfare (EW) applications and spectrum management. Applications include, for example, threat recognition and analysis, construction of effective jamming responses, and radar emitter identification and cognitive radar.

Automatic waveform and modulation recognition has received increased interest during the past few years in communications and radar communities. In radar context, an atomic decomposition (AD) based radar signal interception and waveform recognition method employing chirplet dictionary has been proposed in [1]. The method is especially suitable for linear frequency modulation (LFM). It can, however, be used also for other modulations as demonstrated by the numerical experiments in [1]. In [2] the method is improved by the use of expectation maximization (EM) algorithm. The improved algorithm obtains a sparser representation than the original algorithm. Another similar approach using short-time Fourier transform (STFT) is presented in [3]. Both methods consider classification of only few different waveforms.

The objective of the waveform recognition system is to detect and classify the intercepted radar signals based on the pulse compression waveform. In this paper we develop a supervised classification approach where features extracted from the intercepted radar pulse are fed into a supervised classifier making the decision. The intercepted radar pulses are classified to eight classes: LFM, discrete frequency codes (Costas codes), binary phase, and Frank, P1, P2, P3, and P4 polyphase codes. Supervised classifier determines the decision regions automatically from the training data during the off-line training phase. This facilitates the use of large number of features and classes. In [4], a diagram of a radar waveform classification system was illustrated that employs a supervised classifier using features based, for example, on time-frequency distributions and cyclostationary spectral analysis. In this paper, a supervised classification system using features based on, e.g., second order statistics, instantaneous signal properties, and time-frequency distributions is presented. The supervised classification approach has been previously found successful in automatic modulation recognition of communication signals, e.g., [5]–[7]. Automatic modulation recognition of communication signals is closely related to automatic radar waveform recognition. However, the characteristics of the waveforms are considerably different between the communication and radar signals. Consequently, the employed features and algorithms for their extraction as well as classifier structures have to be selected and designed differently.

The contributions of this paper are as follows. A supervised classification system for classifying common pulse compression radar waveforms is systematically developed. Eight different waveform classes including five polyphase classes are considered. No such extensive families of waveforms have been classified in literature previously. A large set of potential features are presented. Potential features from the communication signal modulation recognition literature have been gathered and are presented. Novel features calculated from the instantaneous frequency estimated from Wigner distribution with adaptive data-driven window length are proposed. Novel features based on Choi–Williams distribution are proposed as well. A normalization procedure enabling effective feature extraction from Choi–Williams distribution images is proposed. The feature set

Manuscript received August 30, 2006; revised February 21, 2007. This work was supported by the Finnish Defence Forces Technical Research Centre. The work of J. Lundén was also supported by GETA Graduate School and Nokia Foundation. Some preliminary results were presented in part at the CISS'05 Conference, Baltimore, MD, in March 2005, and at the IEEE MLSP'05 workshop, Mystic, CT, in September 2005. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Fulvio Gini.

The authors are with the Signal Processing Laboratory, SMARAD CoE, Helsinki University of Technology, Helsinki, FI-02015 TKK, Finland (e-mail: jrlunden@wooster.hut.fi; jarmo.lunden@tkk.fi; visa.koivunen@hut.fi; visa.koivunen@tkk.fi).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JSTSP.2007.897055

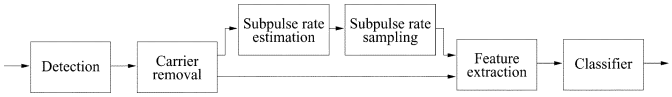


Fig. 1. Block diagram of the waveform recognition system.

is pruned by removing redundant features using an information theoretic feature selection algorithm. A parallel classifier structure based on multilayer perceptron (MLP) networks is proposed. Two robust classifiers, the early-stop committee and the Bayesian MLP, are employed. The classifier performances are tested in scenarios with large random variation in the signal parameters. The classifiers are shown to achieve very reliable performance: over 98% correct classification rate at SNR of 6 dB.

The paper is organized as follows. Section II introduces the recognition system and the signal model. The supervised classifier is presented in Section III. Section IV explains briefly the employed feature selection algorithm, and Section V presents the features. Section VI shows the simulation results. Finally, Section VII gives the concluding remarks.

II. Recognition System Overview

The objective of the waveform recognition system is to detect and classify the intercepted radar pulses based on the pulse compression waveform. In this study, the intercepted waveforms are classified to eight classes: LFM, discrete frequency codes (Costas codes), binary phase, and Frank, P1, P2, P3, and P4 polyphase codes.

Fig. 1 depicts the operation of the waveform recognition system. First the signal is detected. After that the carrier frequency is estimated and removed. Here, the carrier frequency is defined as the center frequency of the signal's frequency band.

The focus on this work was on the classification part of the recognition system. Thus, in general the first two stages of the recognition system (i.e., detection and carrier frequency removal) are assumed to be accomplished. As a result, the complex envelope of the intercepted radar signal is obtained. In addition, the channel is assumed to be additive white Gaussian noise (AWGN) channel. That is, the intercepted discrete time signal model is given by

$$y(k) = x(k) + n(k) = Ae^{j\phi(k)} + n(k) \tag{1}$$

where $y(k)$ and $x(k)$ are the complex envelopes of the intercepted and transmitted signals, respectively, and $n(k)$ is a complex circular white Gaussian noise process. A is constant amplitude and $\phi(k)$ is the instantaneous phase of the complex envelope. The signal is assumed to be a single pulse consisting of a single code period from a single radar emitter.

In one of the simulation experiments, instead of assuming the carrier frequency to be known it is estimated in order to get an indication how much the carrier frequency estimation affects the performance.

Many of the employed features are calculated directly from the complex envelope. However, in order to extract detailed properties of the polyphase coded waveforms, the subpulse rate needs to be estimated. The radar signal is then sampled at the subpulse frequency. The cyclostationarity of the phase coded waveforms is utilized in the subpulse rate estimation. A cyclic correlation based symbol rate estimator is employed [8]. The subpulse rate estimation and sampling complete the preprocessing stage of the recognition system.

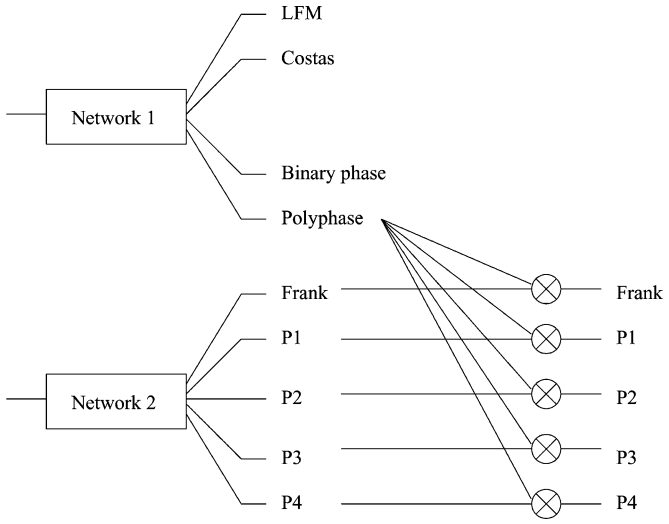


Fig. 2. Waveform classifier comprising two parallel independent multilayer perceptron networks.

After the preprocessing is completed, the features are calculated. Finally, the calculated feature vectors are inserted to the waveform classifier that performs the classification.

III. Waveform Classifier

Due to a large number of potential features and classes, a supervised classifier is a well-suited approach for obtaining good classification performance. During the off-line training phase the supervised classifier determines the decision boundaries from the training data. The employed classifier model is the MLP. MLPs with sigmoidal hidden unit activation functions are universal approximators [9]. Thus, in theory MLPs can approximate any decision boundary to arbitrary accuracy.

Fig. 2 shows the structure of the supervised waveform classifier. It consists of two independently operating parallel MLP networks. The MLP networks have different independently selected input feature vectors (some of the features may be same). That is, the considered feature set for the network 2 intended for polyphase signals contains features calculated from the subpulse rate sampled complex envelope as well.

The rationale for the classifier structure is that subpulse rate sampling is not feasible for the frequency modulated waveforms (especially for LFM which does not have any subpulses). Hence, the classifier is divided to two parts in order to guarantee that the features calculated from the subpulse rate sampled signal do not affect the training nor classification of the frequency modulated waveforms.

The classifier structure stems from the hierarchical classifier structure used in [5] for automatic modulation classification of communication signals. However, instead of making a hard decision about the class in this work the outputs of the proposed classifier structure approximate the posterior probabilities of the output classes given the input feature vectors. In order to ensure

that the outputs can be interpreted as the posterior probabilities of the classes (given the input feature vector) the characteristics of the MLPs have to be chosen appropriately. Choosing the cross-entropy cost function, hyperbolic tangent hidden layer activation function and softmax output activation function allow the outputs to be interpreted as the posterior probabilities of the classes given the input vector [9], [10].

The property that the outputs approximate the posterior probabilities of the classes offers many benefits. For example, it allows the possibility of using different prior probabilities than the ones represented by the training data. Hence, it enables adaptation to changing signal environments. Another important benefit is that the posterior probabilities give a characterization of the correctness of the decision. Consequently, if none of the posterior probabilities exceeds a certain predefined threshold the decision can be rejected.

