LPI RADAR WAVEFORM CLASSIFICATION USING BINARY SVM AND MULTI-CLASS SVM BASED ON PRINCIPAL COMPONENTS OF TFI

Almıla BEKTAŞ and Halit ERGEZER

Çankaya University, Mechatronics Engineering Department, Ankara, TURKEY

ABSTRACT. Since cognition has become an important topic in Electronic Warfare (EW) systems, Electronic Support Measures (ESM) are used to monitor, intercept and analyze radar signals. Low Probability of Intercept (LPI) radars are preferred to be able to detect targets without being detected by ESM systems. Because of their properties as low power, variable frequency, wide bandwidth, LPI Radar waveforms are difficult to intercept with ESM systems. In addition to intercepting, the determination of the waveform types used by the LPI Radars is also very important for applying counter-measures against these radars. In this study, a solution for the LPI Radar waveform recognition is proposed. The solution is based on the training of Support Vector Machine (SVM) after applying Principal Component Analysis (PCA) to the data obtained by Time-Frequency Images (TFI). TFIs are generated using Choi-Williams Distribution. High energy regions on these images are cropped automatically and then resized to obtain uniform data set. To obtain the best result in SVM, the SVM Hyper-Parameters are also optimized. Results are obtained by using one-against-all and one-against-one methods. Better classification performance than those given in the literature has been obtained especially for lower Signal to Noise Ratio (SNR) values. The cross-validated results obtained are compared with the best results in the literature.

1. INTRODUCTION

Electronic Warfare (EW) systems are involved in LPI Radars as they are unlikely to intercept and it is hard to analyse them in detail while they track the targets [1]. By using the information provided by ES systems, Electronic Counter Measures (ECM)

✉ a.bektas@cankaya.edu.tr; halitergezer@cankaya.edu.tr
🆔 0000-0002-5253-7825; 0000-0002-7280-0416

are used for interfering the radar operation by providing false information and noise, that is why LPI Radar waveform recognition is very important for EW systems.

Different techniques are used to insert signal data into the classifiers such as using raw data, filtered data [2], arrays or Time-Frequency Images. Some of the Time-Frequency Analysis (TFA) techniques used for LPI Radar waveform recognition are Choi Williams Distribution (CWD) [1], [3] - [7], Wigner-Ville Distribution (WVD) [1], [4], [7], Radon-WVD [8], short-time Fourier Transform (STFT) [9] and discrete Fourier Transform(DFT) [10]. By using results obtained from these distributions, images that show various frequency over time are generated. These images are called Time-Frequency Images. In literature, the commonly used TFA technique for LPI Radar waveform recognition is CWD because WVD-TFI contains interfering ghost terms and these terms degrade the performance of classification. The effect of different TFA techniques on LPI Radar classification may be the subject of another study. Similar to the ones in the literature, TFIs used in this paper are generated by using CWD. Unlike the methods given in the literature, during the generation of TFIs simple operations that do not require much processing time are performed. While converting the time-frequency graph of the CWD transform, the appropriate parameters used to improve image quality and pixel values are normalized to clarify the distinctive features.

Various techniques have been proposed for classification by using Time-Frequency Images. As a LPI Radar Waveform Recognition Technique [3], [6] uses the Convolutional Neural Network (CNN). The computational cost has become a problem because of the generated TFI sizes. [3] proposes a sampling averaging technique (SAT) which provides a higher sampling rate with a lower computational cost. However, because of the CNN complexity, TFIs are re-sized by using nearest-neighbour interpolation. But by using interpolation, some distinctive features of TFIs may be lost. Instead of this, taking the principal components of the signals that carry most of the information is a more accurate way to decrease computational complexity. It is indicated in [3] that to reduce the number of signal samples, consecutive signal samples are averaged by applying coherent summation, but it is not clear how to guarantee coherency in real-world applications and the effect of coherency on the performance of classification. In this paper, to reduce the computational cost of TFA, Fractional Fourier Transform is utilized. [8] uses Wigner Ville Distribution (WVD) to obtain TFI. Before creating WVD-TFIs, a basic threshold was applied to the amplitude of the signals to fix the raised spikes that the phase changes caused. As a classification technique, they propose the Fractional Fourier transform (FRT) to reduce the computational cost. Proposed LPI Radar waveform recognition technique excludes the four Poly-time signals (T1, T2, T3, T4). The classification performances of [7] are lower than other techniques, although

they classified fewer signal types than others. [4] uses a single-shot multi-box detector (SSD) to detect both Continuous Wave (CW) and Pulse Wave(PW) signals and a supplementary classifier to classify signals that cannot be classified using SSD. In [4], it is mentioned that LPI waveform recognition has been performed for both CW and Pulsed Signals. It is also stated that recognition of CW LPI signals is challenging due to the lower peak power values of CW signals. However, when we evaluate the CW and Pulsed signals having the same average power value, CW signals are clearly distinguishable and traceable in TFI as a continuous curve along the *time axis*. This is not the case for Pulsed signals. For Pulsed signals, small zones of high intensity are observed in certain regions of the TFI. In our study, CW signals are not considered for classification, but our approach can also be applied to CW signals. The rationale behind this is that the highest energy region along the *time axis* is selected during the automatic crop of the respective zones in the TFI. Therefore, it becomes clear that the average power of the signal is important, not the peak power. LPI Radar waveform recognition techniques based on multi-layer perceptron, radial basis function and probabilistic neural networks are utilized in [7]. To minimize the information loss, classification is done based on deep sparse capsule networks in [11] and they used cross ambiguity functions for feature extraction. [5] proposed an image fusion algorithm and used CNN for feature extraction. Yet, poly-phase Frank code was excluded in classification. As a result, the similarity between Frank and other poly-phase codes is not examined. In this case, to compare [5] with other techniques that include both poly-phase codes and Frank code does not give an accurate result. Also, clustering [9], decision trees and SVM [5], [12] are commonly used techniques for classification.

