
Changchun Zhang, Zhen Hu, Terry N. Guo, R. C. Qiu, Kenneth Currie

Abstract—A Cognitive Radio Network (CRN) based Wireless Sensor Network (WSN), as an extension of CRN, is explored for radio frequency (RF) passive target intrusion detection. Compared to a cheap WSN, the CRN based WSN is expected to deliver better results due to its strong communication functions and powerful computing ability. Issues addressed in this paper include experimental architecture, waveform design, and machine learning algorithm for classification. In particular, passive target intrusion is experimentally demonstrated using multiple WARP platforms that serve as the cognitive/sensor nodes. In contrast to traditional localization methods relying on radio propagation properties, the technique used in this research is based on machine learning with measured data, considering complicated multipath environment and high dimensional sensing data collected by the CRN based WSN. Preliminary experimental results are quite encouraging, suggesting that a large-scale CRN based WSN supported by machine learning techniques has promising potential for passive target intrusion detection in harsh RF environments.

Index Terms—Passive Target Intrusion Detection, Machine Learning, Multi-class Support Vector Machine, Dimensionality Reduction

I. INTRODUCTION

Cognitive radio network (CRN) is designed to utilize the radio spectrum optimally for the next generation wireless communications network. Meanwhile, the wireless sensor network (WSN) which is based on communications network is also a practical trend for future WSN deployment as it saves infrastructure cost while taking advantage of the rich communications capabilities. The embedded spectrum sensing capability of CRN node can be used as the sensing function of the WSN node. Further, the function for unlicensed spectrum utilization in CRN is inherently fit for the large scale WSN due to the sharing of radio spectrum which increases the capacity of WSN with deploying hardware nodes as fewer as possible.

Passive target intrusion detection is a very important application of the WSN where the target is device free and the WSN actively senses and detects this kind target mainly through the intrusion effect on the environment under test as there is no direct collaboration between the sensing nodes and the target. This paper proposes and demonstrates a CRN based WSN which is composed of multiple pairs of transmitter and receiver, and applicable to the application of passive intrusion detection, based on the collected energy information by the sensing nodes.

For the passive target intrusion detection in this paper, the target is device free. In radar system, the TDOA or TOA can be used for localization and tracking [1]. However, the multi-path impact within the indoor environment is very significant. The distances between the target and sensors are small. It is not easy to measure the TDOA or TOA very accurately, with the normal communications system whose bandwidth is not wide enough. Thus the location of the target can not be derived from the distance information easily.

In this paper, the passive target intrusion detection is formulated to pattern classification problem. Each location where the target is placed can be mapped to the corresponding unique class. When the target is at different location within the detecting area, the unique pattern will be classified by analyzing some type of sensing data, like the received energy or radio phase at the distributed sensors. Currently, the received energy is used.

Orthogonal frequency-division multiplexing (OFDM) waveform is adopted as the radio signal in active sensing, while the bandwidth is fragmented and allocated to different sensing nodes acting as second users (SUs) in the CRN. In other words, the energy calculation at each receiver will only include the bandwidth allocated to this receiver even the transmitter sends the signal with full bandwidth. This method of energy data collection not only complies with the attribute of the CRN, but also extracts more energy diversity information based on radio spectrum within the indoor multi-path environment.

The machine learning algorithms, like multi-class support vector machine (SVM), are also investigated and demonstrated in this paper to implement the feature oriented intrusion detection over the radio energy data collected by the WSN. Specially, due to the high dimensional data generated by the large scale CRN, the dimensionality reduction can also involved in before the classification. Feature extraction and feature selection methods are usually used for reducing the dimensionality of the high dimensional network data.

The experimental demonstration of the proposed CRN based WSN is developed over our large scale CRN testbed, in which multiple WARP platforms act as the network nodes to collect the sensing data. One of these network nodes are configured as radio transmitter and the rest as receivers. They are controlled through Ethernet by a centralized computer on which the MATLAB is running to perform waveform design and configuration, offline computing and data fusion.
network control, etc. The experimental results provide the real evidence for the future of the CRN based WSN to handle high dimensional network data with machine learning algorithms.