The final classification decision is based on the posterior probabilities. That is, the class with the highest posterior probability is chosen. The first three outputs of the network 1 approximate directly the LFM, Costas, and binary phase class posterior probabilities. The approximations for the posterior probabilities of the polyphase classes are obtained by multiplying the fourth output of the network 1 (i.e., combined polyphase output) with the network 2 outputs (see Fig. 2).

The standard feed-forward MLP using conventional training (i.e., parameter optimization) is prone to overfit to the training data. Thus, two more complex MLP classifiers that provide better solutions against overfitting are employed. The classifiers are the ensemble averaging early-stop committee (ESC) and the Bayesian MLP. The idea is that both networks in Fig. 2 are either ESCs or Bayesian MLPs. ESC is a simple yet very robust classifier. The goal in early-stopping is to stop the training (using part of the training data for validation) before the network overfits to the training data. Using a committee of early-stop MLPs provides robustness against initial conditions of the weights. Bayesian MLP is much more complex than ESC. In Bayesian inference, instead of using one set of weights (as in conventional training), all the possible parameter values weighted with their posterior probabilities are used for prediction. This, in principle, prevents overfitting. For more information about MLPs, early-stopping, and committee machines, see e.g., [9]. A good reference for Bayesian neural networks is [11].

IV. Feature Selection

In this section, the feature selection algorithm for selecting the final feature vectors is briefly introduced. The algorithm will be used to select the final feature vectors from a large set of different features that will be presented in the next section. The goal in feature selection is to remove redundant features in order to reduce complexity of the computation and training.

The employed feature selection algorithm has been proposed in [12]. It is based on the mutual information between the classes and the features. The algorithm selects the features one by one using greedy selection. That is, at each step the feature vector dimension is increased by one. The new selected feature is the one that maximizes the mutual information between the classes and the current feature vector (which includes the previously selected features as well as the new feature). This procedure is continued until the change in the estimated mutual information becomes smaller than a predefined threshold. Mutual information between the classes C and the feature vector $\boldsymbol{X}$ is estimated by [12]

$$\hat{I}(\boldsymbol{X};C) = H(C) - \hat{H}(C\,|\,\boldsymbol{X}) = -\sum_{k=1}^{c} p(c_k)\log p(c_k) + \frac{1}{N}\sum_{j=1}^{N}\sum_{k=1}^{c} \hat{p}(c_k|\boldsymbol{x}_j)\log \hat{p}(c_k|\boldsymbol{x}_j) \quad (2)$$

where c_k is the kth class and c is the total number of classes. $H(\cdot)$ denotes the entropy. N is the number of training data vectors and $\boldsymbol{x}_j$ is the jth training data vector (i.e., feature vector calculated from the jth training data pattern). Probability density estimates $\hat{p}(c_k\,|\,\boldsymbol{x})$ are obtained using Parzen windows. Entropy of the class variable is easily calculated using the training data, i.e., using $p(c_k) = N_k/N$ where N_k is the number of training data vectors from class c_k.

The above feature selection method is computationally very expensive. However, the computational complexity is not a limiting factor in practice since feature selection is an off-line procedure. Note, however, that for a higher number of original features and larger training data sets computational complexity can become a limiting factor even in off-line procedures.

V. Feature Extraction

In this section the considered features and algorithms for their extraction are presented. The section is organized as follows. First the features based on second order statistics are described. Then the features based on power spectral density (PSD) are presented. After that the features based on instantaneous signal properties are given. Three new features are proposed, especially, for discrimination between the P1 and P4 codes. Finally, new features based on Choi–Williams time-frequency distribution are introduced.

Table I lists the features and indicates the network they are considered for. The requirement for subpulse rate sampling is indicated as well. Table I lists only the features presented in this section. In addition, a set of other features were considered as well. These features include higher-order moments (up to 8th order) and cumulants (up to 6th order) as well as few other features, such as PSD symmetry. However, in order to keep the presentation as concise as possible these features are not presented here since they were discarded in the beginning of the feature selection process due to the fact that they were not found discriminative enough. This process will be explained more thoroughly in the simulation section when the feature selection results are presented.

A. Second Order Statistics

1) Moments and Cumulants of the Complex Envelope: Moments and cumulants of the complex envelope characterize the distribution of the complex constellation. In particular, the second order moments and cumulants are very suitable for binary phase signal recognition. This is due to the property that the squared complex envelope of a binary phase signal is constant.

TABLE I
LIST OF FEATURES. BULLETS INDICATE SUITABILITY FOR THE NETWORKS AND SUBPULSE RATE SAMPLING

#	Feature	Net 1	Net 2	Subpulse
1	Moment $\hat{M}_{10}$	•	•	• (Net 2 only)
2	Moment $\hat{M}_{20}$	•	•	• (Net 2 only)
3	Cumulant $\hat{C}_{20}$	•	•	• (Net 2 only)
4	Time lag $t_{\max}$		•	•
5	PSD maximum $\gamma_{\max}$	•	•	
6	PSD squared maximum $\gamma_{2,\max}$	•	•	
7	Std of inst. phase $\hat{\sigma}_\phi$	•	•	
8	Std of inst. freq. $\hat{\sigma}_f$	•		
9	Wigner inst. freq. std σ_{Wf}		•	
10	Wigner inst. freq. autocorr. r		•	
11	Wigner inst. freq. runs p_R		•	
12	Pseudo-Zernike moment $\hat{Z}_{20}$		•	
13	Pseudo-Zernike moment $\hat{Z}_{22}$		•	
14	Pseudo-Zernike moment $\hat{Z}_{30}$		•	
15	Pseudo-Zernike moment $\hat{Z}_{31}$		•	
16	Pseudo-Zernike moment $\hat{Z}_{32}$		•	
17	Pseudo-Zernike moment $\hat{Z}_{33}$		•	
18	Pseudo-Zernike moment $\hat{Z}_{43}$		•	
19	Number of CWD objects N_{obj}	•	•	
20	CWD peak time location $t_{\max}$	•	•	
21	CWD object width std σ_{obj}	•	•	

The nth order zero-lag moment of the complex envelope of a complex random process $y(k)$ may be estimated as

$$\hat{M}_{nm} = \left| \frac{1}{N} \sum_{k=0}^{N-1} y^{n-m}(k)(y^*(k))^m \right| \tag{3}$$

where N is the number of data samples and m is the number of conjugated components. The absolute value is taken in order to make the estimate invariant to constant phase rotation. Scaling invariance can be achieved by normalizing $y(k)$ prior to applying (3). That is, scaling invariance is achieved by (assuming zero mean additive noise)

$$\tilde{y}(k) = \frac{y(k)}{\sqrt{\hat{M}_{21} - \sigma_n^2}} \tag{4}$$

where σ_n^2 is the variance of the additive noise. The normalization requires an estimate of the variance of the noise. In practice, such estimate may be easily obtained.

The first and second order moments $\hat{M}_{10}$ and $\hat{M}_{20}$ as well as the second order cumulant $\hat{C}_{20}$ are used as features. $\hat{C}_{20}$ is calculated similarly as $\hat{M}_{20}$ except that the mean $\hat{M}_{10}$ is first subtracted from $y(k)$. Selected moments and cumulants are unaffected by additive independent complex second order circular noise. For the network 2 these features are calculated from the subpulse rate sampled signal denoted by $y_s(k)$.

2) Cross-Correlation Time Lag: A discriminating feature among the polyphase codes is obtained by calculating the cross-correlation between the subpulse rate sampled pulse and its time-reversed version. Time lag of the maximum cross-correlation is used as a feature, i.e., $\tau_{\max} = \arg\max_\tau |\hat{r}_y(\tau)|$ where $\hat{r}_y(\tau)$ is the cross-correlation given by

$$\hat{r}_y(\tau) = \begin{cases} \sum_{k=0}^{N-\tau-1} y_s(k+\tau) y_s^*(N-1-k), & \tau \geq 0, \\ \sum_{k=-\tau}^{N-1} y_s(k+\tau) y_s^*(N-1-k), & \tau < 0 \end{cases} \tag{5}$$

for all $|\tau| \leq N-1$. Here, $y_s(k)$ denotes the subpulse rate sampled complex envelope. The feature is invariant to constant phase rotation whereas carrier frequency offset can cause problems.

B. Power Spectral Density (PSD) Based Features

PSD describes how the signal power is distributed in frequency domain. Two features based on PSD are utilized. The first feature is the maximum of the PSD of the complex envelope (using periodogram) [13]

$$\gamma_{\max} = \frac{1}{N} \max_n \left\{ \frac{1}{N} \left| \sum_{k=0}^{N-1} \tilde{y}(k) e^{-j2\pi nk/N} \right|^2 \right\} \tag{6}$$

where $\tilde{y}(k)$ is magnitude normalized complex envelope given by (4). This feature is suitable for discriminating the binary phase and Costas codes from the rest of the codes. Normalization by $1/N^2$ provides invariance with respect to the number of data samples.

The second feature is the maximum of the PSD of the squared complex envelope, i.e., in (6) $\tilde{y}(k)$ is replaced with $\tilde{y}^2(k)$. The feature is denoted by $\gamma_{2,\max}$. Due to the fact that the squared complex envelope is constant for binary phase signals, this feature is very good for recognizing the binary phase signals.

C. Features Derived From Instantaneous Signal Properties

The instantaneous properties of the radar signals are very distinctive especially between frequency and phase modulations. Firstly, two features based on the direct estimate of the instantaneous phase (i.e., the phase of the complex envelope) are given. Secondly, in order to improve the discrimination between the P1 and P4 codes an instantaneous frequency estimate with appropriate properties is calculated using Wigner distribution with adaptive data-driven window length, as well. Three features based on this instantaneous frequency estimate are proposed.