In this paper Linear Frequency Modulation (LFM), Costas, Binary Phase Shift Keying (BPSK), Frank, P1, P2, P3, P4, T1, T2, T3, T4 signals are used for classification. Time-frequency images are generated using Choi-Williams Distribution [13].

All images are cropped automatically to make them include maximum information and then resized to a constant dimension for all signals to keep data vectors uniform. SVMs are used to train the dataset. Both binary classification and multi-class classification methods are used. For multi-class classification, Directed Acrylic Graph SVM (DAGSVM) method is used which is based on Decision Directed Acyclic Graph (DDAG) [23].

Support Vector Machine is an effective classification algorithm that is suitable for LPI Radar classification due to its good generalization performance. A major limitation of SVM is the high demand for memory during training. To overcome this limitation, Principal Component Analysis (PCA) is applied to TFIs. Unlike other

methods proposed in the literature, principal components have been used instead of resizing the images that cause entropy increase.

There are a lot of parameters to be turned in SVMs such as choosing the "right" kernel, regularization penalties, and the slack variable. These parameters must be optimized to find the best generalization. Firstly, SVMs are trained by using different Kernel Functions. We notice that different types of LPI signals can be classified better with different Kernel Functions. Then, it is decided to apply Hyper-Parameter Optimization to the Kernel Functions as well. Therefore, the optimal Kernel Function selection is automatically done by Hyper-Parameter Optimization. Then, the optimal parameter sets are used for each SVM. 5-fold cross-validation results of both binary classification and multi-class classification methods are presented. For both of the methods, the same folds are used to be able to compare the performance of binary classification method and multi-class classification method. All SNR values of TFIs between -20dB to 10dB are uniformly distributed and fairly separated between folds.

The paper is organized as follows. Firstly, the mathematical expressions are given in Section 2. The TFA technique used in the proposed solution is expressed and LPI Radar Signal TFIs are shown in Section 3. Then, the proposed solution is explained in detail in Section 4. Section 5 contains the results and comparisons with results in the literature and then all sections are concluded in 6.

2. LPI RADAR SIGNALS

Mathematical explanations of LPI Radar signals used for the proposed solution are given below.

A. Frequency Modulation Continuous Wave (FMCW)

Frequency modulation of FMCW signals is created by using increasing and decreasing frequency signals. Increasing and decreasing frequencies f_i and f_d are [1]:

$$f_i = \left(f_0 - \frac{\Delta F}{2}\right) + \frac{\Delta F}{t_m} t \tag{1}$$

$$f_d = \left(f_0 + \frac{\Delta F}{2}\right) - \frac{\Delta F}{t_m} t \tag{2}$$

respectively, where f_0 is the carrier frequency, t is time, ΔF is modulation bandwidth and t_m is modulation period. The modulation bandwidth of 250, 500,750 and 1000Hz are used to generate FMCW signals.

B. **Binary Phase Shift Keying (BPSK)**

Phase modulation of the BPSK signal is provided by Barker Sequences with various lengths. A Barker Sequence B = [b_0, b_1, ..., b_n] contains +1's and -1's of length n $\geq$ 2. In this paper, Barker Sequence length of 7, 11 and 13 are used to generate BPSK signals.

C. **Costas**

Costas signal contains a set of frequencies that are chosen from the available frequencies $f_1, f_2, f_3, ..., f_m$. Frequency modulation of the carrier signals is provided by the set [22]. When creating the dataset to train SVMs, 3 different sets of frequencies with different lengths are used.

D. **Polyphase**

Polyphase signals' modulation is applied by phase functions given in equations (4) to (7). In order to approach a stepped or linear frequency modulation, the number of phase states is varied and the time spent at each phase state is constant. 5 different phase functions for polyphase signals are given [22]:

Frank

$$\Phi_{FR}(i,j) = \frac{2\pi}{N_p}(i-1)(j-1) \tag{3}$$

P1

$$\Phi_{P1}(i,j) = \frac{-\pi}{N_p}\left(N_p - (2j-1)\right)\left((j-1)N_p + (i-1)\right) \tag{4}$$

P2

$$\Phi_{P2}(i,j) = \left\{\frac{\pi}{2}\left[\left(\frac{N_p-1}{N_p}\right) - \left(\frac{\pi}{N_p}\right)(i-1)\right]\right\}[N_p + 1 - 2j] \tag{5}$$

where $i, j=1...N_p$. N_p is the number of phase states.

P3

$$\Phi_{P3}(k) = \frac{\pi}{N_p}(k-1)^2 \tag{6}$$

P4

$$\Phi_{P4}(k) = \left(\frac{\pi}{N_p}(k-1)^2 - \pi(k-1)\right), \tag{7}$$

where k=1, ..., N_p and N_p is the number of phase states.