The rest of this paper is organized as follows. The section II and III give the theoretical background review on the multi-class SVM and some typical methods of dimensionality reduction, such as PCA, MVU. Then the details of the experiment setup, procedures and experiment result are described in the section IV. In the section V, we conclude this paper and propose some future work and improvements.

II. MULTIPLE CLASS SUPPORT VECTOR MACHINE

Support Vector Machine [2] [3] is a typical supervised machine learning method in which the classifier is generated from the training dataset. Support Vector Machine is initially introduced to solve the two-classes problem. Suppose there is \( M \) training samples denoted as

\[
(x_i, y_i), i = 1, ..., M
\]

(1)

where \( x_i \) is \( N \) dimensional input training sample, while \( y_i \in \{-1, 1\} \) is the output that \( y_i = 1 \) for class 1 and \( y_i = -1 \) for class 2.

Two-class SVM tries to find a separating hyperplane denoted as

\[
w^T \cdot x + b = 0
\]

(2)

which maximizes the geometric margin [4] for all the training samples. This problem is formulated to the convex optimization problem to find the \( w \) and \( b \) as below:

\[
\min_{w, b} \|w\|^2
\]

s.t.

\[
y_i(w^T \cdot x_i + b) \geq 1
\]

(3)

With the help of Lagrangian, suppose \( \alpha = (\alpha_1, \alpha_2, ..., \alpha_M)^T \) and \( \alpha_i \) is the non-negative Lagrange multipliers, the solution to the above optimization problem is as below:

\[
w = \sum_{i=1}^{M} \alpha_i y_i x_i
\]

(4)

And \( b \) can be determined by the support vectors after the \( w \) is found. The support vectors are those training samples with the smallest margins to the hyperplane. Then we have decision function to classify class 1 and class 2:

\[
d(x^i) = w^T \cdot x^i + b = \sum_{i=1}^{M} \alpha_i y_i \langle x_i, x^i \rangle + b
\]

(5)

where \( x^i \in \mathbb{R}^N \) denotes the input testing sample and \( \langle x_i, x^i \rangle \) represents the inner product of the training samples and the input testing sample. The output of the classifier would be class 1 if \( d(x^i) > 0 \) and class 2 if \( d(x^i) < 0 \).

The non-linear SVM is introduced for the situations when the linear SVM is not suitable. Suppose the \( \phi(x) \) is the non-linear mapping function from the original \( x \) to higher dimensional feature space. As the decision function mainly depends on the inner product of the training samples and testing input, the Kernel function \( K(x_i, x^i) \) is applied to represent the inner product of the \( \phi(x_i) \) and \( \phi(x^i) \). The decision function for this situation can be updated as:

\[
d(x^i) = \sum_{i=1}^{M} \alpha_i y_i K(x_i, x^i) + b
\]

(6)

Any Kernel functions that meet the Mercer condition are valid kernels. So the kernel function can be found even the mapping function \( \phi(x) \) is not explicitly known. One of the popular kernel functions is the Gaussian Kernel \( K(x, y) = \exp(-\frac{||x-y||^2}{\sigma^2}) \).

The two-class SVM method can be extended to multi-class problems [5]. There are variant types of multi-class SVM to solve the classification problem with more than 2 classes, such as one-against-all support vector machines, pairwise support vector machines, Error correcting output code (ECOC) support vector machines, and all at once support vector machines, etc. Some papers make the comparison among these typical multi-class SVM [6]. Here this paper gives a short description for the basic one against all support machine and this method is also used in our experiment for simplicity.

Suppose the problem has \( K \) classes. The training set is composed of \( N \) samples, \( \{x_1, y_1\}, ..., \{x_K, y_K\} \), where \( x_i \in \mathbb{R}^N \) represents the \( N \) dimensional vector and \( y_i \in \{1, 2, ..., K\} \) is the output set of multi-class problem. The essential idea of one-against-all SVM is to build \( K \) binary SVM classifiers. Each classifier can be used to separate a class from all other classes in the output set. Based on the previous two-class SVM framework, if we get \( K \) decision functions as

\[
d_1(\cdot), d_2(\cdot), ..., d_K(\cdot),
\]

for the input test vector \( x^i \in \mathbb{R}^N \), it can be estimated as in class \( i \) while \( i = \arg \max_{i=1, ..., K} d_i(x^i) \).