Standard deviation of the absolute value of the instantaneous phase is estimated as [14]

$$\hat{\sigma}_\phi = \sqrt{\frac{1}{M} \left(\sum_k \phi^2(k) \right) - \left(\frac{1}{M} \sum_k |\phi(k)| \right)^2} \tag{7}$$

where $\phi(k)$ is the instantaneous phase sequence estimated directly from the complex envelope and confined between $-\pi$ and π. M is the number of non-weak samples, i.e., samples whose amplitude is larger than some predefined threshold. In the simulation experiments, the employed threshold was 0.2 of the maximum amplitude. Note that the sums are taken over the non-weak samples, as well. The objective is to reduce the sensitivity of the estimate to noise.

Derivative of the instantaneous phase with respect to time is the instantaneous frequency. It can be approximated using finite differences, i.e., $f(k) = \phi_u(k+1) - \phi_u(k)$ where $\phi_u(k)$ is the unwrapped instantaneous phase. In this work, unwrapping changes absolute phase jumps greater than π to their 2π complement by adding multiples of $\pm 2\pi$. Standard deviation of

the absolute value of the normalized centered instantaneous frequency is estimated as [7]

$$\hat{\sigma}_f = \sqrt{\frac{1}{M}\left(\sum_k \tilde{f}^2(k)\right) - \left(\frac{1}{M}\sum_k |\tilde{f}(k)|\right)^2} \qquad (8)$$

where $\tilde{f}(k)$ is the normalized centered instantaneous frequency, i.e., $\tilde{f}(k) = (f(k) - \mu_f)/(\max|f(k) - \mu_f|)$ where $f(k)$ is the instantaneous frequency and μ_f is the mean of $f(k)$. The sums are again taken over non-weak samples. The same threshold value was employed as in (7).

As suggested in [1], the instantaneous frequency was median-filtered in order to suppress the spikes due to the phase changes in the phase coded waveforms. In the simulation experiments, a median-filter with window size 5 was used. This feature is considered only for the network 1.

1) Wigner Distribution Based Instantaneous Frequency Features: The P1 and P4 codes are difficult to distinguish from each other due to the fact that both codes have the smallest phase increments in the center of the code and the largest in the ends of the code. In the following three new features based on the instantaneous frequency are proposed for this problem. The features are based on the fundamental difference between the Frank, P1, and P2 and on the other hand the P3 and P4 codes. The first group of polyphase codes (i.e., the Frank, P1, and P2 codes) are derived from a step approximation to LFM whereas the latter group (i.e., the P3 and P4 codes) are derived by sampling directly the phase of LFM. This property is evident from the instantaneous frequency plots (see Fig. 3).

In the following, instantaneous frequency is estimated using a method based on the Wigner distribution with data-driven adaptive window length proposed in [15]. The idea in the algorithm is to calculate the Wigner spectrum using windows of increasing lengths. The discrete pseudo-Wigner distribution (PWD) is defined as [16]

$$W(\omega, k) = 2\sum_{n=-N+1}^{N-1} w(n)y(k+n)y^*(k-n)e^{-j2\omega n} \qquad (9)$$

where N is a positive integer and $w(k)$ is a real-valued symmetric window of length $2N-1$ and $w(0) = 1$. A rectangular window was used in the simulation experiments.

The maximum at each time instant of each calculated PWD is used as an estimate of the instantaneous frequency. The idea is to select independently for each time instant a window length that provides the best bias-variance tradeoff. This is accomplished by starting from the shortest window and increasing the window length step by step until the bias of the instantaneous frequency estimate becomes too large compared to the variance. For details of the algorithm, see [15].

The instantaneous frequency of the polyphase coded signals consists of a sequence of high spikes occurring at the phase changes. In order to remove the spikes while preserving the main trend of the instantaneous frequency (i.e., for example, the main trend of a P1 coded waveform is a step approximation to LFM), two modifications were made to the instantaneous frequency calculation. The first modification was to define the minimum window length used to calculate the PWDs as

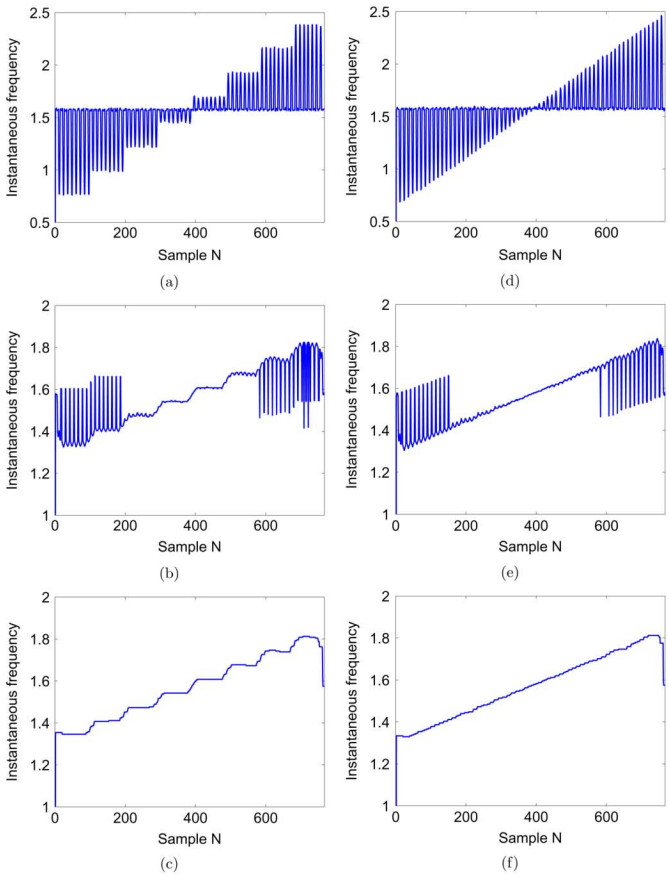


Fig. 3. Instantaneous frequency estimates of P1 (a)–(c) and P4 (d)–(f) coded signals at SNR of 30 dB calculated using a method based on the Wigner distribution with data-driven adaptive window length [15]. (a) and (d) Without minimum window length modification and median-filtering. (b) and (e) With minimum window length modification but without median-filtering. (c) and (f) With minimum window length modification and median-filtering. The high spikes in (b) and (e) are due to the cross-terms of the Wigner distribution.

$$N_{\min} = 3\frac{f_s}{f_e} + \mathrm{rem}\left(3\frac{f_s}{f_e}, 2\right) + 1 \qquad (10)$$

where $\mathrm{rem}(x, 2)$ denotes the remainder of $(x/2)$, and f_s and f_e are the sampling and the subpulse frequencies, respectively. The objective is to guarantee that the minimum window length $N_{\min}$ is always at least three times the number of samples per subpulse. The objective is to ensure that the main trend is extracted. Second and third term in (10) ensure that $N_{\min}$ is odd. In the simulation experiments, window lengths up to 129 were used. The window lengths are given by $N_s = 2(N_{s-1} - 1) + 1$ where $N_0 = N_{\min}$. For example, if $N_{\min} = 9$ the window lengths are $[9, 17, 33, 65, 129]$.

The second modification is to filter the instantaneous frequency estimates using median-filters of increasing length. Median-filters have been proposed in [17] for removing high impulse errors from a time-frequency distribution based instantaneous frequency estimate. In this work, the objective of median-filtering is to remove the spikes that occur as a result of the first modification while simultaneously preserving the steps. These spikes in the instantaneous frequency estimate are due to

the cross-terms of the pseudo-Wigner distribution. The problem with the cross-terms occurs only at the ends of the code for both P1 and P4 codes since the highest auto-term has the lowest magnitude in those regions. This is due to the fact that the largest phase increments between consecutive subpulses are at the ends of the code.

The instantaneous frequency estimate for each window length is filtered using all of the following length median-filters starting from the smallest length: $3, 5, \ldots, (N_{\min} - 1)/2 + 1$. Filtering with several different length median-filters is required due to the fact that the spikes may comprise more than one sample. Fig. 3 illustrates the effect of the two modifications. It can be seen that both modifications are required.

A viable alternative to the above median-filtering approach could be the algorithm proposed in [18]. The algorithm combines the method based on Wigner distribution maxima with the minimization of the instantaneous frequency variations between consecutive points.

After the instantaneous frequency has been estimated the features can be calculated. Feature calculation will be explained in the following. As a first step in feature calculation, the linear trend is removed from the instantaneous frequency. That is, a line is fitted to the estimated instantaneous frequency by minimizing the sum of square errors using a least square (LS) method. But first, due to the fact that the instantaneous frequency estimate for the P1 and P4 coded signals is more unreliable toward the beginning and end of the instantaneous frequency, $1/6$th of the instantaneous frequency is removed from both ends (i.e., in total $1/3$rd). After this, the values of the resulting instantaneous frequency vector $\boldsymbol{f}'$ are normalized between 0 and 1, i.e., $\boldsymbol{f} = (\boldsymbol{f}' - \min \boldsymbol{f}')/(\max\{\boldsymbol{f}' - \min \boldsymbol{f}'\})$. Finally, the line is fitted to the normalized instantaneous frequency by minimizing the sum of square errors. The estimated linear trend is subtracted from the normalized instantaneous frequency estimate. The resulting vector is denoted by $\boldsymbol{f}_n$.