E. **Polytime**

As it is stated earlier in the Polyphase section, the time spent at each phase state is constant. In Poly-time signals the number of phase states is user-defined and the time spent at the phase states is varied to approach a stepped or linear frequency modulation. Phase functions of Poly-time codes are [1]:

T1

$$\Phi_{T1}(t) = mod\left\{\frac{2\pi}{N_p}\left\lfloor(mt-jT)\frac{jN_p}{T}\right\rfloor, 2\pi\right\}$$

$$\tag{8}$$

T2

$$\Phi_{T2}(t) = mod\left\{\frac{2\pi}{N_p}\left\lfloor(mt-jT)\left(\frac{2j-m+1}{T}\right)\frac{N_p}{2}\right\rfloor, 2\pi\right\}$$

$$\tag{9}$$

T3

$$\Phi_{T3}(t) = mod\left\{\frac{2\pi}{N_p}\left\lfloor\frac{N_p\Delta F t^2}{2t_m}\right\rfloor, 2\pi\right\}$$

$$\tag{10}$$

T4

$$\Phi_{T4}(t) = mod\left\{\frac{2\pi}{N_p}\left\lfloor\frac{N_p\Delta F t^2}{2t_m} - \frac{N_p\Delta F t}{2}\right\rfloor, 2\pi\right\}, \tag{11}$$

where j = 0, 1, 2, …, m-1 and m is the number of frequency segments, N_p is the number of phase states and T is overall code period.

3. TIME-FREQUENCY ANALYSIS TECHNIQUE: CHOI-WILLIAMS DISTRIBUTION

Choi Williams Distribution (CWD) [13], [14] has been used to create time-frequency images of the LPI Radar signals. It is a time-frequency analysis technique that is used to extract data from the signal. CWD is included in Cohen's generalized class as a time-frequency distribution. It uses an exponential kernel function while Wigner-Ville Distribution (WVD) uses a kernel as one. The exponential kernel function makes CWD different than other distributions. CWD is expressed in discrete form as:

$$CWD_x(t, \omega) = 2\sum_{\tau=-\infty}^{\infty} e^{-j2\omega\tau} \sum_{\tau=-\infty}^{\infty} \frac{1}{\sqrt{4\pi n^2/\sigma}} e^{-\sigma(\mu-t)^2/(4\tau^2)} x(\mu+\tau)x^*(\mu-\tau) \quad (12)$$

where t is the time index, ω is the angular frequency, $x(\mu)$ is the time signal and $x^*(\mu)$ is its complex conjugate, τ is the time delay and σ is scaling factor [15].
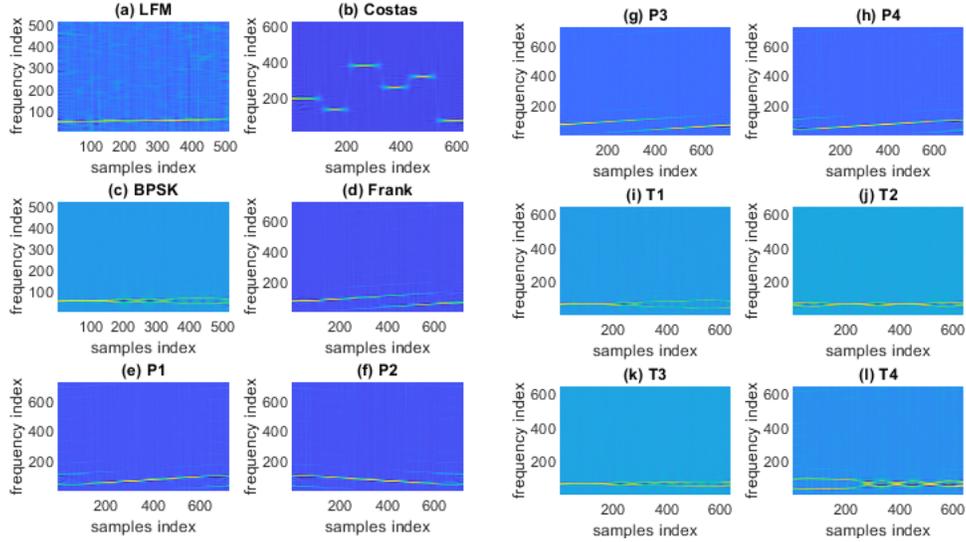

FIGURE 1. 12 LPI Radar signals TFIs that are created by using CWD. All signals created with a SNR value of 0dB.

CWD images of each LPI Radar signal and No-Modulation signal have created at 16 different SNR values, which varies -20dB to 10dB with a step of 2dBs. All of the signals are created 20 different times with 20 different random noises to make the effect of noise more realistic. It is mentioned in Section 2 that some of the signals

are created with different parameters. In Figure 1, 12 different LPI Radar signals at 0dB are shown.

4. PROPOSED LPI RADAR WAVEFORM CLASSIFICATION TECHNIQUE

The block diagram of the proposed technique is given in Figure 2. To recognize 12 different LPI waveforms given in Section 2 and the No-Modulation signal, signals are sampled at a sampling rate of 20 kHz. Then, Choi Williams Distribution is applied to the sampled data. The output CWD-TFI is automatically cropped according to the high energy regions on the image. Cropped images are resized to a fixed size in order to provide uniformity when creating the dataset. This process has been repeated for every image which has different waveforms, parameters and SNR values. A total of N (=33600) images have been created in RGB format, each with a dimension of 64x64 pixels. The resulting RGB images are transformed into feature vectors of size M (=64x64x3 = 12288). Since we have used RGB format instead of transforming TFIs into greyscale images, distinctive features of TFIs are preserved. All feature vectors are collected into a NxM matrix to create the dataset which is the input of PCA. With the application of PCA, a new NxK_e dataset matrix is created using the features that contain the highest information. K_e is the number of features with the highest information in the feature vector. By this way, the data load is reduced. These principal components are used to train SVM. For one-against-all SVMs, the signal type to be trained is labelled as +1 and all other signal types are labelled as -1. For example, to train SVM for BPSK, the BPSK images are labelled as +1, other classes corresponding to 11 LPI signal types and No-Modulation signal are labelled as -1. While training the classifier, the parameters including kernel types and kernel parameters are optimized to obtain the best scenario. For each signal type, 5 SVMs are trained by 5-fold cross-validation. A total of 65 SVMs are trained for one-against-all method and a total of 390 SVMs are trained for one-against-one method. At the end of this process, we have PCA centres and trained classifier.