III. DIMENSION REDUCTION

In large scale cognitive radio networks, there is a significant amount of data. However, in practice, the data is highly correlated. This redundancy in the data increases the overhead of cognitive radio networks for data transmission and data processing. In addition, the number of Degrees of Freedom (DoF) in large scale cognitive radio networks is limited. The DoF of a \( K \) user M x N MIMO interference channel has been discussed in [7]. The total number of DoF is equal to \( \min(K, N) \ast K \) if \( K \leq R \), and \( \min(K, N) \ast \frac{R}{K} \) if \( K > R \) (where \( R \) is the DoF per user). This is achieved based on interference alignment [8], [9], [10]. Theoretical analysis of DoF in cognitive radio has been presented in [11], [12]. The DoF corresponds to the key variables or key features in the network. Processing the high-dimensional data instead of the key variables will not enhance the performance of the network. In some cases, this could even degrade the performance. Hence, compact representations of the data using dimensionality reduction is critical in cognitive radio networks.

Dimensionality reduction [13], [14], [15], [16] finds a low-dimensional embedding of high-dimensional data. There are two categories of techniques to reduce the dimensionality of the input high-dimensional data. 1) Feature extraction [17] retains the original input data but makes a projection from the \( D \)-dimensional data to \( d \)-dimensional data where \( D < d \).
2) Feature selection [18] removes part of the input data that do not contribute significantly to the performance. It reduces the dimensionality by selecting a subset of original data. Three dimensionality reduction methods including both linear methods such as PCA [19], and nonlinear methods such as Kernel PCA (KPCA) [20], and Landmark Maximum Variance Unfolding (LMVU) [21], [22] are reviewed here. Actually they can be categorized as feature extraction techniques [23] [24].

If we assume the original high-dimensional data as a set of $M$ samples $x_i \in \mathbb{R}^N, i = 1, 2, \cdots, M$, then the reduced low-dimensional samples of $x_i$ are $y_i \in \mathbb{R}^K, i = 1, 2, \cdots, M$, where $K << N$. $x_{ij}$ and $y_{ij}$ are component wise elements in $x_i$ and $y_i$, respectively.

PCA [19] is the best-known linear dimensionality reduction method which performs a linear mapping of the high-dimensional data to a low-dimensional space such that the variance of the low-dimensional data is maximized. In reality, the covariance matrix of the data is constructed and the eigenvectors of this matrix are computed. The covariance matrix of $x_i$ can be obtained as,

$$
C = \frac{1}{M} \sum_{i=1}^{M} (x_i - u)(x_i - u)^T
$$

where $u = \frac{1}{M} \sum x_i$ is the mean of the given samples, and $T$ denotes the transpose operator.

The eigenvectors corresponding to the largest eigenvalues can be exploited to obtain a large portion of the variance of the original data. The original high-dimensional space can be reduced to a space spanned by a few dominant eigenvectors. PCA works well for the high-dimensional data with linear relationships, but always fails in a nonlinear scenario. PCA can be applied in the nonlinear situation by using a kernel [25], [26], [27], [28], called KPCA [20]. KPCA is therefore, a kernel-based machine learning algorithm. It uses the kernel function $k$ (which is the same as SVM) to implicitly map the original data to a feature space $F$, where PCA can be applied.

Other nonlinear techniques for dimensionality reduction include manifold learning techniques. Within the framework of manifold learning, the current trend is to learn the kernel using Semi-Definite Programming (SDP) [29], [30], [31], [32], [33] instead of defining a fixed kernel. The most prominent example of such a technique is MVU [21]. MVU can learn the inner product matrix of $y_i$ automatically by maximizing their variance, subject to the constraints that $y_i$ are centered, and local distances of $y_i$ are equal to the local distances of $x_i$.