Three features are calculated from $\boldsymbol{f}_n = [f_n(1), \ldots, f_n(N)]^T$. The normalized instantaneous frequency $\boldsymbol{f}_n$ of the P1 codes shows a clear structure due to the step approximation while for the P4 codes it resembles white noise.

The first feature is the standard deviation of $\boldsymbol{f}_n$

$$\sigma_{Wf} = \sqrt{\frac{1}{N-1}\sum_{k=1}^{N} f_n^2(k) - \left(\frac{1}{N-1}\sum_{k=1}^{N} f_n(k)\right)^2}. \tag{11}$$

The second feature is based on the autocorrelation of $\boldsymbol{f}_n$. It is estimated as $c(m) = \sum_k f_n(k) f_n(k-m)$, $m = 0, 1, \ldots, N-1$.

The feature calculates the ratio between the sidelobe maximum and the maximum

$$r = \frac{\frac{1}{N-m_1}\max_{m\in[m_0,N-1]} c(m)}{\frac{1}{N}\max c(m)} \tag{12}$$

where m_0 is the lag value corresponding to the minimum of $c(m)$, and m_1 is the lag value corresponding to the maximum of $c(m)$ for lag values in the interval $[m_0, N-1]$ (i.e., m_1 is the lag value giving the maximum in the numerator). Fig. 4 depicts examples of autocorrelations of $\boldsymbol{f}_n$ for P1 and P4 coded signals at SNR of 5 dB.

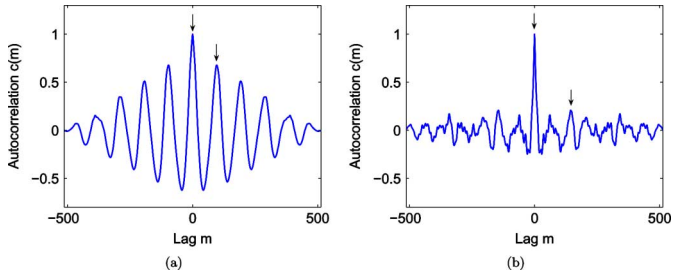


Fig. 4. Autocorrelations of $\boldsymbol{f}_n$ for (a) P1 and (b) P4 coded signals at SNR of 5 dB. The arrows indicate the peaks whose ratio is used as a feature.

Third feature is based on the statistical runs test. The idea is to measure whether $\boldsymbol{f}_n$ resembles white noise and thus has a high number of runs (P4) or has a stepping structure and thus a low number of runs (P1). The feature is calculated from $f_n(k), k = 1, \ldots, N$, as follows. First the mean of $f_n(k)$ is calculated. Then a sequence of 0's and 1's, denoted by $b(k), k = 1, \ldots, N$, is formed

$$b(k) = \begin{cases} 1, & \text{if } f_n(k) > \bar{f}_n, \\ 0, & \text{otherwise} \end{cases} \tag{13}$$

where $\bar{f}_n$ is the mean of $f_n(k)$. A consecutive sequence of 0's or 1's is called a run. Let R denote the number of runs. Furthermore, let N_A and N_B denote the number of observations above and below the mean, respectively. The mean and variance of R can be derived as

$$E[R] = \frac{N + 2N_A N_B}{N}, \qquad \mathrm{Var}(R) = \frac{2N_A N_B(2N_A N_B - N)}{N^2(N-1)} \tag{14}$$

where N is the total number of observations, i.e., $N = N_A + N_B$. When N is relatively large (> 20) the distribution of R is approximately normal, i.e., $Z = (R - E[R])/(\sqrt{\mathrm{Var}(R)}) \sim N(0,1)$. However, due to the limited resolution of the Wigner distribution as well as the median-filtering of the instantaneous frequency, $\boldsymbol{f}_n$ for the P4 codes does not exactly resemble noise. Hence, the values of Z are quite large for the P4 codes as well. This causes computational difficulties (probability values are too small for machine precision). Therefore, the value of R was normalized using the following equation $Y = (R - E[R])/(\mathrm{Var}(R))$.

The final feature is

$$p_R = 2(1 - \Phi(|Y|)) \tag{15}$$

where $\Phi(\cdot)$ denotes the standard Normal cumulative distribution function. This normalizes the value of p_R between 0 and 1 (note that p_R is no longer a probability due to the employed normalization).

D. Choi–Williams Time-Frequency Distribution Features

In this section, features based on Choi–Williams time-frequency distribution are proposed. Choi–Williams distribution

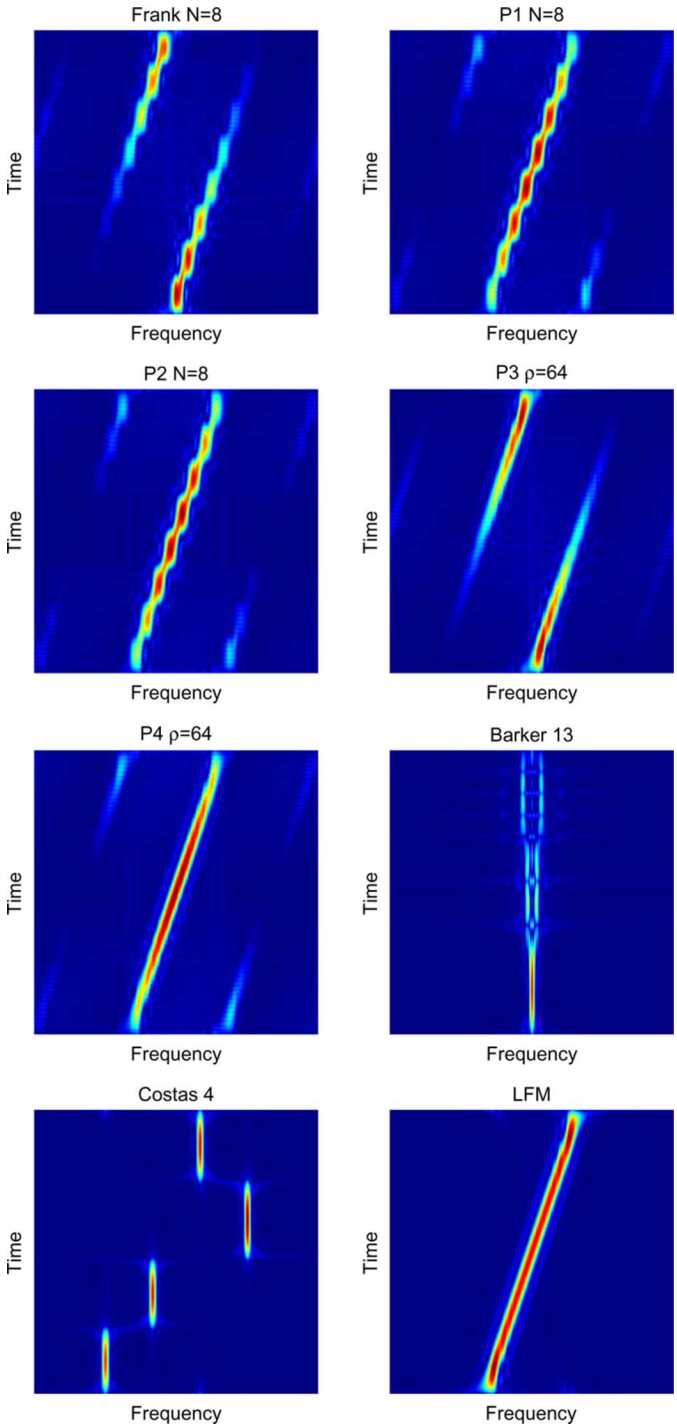


Fig. 5. Example CWDs for the different waveform classes (Frank, P1, P2, P3, P4, Binary phase, Costas, LFM). The scaling factor σ had a value 0.05. Differences among the codes are clearly visible.

(CWD) was selected since it uses an exponential kernel whose behavior can be easily adjusted by changing the value of the scaling parameter. CWD of a continuous time signal $y(t)$ is defined by [19]

$$W_{\mathrm{CW}}(t,\omega) = \iint \frac{1}{\sqrt{4\pi\tau^2/\sigma}} \exp\left(-\frac{(\mu-t)^2}{4\tau^2/\sigma}\right) \cdot y\left(\mu+\frac{\tau}{2}\right) y^*\left(\mu-\frac{\tau}{2}\right) \exp(-j\omega\tau)\, d\mu\, d\tau \quad (16)$$

Fig. 6. The CWD normalization procedure starting from the original image (from left to right): 1) thresholding, 2) time gating and frequency filtering, and 3) aspect ratio normalization.

where σ ($\sigma > 0$) is a scaling factor. The CWD uses an exponential kernel function $\Phi(\theta,\tau) = \exp(-\theta^2\tau^2/\sigma)$ where θ denotes the frequency. The kernel function acts as a low-pass filter where σ controls the attenuation. Note, however, that the kernel function is identically one at the θ- and τ-axes and thus it preserves all the horizontal and vertical cross-terms. This is a drawback of the CWD that is taken into account when the CWD is normalized for feature calculation.

Choosing a small σ will provide good cross-term suppression. However, at the same time it will also cause considerable smearing and loss of resolution of the autoterms. Fortunately, this can actually be a desired effect since image processing methods are used to extract the features. Therefore, it is preferred that the image objects have slightly larger size. Hence, in the experiments a small value 0.05 is used for σ. Fig. 5 shows example images of the CWDs of signals from different classes.