SVM is an algorithm that is generally used to classify the data with a boundary. The total distance between the closest feature vectors of both and the boundary is called margin. Assume (x_i, y_i) is a feature vector where x_i is the feature and y_i is the class label for $i = 1, 2, …, N$. $x_i \in IR^p$ and $y_i \in \{-1, 1\}$.

A hyperplane is defined by

$$\{x: f(x) = x^T\beta + \beta_0 = 0\} \tag{13}$$

where β is a unit vector. Classification rule is:

A. BEKTAŞ, H. ERGEZER

$$G(x) = sign[x^T\beta + \beta_0].$$ (14)

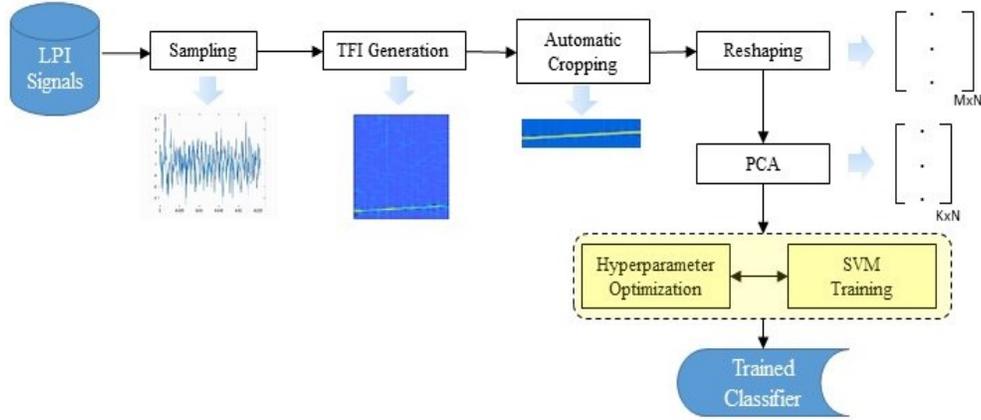The correct classification satisfies $y_i f(x_i) > 0 \,\forall_i$.

FIGURE 2. The flow diagram of the proposed LPI Radar Waveform classification technique.

Then, a hyperplane that creates the biggest margin *m* between points of both classes can be written as [16]:

$$y_i(x_i^T\beta + \beta_0) \geq m, \;\; i = 1, ..., N$$ (15)

In classification, the feature vectors of classes can be linearly-separable or not. For a non-linearly separable case, it is possible to create more flexible boundaries that minimize the amount of misclassified vectors. By designing a hyperplane classifier into a new k-dimensional space using the vectors which are the closest to the decision boundary, the non-linear case can also be classified. Mapping vectors into a high dimensional space provided by kernel operation.

It is stated that the computational complexity of SVM is independent of the dimensionality of the kernel space [17]. Thus, SVM can be used for a number of high dimensional models which make it a highly applicable algorithm. Since CWD creates a 3-dimensional TFI, the independence of computational complexity from the dimensions has become advantageous for the proposed technique.

Some of the commonly used Kernel functions (K_f) are described as:

d^{th}-Degree polynomial: $K_f(x, x') = (1 + \langle x, x' \rangle)^d,$

$$(16)$$

Radial basis: $K_f(x, x') = exp\left(-\gamma ||x - x'||^2\right),$

$$(17)$$

Hyperbolic tangent [16].: $K_f(x, x') = tanh(k_1 \langle x, x' \rangle + k_2)$

$$(18)$$

There is no practical method to choose the best kernel function yet. It is also a challenging problem. For this purpose, No-Modulation signal and 12 LPI Radar signals are trained by SVM using 3 different kernel functions. Linear, 2nd order polynomial and Radial Basis Function are used as Kernel Function for classification. Then, different Kernel Functions are used for different signal types by using Hyper-Parameter Optimization.

During the training process, Hyper-Parameter optimization is applied. There may be multiple boundaries created by SVM that classifies the data. The most important parameters that characterize the boundary are penalty factor and Kernel parameters. By using these parameters, dozens of boundaries can be created that give correct classification. Finding the boundary that provides the most effectivity with minimum error is the objective of the SVM algorithm. Therefore, it becomes necessary to optimize the penalty factor and kernel parameters. These parameters are called Hyper-Parameters of SVM.

Bayesian Optimization [18] method is used for Hyper-Parameter Optimization. Bayesian Optimization is done by using a probabilistic model and an acquisition function. The probabilistic model is based on the observations that were made previously. The probabilistic model gives primary knowledge to find the locations of potential Hyper-Parameters. Acquisition function is used by defining an exploration ratio after each iteration and program evaluates whether the next point is over-exploiting or not. Exploration ratio provides the balance between exploring new points vs. concentrating near points. By exploring different points according to the given acquisition function, it results in a model.