Here, the local distances represent the distances between $y_i$ ($x_i$) and its $k$ nearest neighbors, in which $k$ is a parameter. The corresponding SDP can be cast into the following form [21],

$$
\text{maximize } \text{trace}(I) \\
\text{subject to } \\
I \succ 0 \\
\sum_{ij} I_{ij} = 0 \\
I_{ii} - 2I_{ij} + I_{jj} = D_{ij}, \text{ when } \eta_{ij} = 1
$$

where $I$ is an inner product matrix of $y_i$, $D_{ij} = \|x_i - x_j\|^2$, and $I \succ 0$ implies that $I$ is a Positive Semi-Definite (PSD) matrix.
IV. EXPERIMENTAL DEMONSTRATION

A. Experiment Setup and Procedures

The experiment complies with the concept of the cognitive radio network as wireless sensor network. One of the sensor nodes acts as radio transmitter, and all the other sensor nodes act as radio receivers. These nodes are implemented by the WARP platform from Rice University [34]. 6 receivers and 1 transmitter are deployed as a sensor network within an indoor environment as shown in Fig. 1 with the size of a normal office where the radio multi-path effect is significant. A pillar with irregular shape covered by metal surface is used to emulate the passive device free target. The locations of the transmitter and receivers are randomly arranged, as shown in Fig. 5. It means that the experiment scenario should be independent of the topology of the sensor network. But during the experiment, the topology of the network should be kept static.

All the WARP platforms are connected to a high speed Ethernet switch. A host PC running MATLAB is also connected to this switch to control all these WARP platforms. The baseband waveform is generated within MATLAB and downloaded to the memory of the transmitter. After up-conversion, this waveform is transmitted over air. All the received waveform at the receivers is transferred back to host PC for processing. Because the sensor nodes which take the role of secondary users can only use the bandwidth that is not occupied by the primary users, here the Non-Contiguous orthogonal frequency-division multiplexing (NC-OFDM) signal is applied as the waveform for the purpose of detection and localization. The application of the NC-OFDM signal can be consistent with the dynamic spectrum access attribute of the cognitive radio, and also emulates the situation in which part of the spectrum are interfered or faded in some complex environment. In each experiment, 5 NC-OFDM symbols are sent with fixed interval between symbols within a fixed length of time frame. The synchronization at Ethernet control level enables that all receivers work together within that time frame and ensure that these receivers receive all the NC-OFDM symbols to calculate the energy. Fig. 3 shows the received time series of the signal at one of the receivers. All the NC-OFDM symbols use the 40 sub-carriers out of the 64 sub-carriers within the whole 10MHz bandwidth. The rest of the sub-carriers are suppressed. Fig. 4 is an example of the spectrum for the received NC-OFDM signal at one of the receivers.

Our experiment mainly tries to find the available features from the energy at each sub-carrier of the received NC-OFDM
signal. The data processing flow is shown as Fig. 2. At each receiver, we get a energy vector with dimension of 40, denoted as \( \mathbf{x}^{(i)} = \left( x_{1}^{(i)}, x_{2}^{(i)}, \ldots, x_{40}^{(i)} \right) \in \mathbb{R}^{40} \), while the energy at certain sub-carrier is represented by \( x_{j}^{(i)} \in \mathbb{R}^{40}, j \in \{1, 2, \ldots, 40\} \). For 6 receivers, we get a energy matrix with size of \( 6 \times 40 \) from the wireless sensor network, denoted as \( \mathbf{X} = \begin{pmatrix} \mathbf{x}^{(1)} \\ \mathbf{x}^{(2)} \\ \vdots \\ \mathbf{x}^{(6)} \end{pmatrix} \). And this matrix can be converted to a high dimensional vector with dimension of 240 denoted as \( \mathbf{x}_{l} = \left( x_{1}^{(1)}, x_{2}^{(2)}, \ldots, x_{6}^{(6)} \right) \in \mathbb{R}^{240} \). If we directly feed this high dimensional vector to the multi-class SVM, the computational requirement could be very high, especially if we increase the scale of the sensor network in future. So before applying the SVM, the feature selection/extraction operation for dimensional reduction can be inserted.