In the following, the CWD is treated as a 2-D image. The CWD image is normalized in order to minimize the effect of the signal's bandwidth and the sampling frequency.

1) Normalization of the CWD Image: The normalization procedure comprises three stages.

1) Thresholding of the CWD image into a binary image.
2) Time gating and frequency filtering of the binary image, i.e., removal of areas not containing signal components from the edges of the binary image.
3) Aspect ratio normalization of the binary image to one.

Fig. 6 depicts the three normalization stages for a Frank coded waveform.

The goal in the thresholding stage is to create a binary CWD image that contains only significant signal components. That is, the thresholded binary CWD image should not contain any isolated noise points since the outcome of the second stage can easily be affected by such noise. In order to accomplish this the image is first thresholded into a binary image using a global threshold T. The employed algorithm is the basic global thresholding algorithm ([20, p. 598]).

1) Select an initial estimate for the global threshold T.
2) Segment the image using T. This produces two pixel groups G_1 and G_2. G_1 consists of all pixels with gray level values $> T$ and G_2 consists of all pixels with gray level values $\leq T$.
3) Compute the average gray levels μ_1 and μ_2 for the pixels in groups G_1 and G_2.
4) Compute the new threshold value $T = (1/2)(\mu_1 + \mu_2)$.
5) Repeat steps 2 through 4 until the difference in T between successive iterations is smaller than a predefined convergence parameter.

The initial estimate for T can be selected to be the average of the maximum and minimum gray levels in the CWD image.

The above thresholding algorithm cannot guarantee complete removal of isolated noise. In addition, the form of the Choi–Williams kernel can cause, especially at low SNR regime, undesired horizontal and vertical lines to the globally thresholded image. With morphological opening followed by a removal of objects that are not large enough (e.g., at minimum 10% of the size of the largest object) the problems can be alleviated. Morphological opening (i.e., erosion followed by dilation) smoothens the edges of the image objects while removing small noise samples as well as the undesired horizontal and vertical lines that usually are very thin. In the simulation experiments the morphological operations were conducted using a 3×3 square mask. Morphological opening will remove only small or thin objects. Labeling followed by a removal of small objects will remove slightly larger objects as well. These objects might be caused by the aforementioned reasons as well or they might be due to the sideterms observed in the CWDs of the P1, P2, and P4 codes (see Fig. 5). The occurrence of these objects will degrade the discrimination capability of the extracted features. Hence, they should be removed.

An interesting alternative solution for the first stage of the normalization procedure would be the method of adaptive thresholding of a time-frequency distribution proposed in [21]. Nevertheless, the algorithm proposed above is very well suited for the problem for its simplicity of operation (basically no parameters have to be set) and good performance at different SNR regimes.

In the second stage of the normalization procedure, the binary CWD image is cropped by removing the areas not containing signal components from the edges of the image. In the third and final stage of the normalization procedure, the aspect ratio of the image is normalized to one by resizing the image to a square image. The size of the resized image is $M \times M$ where M is the minimum dimension of the image before resizing. The resizing can be conducted using the nearest neighbor interpolation method.

2) CWD Features: The features introduced in this section include a number of pseudo-Zernike moments as well as three other features specifically suitable for pulse compression radar waveform recognition. The pseudo-Zernike moments have been used, e.g., for handwritten character recognition in [22]. The fact that the chosen pseudo-Zernike moments are invariant to translation, scaling, rotation, and mirroring makes them very suitable to the problem at hand. The invariance properties reduce the requirement for training signals, i.e., a different training signal is not required for each special case.

The pseudo-Zernike moments characterize the normalized binary CWD image as a whole. Calculating the pseudo-Zernike moments using scaled geometric moments renders them invariant to translation and scaling. The geometric moments of order $p + q$ of a digital image $f(x, y)$ are given by $m_{pq} = \sum_x \sum_y f(x, y) x^p y^q$.

The translation and scale invariant central geometric moments are defined as [22]

$$G_{pq} = \frac{1}{m_{00}^{(p+q+2)/2}} \sum_x \sum_y f(x, y)(x - \bar{x})^p (y - \bar{y})^q \quad (17)$$

where $\bar{x} = (m_{10})/(m_{00})$ and $\bar{y} = (m_{01})/(m_{00})$.

The translation and scale invariant radial-geometric moments are defined as [22]

$$R_{pq} = \frac{1}{m_{00}^{(p+q+3)/2}} \sum_x \sum_y f(x, y)(\tilde{x}^2 + \tilde{y}^2)^{1/2} \tilde{x}^p \tilde{y}^q \quad (18)$$

where $\tilde{x} = x - \bar{x}$ and $\tilde{y} = y - \bar{y}$.

The pseudo-Zernike moments Z_{nm} of order n with repetition m ($|m| \leq n$) are given by [23], [22]

$$\begin{aligned} Z_{nm} = {} & \frac{n+1}{\pi} \sum_{\substack{s=0 \\ n-s-m=even}}^{n-|m|} D_{nms} \sum_{a=0}^{k} \sum_{b=0}^{m} \\ & \times (-j)^b \binom{k}{a} \binom{m}{b} B_{nms} G_{2k-2a+m-b,2a+b} \\ & + \frac{n+1}{\pi} \sum_{\substack{s=0 \\ n-s-m=odd}}^{n-|m|} D_{nms} \sum_{a=0}^{d} \sum_{b=0}^{m} \\ & \times (-j)^b \binom{d}{a} \binom{m}{b} B_{nms} R_{2d-2a+m-b,2a+b} \end{aligned} \quad (19)$$

where $k = \frac{n-s-m}{2}$, $d = \frac{n-s-m-1}{2}$, and

$$B_{nms} = (-1)^s \frac{(n-s)!}{s! \left(\frac{n+|m|}{2} - s\right)! \left(\frac{n-|m|}{2} - s\right)!}, \quad (20)$$

$$D_{nms} = (-1)^s \frac{(2n+1-s)!}{s!\,(n-|m|-s)!\,(n+|m|+1-s)!}. \quad (21)$$

Rotation invariance is achieved by taking the absolute value of the pseudo-Zernike moments Z_{nm} [23]. The dynamic range can be reduced by taking the logarithm. These two procedures give the final features $\hat{Z}_{nm} = \ln |Z_{nm}|$.

The following pseudo-Zernike moments of the binary CWD image were experimentally selected as features: $\hat{Z}_{20}$, $\hat{Z}_{22}$, $\hat{Z}_{30}$, $\hat{Z}_{31}$, $\hat{Z}_{32}$, $\hat{Z}_{33}$, and $\hat{Z}_{43}$. The pseudo-Zernike moments are used only in network 2 due to the fact that the CWDs of the Costas and binary phase codes vary too much depending on the specific code to enable reliable generalization from the training data.

Number of objects in the normalized binary CWD image (denoted by N_{obj}) is a discriminating feature among the radar waveforms. Provided that the normalization is done successfully, the Frank and P3 codes have two image objects, the other polyphase codes and LFM codes have one object and the Costas codes have as many objects as there are different frequencies. For the binary phase codes this feature is not reliable. Although the normalization procedure should already remove small insignificant objects, to further increase the robustness of the feature, objects smaller than 20% of the size of the largest object are discarded.

Location of the peak energy in time coordinate of the CWD image is another discriminating feature. The feature is defined as

$$t_{\max} = \frac{1}{N-1} \arg\max_x \{W_{\mathrm{CW}}(x, y)\} \quad (22)$$

where $W_{\mathrm{CW}}(x, y)$ is the time-gated CWD image, x and y are the time and frequency axes, respectively, and N is the length of

the time axis. The feature cannot be calculated from the binary CWD image. Hence, it does not require the complete normalization procedure of the CWD image. Only time gating is required.

The final CWD feature measures the standard deviation of the width of the signal objects in the binary CWD image. The feature is suitable particularly for polyphase code discrimination. The feature is based on the same property as the Wigner distribution instantaneous frequency features introduced earlier in the paper. That is, the Frank, P1, and P2 codes are derived from a step approximation to LFM while P3 and P4 codes are derived by directly approximating LFM (see Fig. 5). The feature calculation proceeds as follows.

- For each object $k = 1, \ldots, N_{\text{obj}}$.
  1) Select object k and mask the other objects away to obtain binary image $B(x, y)$.
  2) Calculate the principal components of $B(x, y)$.
  3) Rotate $B(x, y)$ so that the first principal axis corresponding to the largest eigenvalue is parallel to the vertical axis. Nearest neighbor interpolation can be used in the rotation.
  4) Calculate the row sum (i.e., the number of white pixels in each row) $r(x) = \sum_{y=0}^{N-1} B_r(x, y)$, $x = 0, 1, \ldots, N-1$, where $B_r(x, y)$ is the rotated image.
  5) Normalize $r(x)$ between 0 and 1, i.e., $\hat{r}(x) = \frac{r(x)}{\max r(x)}$.
  6) Calculate the standard deviation of $\hat{r}(x)$, i.e., $\sigma_{k,\text{obj}} = \sqrt{(1/M)\sum_x \hat{r}^2(x) - ((1/M)\sum_x \hat{r}(x))^2}$ where the sums are taken over non-weak samples, i.e., $\hat{r}(x) \geq T_{\text{obj}}$, and M is the number of non-weak samples.
- The final feature value is the average of the standard deviations, i.e., $\hat{\sigma}_{\text{obj}} = (1)/(N_{\text{obj}}) \sum_{k=1}^{N_{\text{obj}}} \sigma_{k,\text{obj}}$

The threshold value T_{obj} can be chosen, for example, as 0.3 (i.e., 30% of the maximum of $\hat{r}(x)$). This is to guarantee that especially the rows that do not contain any signal do not affect the feature value. Such rows will occur due to the rotation as well as when there are more than one object. The feature is beneficial in network 1 as well.

