Deep CNN for Spectrum Sensing in Cognitive Radio

Chang Liu*, *Member, IEEE*, Xuemeng Liu§, and Ying-Chang Liang*, *Fellow, IEEE*
*University of Electronic Science and Technology of China (UESTC), Chengdu, P. R. China
§Dalian University of Technology, Dalian, P. R. China
Email: changliu.wcom@gmail.com, cranberry_meng@mail.dlut.edu.cn, and liangyc@ieee.org

***Abstract*—The existing spectrum sensing methods mostly make decisions using model-driven test statistics, such as energy and eigenvalues. A weakness of these model-driven methods is the difficulty in accurately modeling for practical environment. In contrast to the model-driven approach, in this paper, we use a deep neural network to automatically learn features from data itself, and develop a data-driven detection approach. Inspired by the powerful capability of convolutional neural network (CNN) in extracting features of matrix-shaped data, we use the sample covariance matrix as the input of CNN, proposing a novel covariance matrix-aware CNN-based detection scheme, which consists of offline training and online detection. Different from the existing deep learning-based detection methods which replace the whole detection system by an end-to-end neural network, in this work, we use CNN for offline test statistic design and develop a practical threshold-based online detection mechanism. Specially, according to the maximum a posteriori probability (MAP) criterion, we derive the cost function for offline training in the spectrum sensing model, which guarantees the optimality of the designed test statistic. Simulation results have shown that whether the PU signals are independent or correlated, the detection performance of the proposed method is close to the optimal bound of estimator-correlator detector. Particularly, when the PU signals are correlated with a correlation coefficient 0.7, the probability of detection of the proposed method outperforms the conventional maximum eigenvalue detection method by nearly 7.5 times at SNR = -14dB.**

***Index Terms*—Cognitive radio, spectrum sensing, deep learning, convolutional neural network, covariance matrix-based detection.**

I. Introduction

Cognitive radio (CR), which enables the secondary users (SUs) to opportunistically access the licensed spectrum bands of primary users (PUs), is a promising technology for remitting the problem of spectrum crisis [1], [2]. To achieve this, the SUs need to frequently perform spectrum sensing, that is, detecting the activity states of PUs [3], [4]. Therefore, spectrum sensing plays an important role in CR system and the studies of spectrum sensing have attracted tremendous attention from both the academia and industry in the past decade [5].

Many spectrum sensing schemes have been proposed [6], and they are mostly based on model-driven test statistics which are derived from statistical models. Among these model-driven test statistics, the sample covariance matrix is a common choose for it contains various discriminative properties. Specifically, if the accurate statistical covariance matrices of the signal and noises are available, the estimator-correlator (E-C) detector [7] can be designed to achieve the optimal detection performance. Since the knowledge of signal is not always available in practice, thus, semi-blind detection methods, which need the noise knowledge only, have been proposed. Among these semi-blind methods, the energy detection (ED) [8] and the maximum eigenvalue detection (MED) [9], [10] are two classical algorithms for spectrum sensing. However, due to the noise uncertainty problem, the noise power level could be changing over time, thus we cannot get the noise power accurately [11]. In this case, the performance of the semi-blind methods degrades severely. To overcome this problem, the totally-blind detection methods have been proposed, which do not need any priori knowledge of the signal and noises. For example, the blindly combined energy detection (BCED) [12] achieves good detection performance by using the ratio of the maximum eigenvalue to the trace. Besides, the covariance absolute value (CAV) detection [13] uses the absolute values of the non-diagonal elements of the sample covariance matrix for detection, and it further improves detection performance. In summary, the model-driven methods can perform well through capturing the properties of assumed statistical models.

However, a weakness of these model-driven methods is the difficulty in accurately modeling for practical environment. In contrast to studying the model-driven methods, the deep learning (DL) technology uses a deep neural network to automatically learn data-driven features from the raw data [14]. To further improve the detection performance, the DL-based spectrum sensing methods have been studied. In [15], the authors use the statistics of energy and cyclostationary as the input of the artificial neural network (ANN), and propose an ANN-based spectrum sensing method. This method, however, needs the priori knowledge of PU signals, which is not always available in CR scenarios. To make detection method more practical, the work of [16] feeds ANN with the statistics of likelihood ratio and energy. Although these DL-based methods can further improve detection performance, the ANN only learns features from the limited test statistics, and there is still much room left for performance improvement. Note that covariance matrix is a versatile test statistic which includes various discriminative features, and the convolutional neural network (CNN) has powerful capability in extracting features of matrix-shaped data [17]. In this paper, we propose a covariance matrix-aware CNN-based detection scheme for spectrum sensing, and the main contributions of this work are summarized in the following.