As acquisition function, expected improvement (EI) plus is used.

$$E[I(\lambda)] = \left(f_{min} - \mu(\gamma)\right)\Phi\left(\frac{f_{min} - \mu(x)}{\sigma}\right) + \sigma\phi\left(\frac{f_{min} - \mu(\gamma)}{\sigma}\right) \qquad (19)$$

where γ is a model of predictions which follows the normal distribution, $\Phi(*)$ is standard normal distribution function and $\phi(*)$ is standard normal density function and f_{min} is the best-observed value so far [19].

The blue points in Figure 3 show the explored Hyper-Parameter values with respect to estimated objective function values. As it can be understood from the model, the parameters with the highest estimated objective function values are the optimal Hyper-Parameters.

As a conclusion, the best scenario for SVM is created by optimizing parameters and using different Kernel functions.
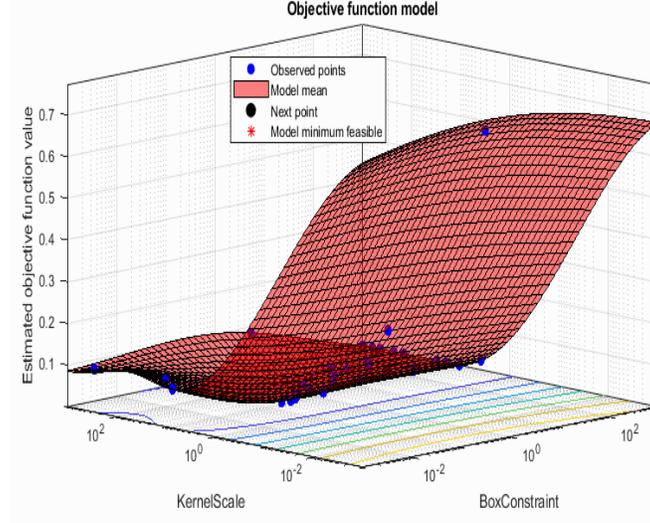

FIGURE 3. The exploration of objective function vs. SVM parameters during Hyper-Parameter Optimization

However, SVM is a classifier that requires high computational cost for the training process. In order to reduce the computational cost, principal components of the data matrix are obtained by computing the eigenvalues and eigenvectors. Then, K_e largest eigenvalues are chosen to form a transformation matrix **A** that contains eigenvectors. Each *1xM (=1x12288)* dimensional feature vector **x** transformed into a new *1xK_e* dimensional vector y that carries most of the information [20]. It can be calculated by

$$\mathbf{y} = \mathbf{A}^T\mathbf{x} \tag{20}$$

The correlation matrix S_y is determined by using (32)

$$S_y = E\,[\mathbf{y}\mathbf{y}^T] = E[\mathbf{A}^T\mathbf{x}\mathbf{x}^T\mathbf{A}] = \mathbf{A}^T S_x \mathbf{A} \tag{21}$$

where S is the diagonal matrix of eigenvalues. E [*] indicates the expected value of *.

By using the mean square error approximation, the K_e value can be specified for given conditions.

$$x = \sum_{i=0}^{M-1} y(i)a_i \ and \ x' = \sum_{i=0}^{K_e-1} y(i)a_i.$$

Representing x by using x' gives a mean square error of

$$E[\|x - x'\|^2] = E\left[\left\|\sum_{i=K_e}^{M-1} y(i)a_i\right\|^2\right] \tag{22}$$

By eigenvector definition, the mean square error is determined using

$$\sum_{i=K_e}^{M-1} a_i^T E[xx^T] a_i = \sum_{i=K_e}^{M-1} \lambda_i \tag{23}$$

As a result, to specify a K_e for a given mean square error can be determined by (23), [17].
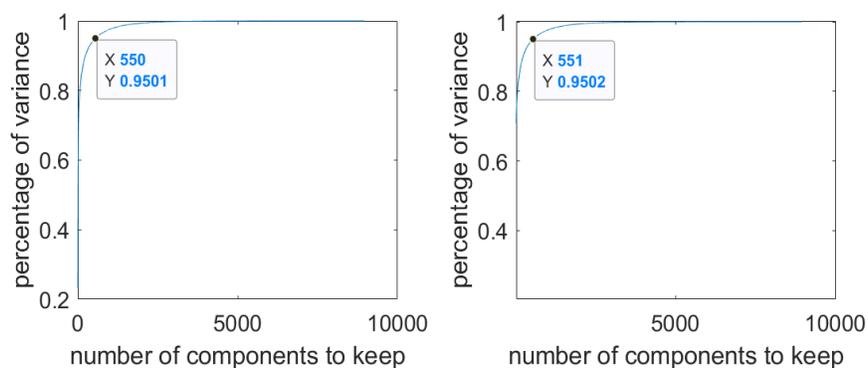
FIGURE 4. Percentage of variance vs. number of components to keep graph for PCA.

In the proposed technique, the mean square error of 5% is used where the percentage of variance becomes smaller than 0.01%. The result of the change of percentage of variance value is shown in Figure 4. A trial and error process is applied to set a mean square error value. In fact, the mean squared error value can be increased with a very little amount of information loss. Also, it results in a considerable reduction in the data load. However, trying to reduce the data load with different mean squared error values which are greater than 5% result in a performance loss in classification. As can be seen in Figure 4, from a certain point, the change in the percentage of variance becomes much smaller than before, meaning that from this point, features after K^{th} feature will make minor differences in classification.

By using PCA, around 550 of 12288 features are kept for one-against-all classification, around 130 of 12288 for one-against-one classification to represent 95% of each feature vector. The computational cost problem of SVM is reduced depending on PCA. The dataset with principal components makes the SVM an appropriate classification algorithm for LPI Radar waveform classification.