VI. Simulation Results and Discussion

In this section the performance of the proposed classification system is analyzed using simulated data. Classification performance is measured as a function of the SNR in AWGN channel. The SNR is defined as $\text{SNR}_{\text{dB}} = 10 \log_{10} (\sigma_x^2)/(\sigma_n^2)$ where σ_x^2 and σ_n^2 are the variances of the original signal and the noise, respectively.

In each of the simulations two classifiers are trained. The first classifier comprises two ensemble averaging ESCs, i.e., one for each network (see Fig. 2). Both of the ESCs consist of 10 MLPs. Each MLP has one hidden layer with 30 neurons. In early-stop training 90% of the training data patterns were used for training and 10% were reserved for validation (i.e., for determining the early-stopping point). The partitioning to training and validation sets was done randomly for each MLP. The MLP training was conducted using the scaled conjugate gradient algorithm [24].

The second classifier comprises two Bayesian MLPs. Both MLPs have one hidden layer with 30 neurons. One common precision hyperparameter was used for the input-to-hidden layer weights. The prior distribution for the weights was the Gaussian distribution and for the precision hyperparameters the Gamma distribution. The parameters of the Gamma distributions were chosen to correspond to non-informative hyperpriors, i.e., the shape parameter had always value $\nu_0 = 0.5$ while the choice of the mean parameter μ depended on the distribution. The values $\mu_{w_1} = 400 N_i^{2/\nu_0}$, $\mu_{w_2} = 400 N_h^{2/\nu_0}$, and $\mu_{b_1} = \mu_{b_2} = 400$ were chosen for the input and hidden layer weights and biases, respectively. N_i and N_h are the number of input and hidden neurons. The Bayesian learning was performed using the Flexible Bayesian Modeling (FBM) software.[1] The number of samples in Markov chains was 20000. The trajectory length was 100 and a window size of 10 was used. Step size adjustment factor was 0.05 and the momentum parameter was 0.9. For explanation of the parameters, see [11] and the FBM software documentation. Network predictions were made using every 20th sample from the last 2000 samples, i.e., 100 equidistant samples.

TABLE II
Simulation Parameters. $U(\cdot, \cdot)$ Denotes a Uniform Distribution

Parameter	Data spec 1	Data spec 2
General parameters		
Sampling frequency	1	1
Polyphase codes		
Carrier frequency f_c	U($\frac{1}{3}, \frac{1}{4}$)	U($\frac{1}{3}, \frac{1}{4}$)
Subpulse frequency	$\frac{1}{4} f_c$	$\frac{1}{2} f_c$
Frank, P1; N [25, pp. 10–11]	3–7	8–10
P2; N [25, p. 12]	4, 6	8, 10
P3, P4; ρ [25, pp. 11–12]	N^2, $N \in [3, 7]$	N^2, $N \in [8, 10]$
Binary phase codes		
Carrier frequency f_c	U($\frac{1}{3}, \frac{1}{4}$)	U($\frac{1}{3}, \frac{1}{4}$)
Subpulse frequency	$\frac{1}{4} f_c$	$\frac{1}{2} f_c$
Barker, Quadratic residue sequences, Maximal length sequences, Gold codes (pulse lengths)	84–496 samples	180–2032 samples
Costas codes		
Number of subpulses N	2, 4, 6	4, 6
Lowest frequency f_0	$\frac{1}{24}$	$\frac{1}{24}$
Frequency hop Δf	U($\frac{1}{24}, \frac{1}{12}$)	U($\frac{1}{24}, \frac{1}{12}$)
Subpulse length τ_1	U(60,120) samples	U(60,120) samples
Linear frequency modulation		
Carrier frequency f_c	$\frac{1}{4}$	$\frac{1}{4}$
Bandwidth Δf	U($\frac{1}{24}, \frac{1}{8}$)	U($\frac{1}{8}, \frac{1}{4}$)
Modulation period τ (pulse length)	U(240,480) samples	U(240,480) samples

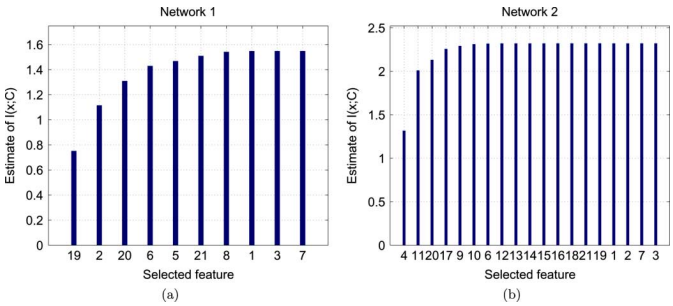
The input features x_i^n are normalized using simple linear rescaling, i.e., $\tilde{x}_i^n = (x_i^n - \mu_i)/(\sigma_i)$ where μ_i and σ_i are the mean and standard deviation of the ith feature estimated from the training data. Superscript n denotes the nth data pattern.

Table II lists the simulation parameters. Two different data specifications are defined. The purpose is to measure the generalization capability of the classification system. That is, to measure how the classification system performs on data different from training data.

A. Feature Selection Results

Feature selection was based on a data set consisting of 2000 pulses per class generated using both data specifications. That is, 1000 pulses from both data specifications with random parameter values. In order to obtain robustness against SNR variations, the SNR of each pulse was randomly selected between 0 and 20 dB.

[1] http://www.cs.toronto.edu/~radford/fbm.software.html



(a) (b)

Fig. 7. Estimated mutual information during the progress of the feature selection algorithm (a) network 1 (10 features out of 11 were selected, $H(C) = 1.5488$) and (b) network 2 (9 features out of 44 were selected, $H(C) = 2.3219$).

At this stage a total of 11 and 44 features for the networks 1 and 2 were considered, respectively. The mutual information between the classes and each individual feature was estimated. In order to expedite the final feature vector selection, the features whose estimated mutual information with the classes was close to zero (less than 0.05) were discarded at this stage. The discarded features include, for example, many higher order moments and cumulants as well as few other features.

After the initial pruning, there were 10 and 20 features remaining for the networks 1 and 2, respectively. Fig. 7 plots the estimated mutual information at each step of the feature selection algorithm for the networks 1 and 2, respectively. For the sake of completeness, the figures show the estimated mutual information for all possible feature vector dimensions. However, the feature selection was stopped when the difference between the estimated mutual information of successive iterations was less than 10^{-4}. This is a conservative value. It was chosen in order not to sacrifice the classification performance since the Parzen window estimate of the mutual information in (2) is not very accurate especially in high dimensional cases [26]. Consequently, 10 out of 11 features (i.e., all of the features remaining after the initial pruning) were selected for the network 1 and 9 out of 44 for the network 2. See Fig. 7 along with Table I for the list of selected features.

B. Classification Performance

In this section, the classification performance is measured on data similar to the training data. The goal is to get an idea of the performance of the classification system when the training data is comprehensive for the signal environment. That is, the signal lengths and parameter ranges are the same although the parameter values are selected randomly. For this purpose training and testing data sets were generated. Testing data set consisted of 500 pulses per class from both data specifications (i.e., 1000 in total) for SNRs of $-3, 0, \ldots, 30$ dB. Training data set consisted of 300 pulses per class from both data specifications (i.e., 600 in total). The goal is to obtain good classification performance for large range of SNRs. Thus, the pulse SNRs were selected randomly between 0 and 20 dB.

Fig. 8 plots the classification probabilities as a function of the SNR. The probabilities are measured on the testing data. Very reliable performance is obtained with both classifiers. The overall correct classification rate is over 98% at SNR of 6 dB for both classifiers. The Bayesian MLP has sligthly better performance than the ESC at low SNR regime.