Currently in our experiment, for simplicity we use an intuitive way to reduce the dimension of the network data, instead of the methods reviewed in section III. In the multi-path environment, the target within the radio network usually brings the different impact for the lower, middle and higher part of the communication bandwidth. So the \( \mathbf{x}^{(i)} \) can be turned into a new vector \( \tilde{\mathbf{x}}^{(i)} = \left( \tilde{x}_{1}^{(i)}, \tilde{x}_{2}^{(i)}, \tilde{x}_{3}^{(i)} \right) \) by adding the energy at lower, middle and higher sub-carriers respectively, which means:

\[
\tilde{x}_{1}^{(i)} = \sum_{j=1}^{13} x_{j}^{(i)} \\
\tilde{x}_{2}^{(i)} = \sum_{j=14}^{26} x_{j}^{(i)} \\
\tilde{x}_{3}^{(i)} = \sum_{j=27}^{40} x_{j}^{(i)}
\]

Thus the original high dimensional energy vector \( \mathbf{x}_{l} \) can be transformed to a new low dimensional energy vector \( \tilde{\mathbf{x}}_{l} = \left( \tilde{x}_{1}^{(1)}, \tilde{x}_{2}^{(2)}, \ldots, \tilde{x}_{3}^{(6)} \right) \in \mathbb{R}^{18} \). This method actually can be regarded as kind of feature extraction. Then the dimension of the network data is reduced from 240 to 18 which is suitable to be fed into multi-class SVM.

The one-against-all SVM is used in this experiment. The MATLAB code of SVM and Kernel Methods Matlab Toolbox is from [35].

B. Experiment Result and Analysis

The experiment is composed of two stages - training stage and testing stage. At the training stage, we randomly select 3 locations as shown in the Fig. 5. 30 training vectors are captured for each location where the target is placed. So in total we have 90 training samples for all three locations:

\[
\text{location}1 : \{ \mathbf{x}_{11}, y_{1} \}, \{ \mathbf{x}_{12}, y_{1} \}, \ldots, \{ \mathbf{x}_{30}, y_{1} \} \\
\text{location}2 : \{ \mathbf{x}_{11}, y_{2} \}, \{ \mathbf{x}_{12}, y_{2} \}, \ldots, \{ \mathbf{x}_{30}, y_{2} \} \\
\text{location}3 : \{ \mathbf{x}_{11}, y_{3} \}, \{ \mathbf{x}_{12}, y_{3} \}, \ldots, \{ \mathbf{x}_{30}, y_{3} \} \\
\] (10)

where \( y_{i} = i \) is corresponding to the certain location.

After dimension reduction, these training samples are fed into the multi-class SVM toolbox code to determine the multi-class classifier. Then at the testing stage, 5 testing samples are captured for each different location. The target location is estimated through the classification by the multi-class classifier. From the Fig. 5, it can be seen that if the target is placed at location 1 and 2, all the tests of the localization successfully get the correct output. For location 3, three of five tests succeed. This result demonstrates availability for the passive target detection and localization based on the machine learning method like multi-class SVM, especially in the multi-path environment. At location 3, the error rate of the prediction is acceptable at least more than half of the tests get the true location. Intuitively, the location 3 is at the edge of the network coverage. The target at this location may bring less meaningful features than locations closer to the center of the network coverage.

V. CONCLUSIONS AND FUTURE WORK

This paper is focused in demonstrating the application of the machine learning method in the passive target detection and localization within the complex multi-path environment where the large scale CRN based WSN is deployed. The localization is achieved through the classification utilizing the multi-class SVM. To handle the high dimensional network data, the dimensionality reduction is also applied. The experiment result discloses the big potential of the machine learning method in such application.

On the other hand, in the current experiment only 6 receivers are deployed and the number of training samples are not very large. With increasing the scale of the sensor network, and capturing more training samples, the localization accuracy is expected to be improved. Advanced method of dimensionality reduction can also be applied to handle higher dimensional network data. Meanwhile, the future experiment will perform classification regarding more locations. Ideally, with some acceptable resolution, the target at every possible location within the detecting area can be localized by machine learning methods.

ACKNOWLEDGMENTS

This work is funded by National Science Foundation through two grants (ECCS-0901420 and ECCS-0821658), and Office of Naval Research through two grants (N00010-10-1-0810 and N00014-11-1-0006).

REFERENCES