Since there are C=13 classes, it is needed to train $C*(C-1)/2$ = 78 binary SVMs. These SVMs are used to construct DDAG.

5. RESULTS

In the proposed technique, one-against-one and one-against-all methods are applied. For the one-against-one method, DDAG is used to find the best class. DDAG structure is given in Figure 5. The Confusion Matrix of the one-against-one method is shown in Table 6. For the one-against-all method, the maximum score-values of 13 SVMs are used to find the best class. The Confusion Matrix of the one-against-all method is shown in Table 5.
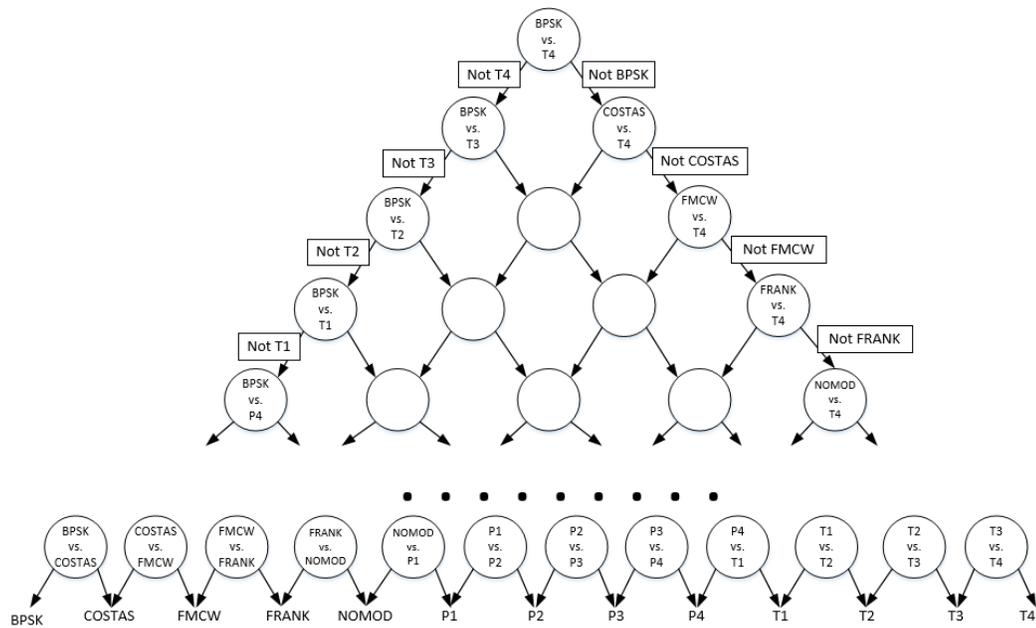

FIGURE 5. The Decision Directed Acyclic Graph for finding the best class out of 13 classes. (Not all of the nodes has been shown for the sake of clarity).

In addition, there are 4 important parameters that qualify the classification performance. *Accuracy*, *recall*, *precision* and *F-measure*. *Accuracy*, i.e. classification rate can be calculated by [21]:

$$Overall\ Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{24}$$

Where *TP* is the correctly classified feature vectors with label +1, *FP* is the misclassified feature vectors with label +1 and *TN* is the correctly classified feature vectors with label -1, *FN* is the misclassified feature vectors with label -1.

The *overall accuracy* of the one-against-all method is 98.73%. The poly-phase signals especially P1 and P4 are misclassified in between. The results of binary classification and multi-class classification are similar; the *overall accuracy* of the one-against-one method is 98.61%.

The *recall* is the ratio of the total number of correctly classified vectors of the class with label +1 and all vectors of the class with label +1. The *recall* value closer to 1 shows that the class with label +1 is correctly classified.

For a given feature vector labelled as +1, the *precision* gives the percentage that the given vector is actually labelled as +1. *Precision* is given by [21]:

$$Precision = \frac{TP}{TP+FP} \tag{25}$$

F-Measure represents both the precision and recall by taking their harmonic mean to save the balance in between. F-Measure is given by [21]:

$$F\_Measure = \frac{2 \times Recall \times Precision}{Recall + Precision} \tag{26}$$

Classification performance parameters of one-against-all and one-against-one methods are given in Table 1 and Table 2, respectively.

Table 1. Accuracy, Recall, Precision, F-Measure values for one-against-all method

Accuracy	0.98730
Recall	0.98989
Precision	0.99237
F-Measure	0.99113

Table 2. Accuracy, Recall, Precision, F-Measure values for one-against-one method

Accuracy	0.98610

Recall	0.98996
Precision	0.99174
F-Measure	0.99085

The *recall*, *precision* and *F-measure* values cannot be compared to existing methods because there are no such values presented in these articles.

Table 3. Classification accuracy value comparison with other proposed techniques

	Proposed Technique	[3]	[4]	[5]	[8]
BPSK	98.96%	99.00%	66.17%	98.00%	NC**
COSTAS	100%	99.00%	77.67%	99.00%	NC**
LFM	100%	94.00%	68.58%	99.00%	95.00%
FRANK	100%	89.00%	65.83%	NA	85.00%
P1	87.35%	86.00%		86.00%	85.00%
P2	98.8%	100%	68.08%	100%	95.00%
P3	96.69%	91.00%	64.5%	98.00%	85.00%
P4	84.55%	85.00%	*64.08%	91.00%	85.00%
T1	100%	97.00%	66.00%	84.00%	NC**
T2	100%	94.00%	94.50%	92.00%	NC**
T3	98.36%	95.00%	68.83%	91.00%	NC**
T4	96.72%	94.00%	NC**	99.00%	NC**
BFSK***	NC**	NC**	NC**	NC**	95.00%
No Modulation	99.12%	NC**	NC**	NC**	95.00%

*In [4], P3 signal is taken as two different signal objects and one of this part and P4 are considered as one class. The other part of the P3 signal is taken as class P3. ** NC: Not Considered. *** BFSK signal stands for Binary Frequency Shift Keying and used only in [8].