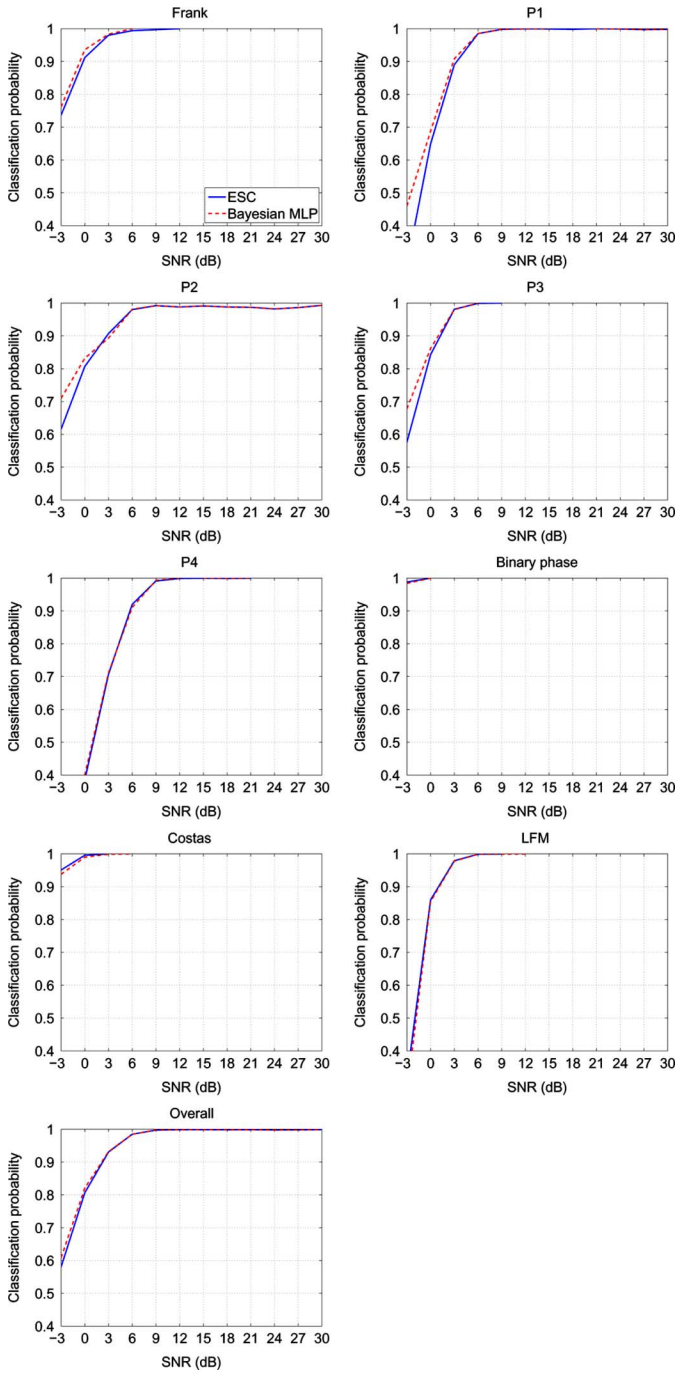


Fig. 8. Classification performance as a function of the SNR on data similar to the training data.

Table III shows the confusion table for the Bayesian MLP at SNR of 6 dB. As the SNR increases the confusion decreases except between the P1 and P2 codes. Both classifiers classify roughly 1–2% of the P2 codes incorrectly to the P1 codes even at high SNRs. The P1 and P2 codes are very similar to each other. For even length codes both codes have the same phase increments within each frequency, only the starting phases are different. Hence, discriminating them is very difficult.

C. Generalization Performance

In order to measure the generalization performance of the classification system in a more difficult scenario, the classifiers

TABLE III
CONFUSION MATRIX FOR THE BAYESIAN MLP AT SNR OF 6 dB. THE OVERALL CORRECT CLASSIFICATION RATE WAS 98.5%

	Frank	P1	P2	P3	P4	Binary	Costas	LFM
Frank	100	0	0	0	0	0	0	0
P1	0	98.5	0	0	1.5	0	0	0
P2	0	1.2	98.1	0	0	0	0	0.7
P3	0	0	0	100	0	0	0	0
P4	0	1.3	0	0	91.1	0	0	7.6
Binary	0	0	0	0	0	100	0	0
Costas	0	0	0	0	0	0	100	0
LFM	0	0	0.1	0	0	0	0	99.9

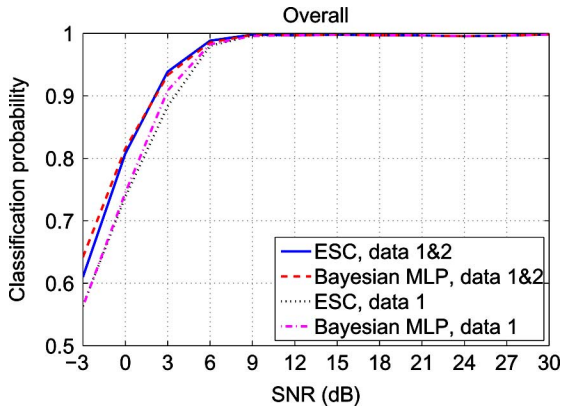


Fig. 9. Overall correct classification rate on testing data set generated using data specification 2. Legends indicate the classifiers and the data specifications used to generate the training data.

were trained using a training data set generated using one data specification while the testing data set was generated using the other data specification. Hence, in addition to random parameter values (i.e., carrier frequencies, bandwidths, and frequency hops) used in the previous simulation, the signal lengths and/or parameter ranges are different as well (see Table II).

Fig. 9 plots the overall correct classification rate of the classifiers trained using the data specification 1 on a testing data set generated using the data specification 2 (the opposite case is not tested due to the expected poor performance since in practice the shorter codes define the discrimination boundaries). For comparison, the classification performances of the classifiers from the previous experiment are plotted as well. That is, classifiers whose training data set was generated using both data specifications.

The generalization performance from the data specification 1 to the data specification 2 is very good. Only a slight loss in performance is observed at low SNR regime. For SNRs of 6 dB and higher the performances are practically identical. Classifier comparison shows that the Bayesian MLP has better generalization performance than the ESC at low SNR regime.

D. Classification Performance With Carrier Frequency Estimation

In this simulation experiment instead of assuming the carrier frequency known it was estimated. The Lank, Reed, Pollon frequency estimator was employed. That is, the estimate of the

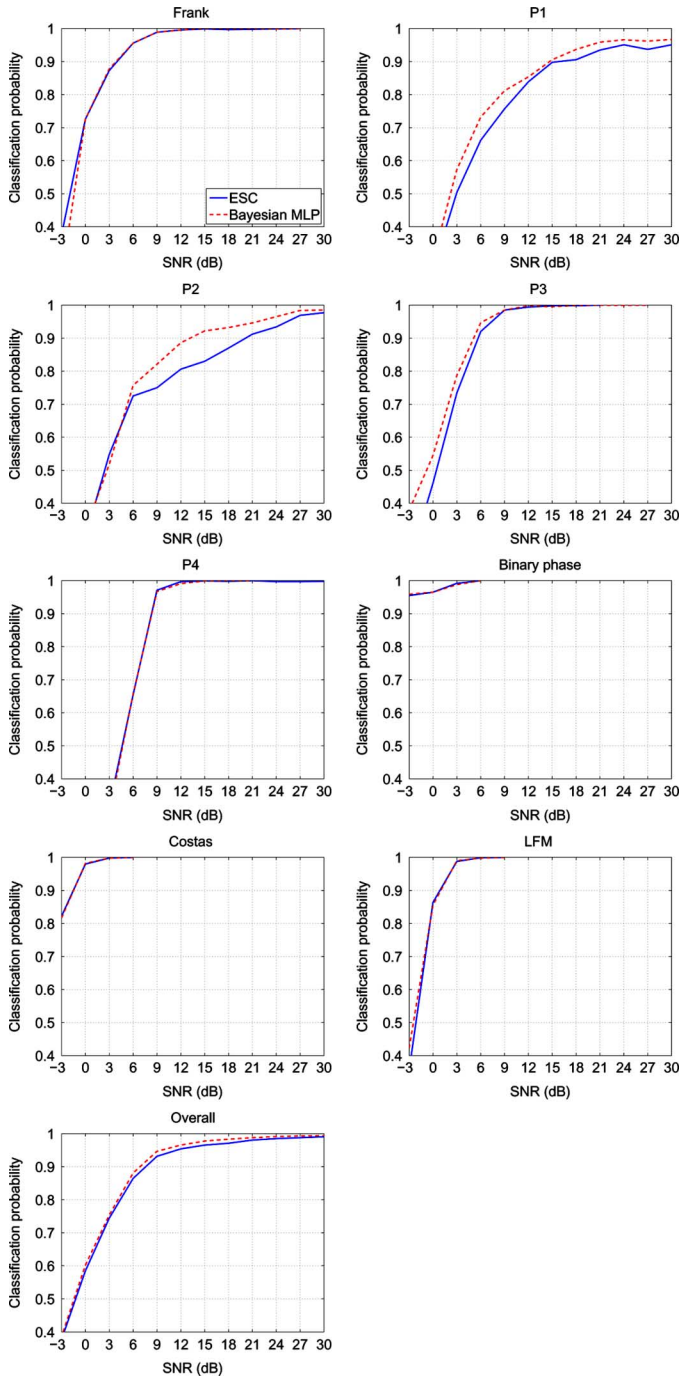


Fig. 10. Classification performance as a function of the SNR on data similar to the training data with carrier frequency estimation and removal.

carrier frequency of the pulse $y(k)$ is defined as [27]

$$\hat{\omega} = \angle \sum_{k=1}^{N-1} y(k) y^*(k-1) \tag{23}$$

where $\angle$ denotes the phase angle. The estimator is computationally very efficient. It gives an estimate of the mean frequency. Thus, it can also be used for signals that do not have a unique maximum in their frequency spectrum, such as the Costas codes.

Fig. 10 depicts the classification performance with carrier frequency estimation. The training data consisted of 600 pulses per

class generated using both data specifications. Comparison with the results when carrier frequency was assumed to be known (see Fig. 8) shows a clear performance loss for the P1 and P2 codes. The P1 and P2 codes get confused with each other. The primary feature discriminating between the P1 and P2 codes is the cross-correlation time lag feature. This feature is affected by carrier frequency offset, i.e., a frequency offset can shift the location of the cross-correlation peak, and thus alter the feature value. For the other codes, a loss in performance is observed at low SNR regime (i.e., 6 dB and below depending on class). At high SNR regime, the performance is not affected by carrier frequency estimation (except for the P1 and P2 codes).