Results of the proposed solution and [3], [4], [5] and [8] are given in Table 3. Since the *precision* and F-*Measure* values are not given in any of these, the reliability of these accuracy values can be arguable. In [3] the Confusion Matrix is created by using the results that have SNR value of -6dB. But Table 4 and Table 5 show the classification performance of all the SNR values examined including -18 and -20dB. Lower SNR values have substantial effects on classification performance. All of the accuracy values except the Costas Signal are higher in the proposed solution with the effect of lower SNRs. Therefore, the performance of the proposed solution can be said to be better. There is an abrupt change of classification performance for SNR values smaller than -12dB. In our technique, by the application of PCA, the confusing elements of the images are also eliminated, so the classification performance at these SNR values are also higher.

[4] proposes a different technique to classify Frank, P1, P3 and P4 codes. The Confusion Matrix is created by using lower SNR values so the accuracy values are lower than other techniques. But the point where the abrupt change occurs that is mentioned earlier is -14dB in [4]. For lower SNR values, the results are similar to the results obtained from the solution that we propose. However, the precision and *F-Measure* values are not mentioned in [4] either, so the overall results cannot be compared properly.

LPI Radar waveform recognition technique that is proposed in [5] gives better results for P3 and P4 signals. For poly-time signals, especially for T1, T2 and T3, there are major differences in the accuracy. The results that we obtained are much higher than the results in [5]. Since the Frank signal is excluded in [5], the similarity with the other poly-phase codes are not examined in between.

[8] used higher SNR values and fewer signal types for classification. However, their classification results are lower. For P4 signal, the classification performance is very close to the result that we obtained. But the resolution of the result of [8] is lower than ours and 84,55% can also be rounded to 85,00%. So they can be assumed to be the same. Simulations have been performed on the computer having 3.7 GHz processor and 16 GB memory. The required time to train one SVM in the one-against-all method is 1838 seconds on average, including PCA and Hyper-Parameter Optimization. The required time to train one SVM in the one-against-one method is 328 seconds on average, including PCA and Hyper-Parameter Optimization.

6. CONCLUSIONS

In this paper, an automatic LPI Radar classification method has been performed using SVM and Principal Component Analysis. SVMs are trained using methods of one-against-all and one-against-one with DDAG. Up to the PCA step, all distinctive features have been preserved.

Table 4. Confusion Matrix of the one-against-all method.

		SVM RESULTS											
		BPSK	COSTAS	FMCW	FRANK	NOMOD	P1	P2	P3	P4	T1	T2	T3	T4
ACTUAL SIGNAL TYPE	BPSK	0.9896	0.0000	0.0000	0.0000	0.0083	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0021	0.0000
	COSTAS	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	FMCW	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	FRANK	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	NOMOD*	0.0088	0.0000	0.0000	0.0000	0.9912	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	P1	0.0000	0.0000	0.0000	0.0000	0.0000	0.8735	0.0158	0.0065	0.1011	0.0000	0.0031	0.0000	0.0000
	P2	0.0000	0.0000	0.0000	0.0000	0.0000	0.0031	0.9880	0.0029	0.0060	0.0000	0.0000	0.0000	0.0000
	P3	0.0000	0.0000	0.0000	0.0000	0.0000	0.0069	0.0232	0.9669	0.0029	0.0000	0.0000	0.0000	0.0000

	BPSK	COSTAS	FMCW	FRANK	NOMOD	P1	P2	P3	P4	T1	T2	T3	T4
P4	0.0000	0.0000	0.0000	0.0000	0.0000	0.1113	0.0191	0.0172	0.8455	0.0069	0.0000	0.0000	0.0000
T1	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000
T2	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000
T3	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0035	0.0000	0.0000	0.9836	0.0129
T4	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0328	0.9672

*NOMOD abbreviation is used instead of No-Modulation signal for the sake of simplicity.

Contrary to previous studies, operations such as averaging and converting to grey level are avoided. The Hyper-Parameters of SVMs has been optimized including kernel type by using Bayesian Optimization. The performances of methods have been obtained through simulations using different SNR levels. The overall accuracy level of 98.60% has been obtained from one-against-one and 98.73% has been obtained from the one-against-all method. The measure of classification performance is evaluated by calculating *F-measures*. It is observed that the classification performances are satisfactory compared to existing methods, except for -18 dB and -20 dB SNR values.

For low SNR values, the accuracy levels are also low for existing methods. The computational complexity of the method is enough for real-world ESM applications since the number of feature vectors is reduced using SVD. It can be concluded that the confused waveform cannot be considered as a big problem since the main objective of the LPI classification is to assign the jamming method against the LPI Radar, and similar jamming methods can be applied to confused LPI waveforms.

TABLE 5. *Confusion Matrix of the one-against-one method.*

		SVM RESULTS											
		BPSK	COSTAS	FMCW	FRANK	NOMOD	P1	P2	P3	P4	T1	T2	T3	T4
ACTUAL SIGNAL TYPE	**BPSK**	0.9969	0.0000	0.0000	0.0000	0.0083	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0021	0.0000
	COSTAS	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	FMCW	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	FRANK	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	NOMOD	0.0055	0.0000	0.0000	0.0000	0.9945	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	P1	0.0000	0.0000	0.0000	0.0000	0.0000	0.8438	0.0250	0.0063	0.1250	0.0000	0.0000	0.0000	0.0000
	P2	0.0000	0.0000	0.0000	0.0000	0.0000	0.0063	0.9625	0.0281	0.0031	0.0000	0.0000	0.0000	0.0000
	P3	0.0000	0.0000	0.0000	0.0000	0.0000	0.0063	0.0469	0.9438	0.0031	0.0000	0.0000	0.0000	0.0000
	P4	0.0000	0.0000	0.0000	0.0000	0.0000	0.1063	0.0219	0.0188	0.8531	0.0000	0.0000	0.0000	0.0000
	T1	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000
	T2	0.0000	0.0000	0.0000	0.0031	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.9969	0.0000	0.0000
	T3	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.9688	0.0313
	T4	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0250	0.9750

REFERENCES

[1] Pace, P.E., Detecting and Classifying Low Probability of Intercept Radar. 2nd Ed., Artech House, Norwood, MA, USA, 2009.

[2] Tao, R., Li, B., Sun, H., Research Progress of the Algebraic and Geometric Signal Processing, *Defence Technology*, 9(1) (2013), 40-47.

[3] Kong, S.H., Kim, M., Hoang, L.M., Kim, E., Automatic LPI Radar Waveform Recognition Using CNN, *IEEE Access*, 6 (2018), 4207-4219.

[4] Hoang, L.M., Kim, M., Kong, S.H., Automatic Recognition of General LPI Radar Waveform Using SSD and Supplementary Classifier, *IEEE Transactions on Signal Processing*, 67(13) 2019, 3516-3530.

[5] Gao, L., Zhang, X., Gao, J., You, S., Fusion Image Based Radar Signal Feature Extraction and Modulation Recognition, *Access IEEE*, 7 (2019), 13135-13148.

[6] Huang, Z., Ma, Z., Huang, G., Radar Waveform Recognition Based on Multiple Autocorrelation Images, *Access IEEE*, 7 (2019), 98653-98668.

[7] Gulum, T., Autonomous Non-Linear Classification of LPI Radar Signal Modulations, https://calhoun.nps.edu/handle/10945/3302; 2007 [accessed 24 September 2019].

[8] Kishore, T.R., Rao, K.D., Automatic intrapulse modulation classification of advanced LPI radar waveforms, *IEEE Trans. Aerosp. Electron. Syst.,* 53(2) (2017), 901-914.

[9] Tong, X., Modelling and realization of real time electronic countermeasure simulation system based on SystemVue, *Defence Technology*, 2019.

[10] Deng, B., Luan, J., Cui, S., Analysis of parameter estimation using the sampling-type algorithm of discrete fractional Fourier transform, *Defence Technology*, 10(4) (2014), 321-327.

[11] Liu, M., Liao, G., Yang, Z., Song, H., Gong, F., Electromagnetic Signal Classification Based on Deep Sparse Capsule Networks, *Access IEEE*, 7 (2019), 83974-83983.

[12] Zeng, X., Wang, S., Bark-wavelet Analysis and Hilbert–Huang Transform for Underwater Target Recognition, *Defence Technology*, 9(2) (2013), 115-120.

[13] Choi, H.I., Williams, W.J., Improved time-frequency representation of multicomponent signals using exponential kernels, *IEEE Trans. Acoust., Speech, Signal Process.,* 37(6) (1989), 862-871.

[14] Hollinger, K.B., Code optimization for the Choi-Williams distribution for ELINT applications, https://calhoun.nps.edu/handle/10945/4422; 2009 [accessed 24 September 2019].

[15] Liu, Y., Xiao, P., Wu, H., Xiao, W., LPI radar signal detection based on radial integration of Choi-Williams time-frequency image, *Journal of Systems Engineering and Electronics*, (2015), 973-981.

[16] Hastie, T., Tibshirani, R., Friedman, J., The Elements of Statistical Learning, 2nd Ed., Springer New York Inc, New York, NY, USA, 2001.

[17] Theodoridis, S., Koutroumbas, K., Pattern Recognition, 4th Ed., Academic Press. Inc., Orlando, FL, USA, 2009.

[18] Snoek, J., Larochelle, H., Adams, R.P., Practical Bayesian optimization of machine learning algorithms, *Advances in Neural Information Processing Systems*, 25 (2012), 2960-2968.

[19] Feurer, M., Hutter, F., Hyperparameter Optimization. In: Hutter F, Kotthoff L, Vanschoren, J., editors, Automated Machine Learning, *Cham: Springer International Publishing*, (2019), 3-33.

[20] Duda, R.O., Hart, P.E., Stork, D.G., Pattern Classification, 2nd Ed., John Wiley & Sons, Inc. New York, 2001.

[21] Powers, D.M.W., Evaluation: From Precision, Recall and F-Measure to Roc, Informedness, Markedness & Correlation, *Journal of Machine Learning Technologies*, 2(1) (2011), 37-63.

[22] Lima, A.F., Analysis of low probability of intercept (LPI) radar signals using cyclostationary processing, https://calhoun.nps.edu/handle/10945/4944, 2002 [accessed 24 December 2019].

[23] Platt, J. C., Cristianini, N., Shawe-Taylor, J., Large margin DAGs for multiclass classification, *Advances in Neural Information Processing Systems*, MIT Press,12 (2000), 547–553.