One source of error might also be the fact that the Lank, Reed, Pollon estimator provides an estimate of the mean frequency. For the waveforms with non-symmetric PSDs (for example, the polyphase waveforms), the mean frequency is not the same as the center frequency of the signal's frequency band. Especially the fact that the performance for the P1 and P2 codes is affected even at very high SNR regime supports this theory. Note, however, that most of the features are not affected by nor do they require carrier frequency estimation. Such features are the CWD features, the PSD features as well as the features based on instantaneous frequency.

By comparing the performances of the classifiers, it can be seen that the Bayesian MLP clearly outperforms the ESC in this scenario.

VII. Conclusion

In this paper, a classification system for classifying common pulse compression radar waveforms has been developed. Eight different waveform classes were considered. No such an extensive classification of waveforms has been presented in literature previously. The classification system is based on features extracted from an intercepted radar pulse. The extracted feature vectors are fed into a supervised classifier that performs the final classification.

New features for radar waveform recognition have been proposed. In particular, the difficult task of discriminating among polyphase codes has been addressed in detail. New features based on Wigner and Choi–Williams time-frequency distributions have been proposed. The final feature vectors have been selected using an information theoretic feature selection algorithm.

A parallel classifier structure based on MLP networks has been proposed. Two classifier structures based on the ESC and the Bayesian MLP, respectively, have been trained and tested in the simulations. Simulation results showed that the classifiers perform very reliably achieving overall correct classification rate of 98% at SNR of 6 dB. Simulation experiments also showed that the proposed classification system can in certain conditions generalize outside the training data.

In order to measure the performance degradation due to carrier frequency estimation errors, in one simulation experiment the carrier frequency was estimated instead of assuming it to be known. The results showed that the performance can drop if the carrier frequency estimate is not accurate enough. More work is required to find an appropriate carrier frequency estimator.

Comparison of the work to the techniques such as AD based [1] or STFT based [3] radar waveform recognition is difficult due to the fact that in our system, the number of different classes is significantly higher and includes polyphase codes as well. Significant part of our work was aimed at discriminating between the polyphase codes. This problem was not considered in [1] or [3].

Acknowledgment

The authors would like to thank L. Terho and Prof. V. Halonen for useful discussions.

References

[1] G. López-Risueño, J. Grajal, and O. Yeste-Ojeda, "Atomic decomposition-based radar complex signal interception," *Proc. Inst. Elect. Eng., Radar, Sonar, Navig.*, vol. 150, no. 4, pp. 323–331, Aug. 2003.

[2] G. López-Risueño and J. Grajal, "Multiple signal detection and estimation using atomic decomposition and EM," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 42, no. 1, pp. 84–102, Jan. 2006.

[3] G. López-Risueño, J. Grajal, and Á. Sanz-Osorio, "Digital channelized receiver based on time-frequency analysis for signal interception," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 41, no. 3, pp. 879–898, Jul. 2005.

[4] P. E. Pace, *Detecting and Classifying Low Probability of Intercept Radar*. Norwood, MA: Artech House, 2004.

[5] C. Louis and P. Sehier, "Automatic modulation recognition with a hierarchical neural network," in *Proc. Military Communications Conf.*, Oct. 1994, vol. 3, pp. 713–717.

[6] J. Venäläinen, L. Terho, and V. Koivunen, "Modulation classification in fading multipath channel," in *Proc. 36th Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, CA, Nov. 3–6, 2002, vol. 2, pp. 1890–1894.

[7] M. L. D. Wong and A. K. Nandi, "Automatic digital modulation recognition using artificial neural network and genetic algorithm," *Signal Process.*, vol. 84, no. 2, pp. 351–365, Feb. 2004.

[8] P. Ciblat, P. Loubaton, E. Serpedin, and G. B. Giannakis, "Asymptotic analysis of blind cyclic correlation-based symbol-rate estimators," *IEEE Trans. Inform. Theory*, vol. 48, no. 7, pp. 1922–1934, Jul. 2002.

[9] C. M. Bishop, *Neural Networks for Pattern Recognition*. New York: Oxford University Press, 1995.

[10] J. B. Hampshire II and B. A. Pearlmutter, "Equivalence proofs for multi-layer perceptron classifiers and the Bayesian discriminant function," in *Proc. 1990 Connectionist Models Summer School*, D. Touretzky, J. Elman, T. Sejnowski, and G. Hinton, Eds. San Mateo, CA: Morgan Kaufmann, 1990, pp. 159–172.

[11] R. M. Neal, *Bayesian Learning for Neural Networks, Lecture Notes in Statistics*. New York: Springer-Verlag, 1996, vol. 118.

[12] N. Kwak and C.-H. Choi, "Input feature selection by mutual information based on Parzen windows," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 24, no. 12, pp. 1667–1671, Dec. 2002.

[13] A. K. Nandi and E. E. Azzouz, "Algorithms for automatic modulation recognition of communication signals," *IEEE Trans. Commun.*, vol. 46, no. 4, pp. 431–436, Apr. 1998.

[14] A. K. Nandi and E. E. Azzouz, "Automatic analogue modulation recognition," *Signal Process.*, vol. 46, no. 2, pp. 211–222, Oct. 1995.

[15] V. Katkovnik and L. Stanković, "Instantaneous frequency estimation using the Wigner distribution with varying and data-driven window length," *IEEE Trans. Signal Processing*, vol. 46, no. 9, Sep. 1998.

[16] M. Sun, C.-C. Li, L. N. Sekhar, and R. J. Sclabassi, "Efficient computation of the discrete pseudo-Wigner distribution," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, no. 11, pp. 1735–1742, Nov. 1989.

[17] I. Djurović and L. Stanković, "Robust Wigner distribution with application to the instantaneous frequency estimation," *IEEE Trans. Signal Processing*, vol. 49, no. 12, pp. 2985–2993, Dec. 2001.

[18] I. Djurović and L. Stanković, "An algorithm for the Wigner distribution based instantaneous frequency estimation in a high noise environment," *Signal Process.*, vol. 84, no. 3, pp. 631–643, Mar. 2004.

[19] H.-I. Choi and W. J. Williams, "Improved time-frequency representation of multicomponent signals using exponential kernels," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, no. 6, pp. 862–871, Jun. 1989.

[20] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, Int. ed. Englewood Cliffs, NJ: Prentice-Hall, 2002.

[21] I. Djurović, M. Urlaub, L. Stanković, and J. F. Böhme, "Estimation of multicomponent signals by using time-frequency representations with application to knock signal analysis," in *Proc. Eur. Signal Processing Conf. EUSIPCO 2004*, Vienna, Austria, Sep. 6–10, 2004, pp. 1785–1788.

[22] M. Dehghan and K. Faez, "Farsi handwritten character recognition with moment invariants," in *Proc. 13th Int. Conf. Digital Signal Processing*, Jul. 1997, vol. 2, pp. 507–510.

[23] R. B. Bailey and M. Srinath, "Orthogonal moment features for use with parametric and non-parametric classifiers," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 18, no. 4, pp. 389–399, Apr. 1996.

[24] M. F. Møller, "A scaled conjugate gradient algorithm for fast supervised learning," *Neural Networks*, vol. 6, no. 4, pp. 525–533, 1993.

[25] B. L. Lewis, F. F. Kretschmer, Jr., and W. W. Shelton, *Aspects of Radar Signal Processing*. Norwood, MA: Artech House, 1986.

[26] M. M. Van Hulle, "Edgeworth approximation of multivariate differential entropy," *Neural Comput.*, vol. 17, no. 9, pp. 1903–1910, Sep. 2005.

[27] G. W. Lank, I. S. Reed, and G. E. Pollon, "A semicoherent detection and Doppler estimation statistic," *IEEE Trans. Aerosp. Electron. Syst.*, vol. AES-9, pp. 151–165, Mar. 1973.

Jarmo Lundén (S'04) received the M.Sc. (Tech.) degree with distinction in communications engineering from the Department of Electrical and Communications Engineering, Helsinki University of Technology (TKK), Helsinki, Finland, in 2005. He is currently pursuing the Ph.D. degree at TKK.

His research interests include signal processing for radar interception and identification, and spectrum sensing for cognitive radio.

Visa Koivunen (SM'98) received the D.Sc. (Tech.) degree with honors from the Department of Electrical Engineering, University of Oulu, Oulu, Finland.

From 1992 to 1995, he was a Visiting Researcher at the University of Pennsylvania, Philadelphia. In 1996–1999, he held faculty positions at the University of Oulu and Tampere University of Technology, Finland. Since 1999, he has been a Professor of signal processing at Helsinki University of Technology (TKK), Helsinki, Finland. He is one of the Principal Investigators in SMARAD (Smart and Novel Radios) Center of Excellence in Radio and Communications Engineering nominated by the Academy of Finland. During his sabbatical leave in 2006–2007, he was Nokia Visiting Fellow at Nokia Research Center as well as Visiting Fellow at Princeton University, Princeton, NJ. His research interest include statistical, communications and sensor array signal processing. He has published more than 200 papers in international scientific conferences and journals.

Dr. Koivunen co-authored the papers receiving the Best Paper Award in IEEE PIMRC 2005, EUSIPCO 2006, and EuCAP 2006. He has served as an associate editor for IEEE SIGNAL PROCESSING LETTERS, *Signal Processing*, and the *Journal of Wireless Communication and Networking*. He is a member of the IEEE Signal Processing for Communication Technical Committee (SPCOM-TC). He is the general chair of the IEEE SPAWC (Signal Processing Advances in Wireless Communication) 2007 Conference in Helsinki in June 2007.