(1) In contrast to the conventional model-driven spectrum sensing methods, we use the sample covariance matrix as the input of CNN to generate the test statistic, and

978-1-5386-8088-9/19/$31.00 ©2019 IEEE

propose a data-driven detection approach which consists of offline training and online detection. Taking advantage of the powerful capability of CNN in extracting features of matrix-shaped data, the proposed scheme is able to automatically learn more discriminative features for detection performance improvement.

(2) Different from the existing DL-based detection methods which replace the whole detection system by an end-to-end neural network, we use CNN for offline test statistic design and compare the well-trained test statistic with a threshold for online decision.

(3) According to the maximum a posteriori probability (MAP) criterion, we derive the cost function for offline training in the spectrum sensing model, which theoretically proves the optimality of the designed test statistic.

The paper is organized as follows. Section II formulates the system model of spectrum sensing. Based on the sensing scenario, the CNN-based detection scheme is proposed and analyzed in Section III, and simulation results are presented to verify the efficiency of the proposed method in Section IV. Finally, we conclude the work of the paper in Section V.

II. System Model

In the system model, a classical multi-antenna cognitive radio scenario is considered, where a PU is surrounded by some randomly distributed SUs equipped with multi-antenna array. If the PU is active, the broadcasting signal samples from the PU will be collected by the SUs; otherwise, only noise samples are received. Based on the collected samples, the SUs perform spectrum sensing to infer if the PU is active or not.

We assume that there are N observation vectors received by the SU with an M-element antenna array, and each observation vector is denoted by $\mathbf{x}(n) = [x_1(n), x_2(k), \cdots, x_M(n)]^T, n = 0, 1, \cdots, N-1$, where $x_i(n), i = 1, 2, \cdots, M$ denotes the n-th sample at the i-th antenna, and superscript T stands for transpose. In this case, the spectrum sensing problem at the SU terminal can be formulated as a binary hypothesis testing problem:

$$\begin{aligned} &H_0 : \mathbf{x}(n) = \mathbf{u}(n), \\ &H_1 : \mathbf{x}(n) = \mathbf{s}(n) + \mathbf{u}(n). \end{aligned} \tag{1}$$

Here H_0 and H_1 represent the hypothesis that the PU is absent and present, respectively. The term $\mathbf{s}(n)$ denotes the received signal vector, which suffers from channel fading and path loss. Note that the priori knowledge of PU is not available in practice, it is reasonable to formulate $\mathbf{s}(n)$ as a Gaussian random vector with zero mean and covariance matrix $\mathbf{R_s} = E(\mathbf{s}(n)\mathbf{s}^H(n))$, where $E(\cdot)$ represents the statistical expectation, and superscript H denotes conjugate transpose. In addition, $\mathbf{u}(n)$ is assumed to be an independent and identically distributed (*i.i.d.*) Gaussian vector with zero mean and covariance matrix $\mathbf{R_u} = E(\mathbf{u}(n)\mathbf{u}^H(n)) = \sigma_u^2\mathbf{I}_M$, where $\mathbf{I}_M$ indicates the identity matrix with order M, and σ_u^2 denotes the noise variance.

Based on the system model, we can then get the expressions of covariance matrix of the observation vector under two

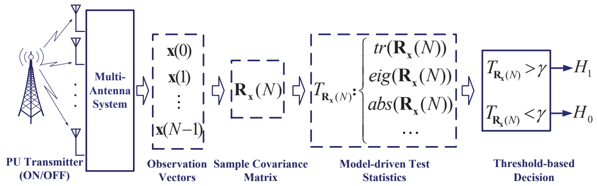

Fig. 1. Covariance matrix-based detection framework for spectrum sensing.

hypotheses:

$$\mathbf{R_x} = E(\mathbf{x}(n)\mathbf{x}^H(n)) = \begin{cases} \sigma_u^2\mathbf{I}_M, & H_0 \\ \mathbf{R_s} + \sigma_u^2\mathbf{I}_M, & H_1 \end{cases}. \tag{2}$$

It can be observed that the values of the elements of $\mathbf{R_x}$ under H_1 are dramatically different from that under H_0 due to the existence of the PU. Hence, the covariance matrix can be used to design test statistics for detection. For example, the optimal E-C detector can be expressed by [7]

$$T_{E-C} = \sum_{n=0}^{N-1} \mathbf{x}^H(n)\mathbf{R_s}(\mathbf{R_s} + \sigma_u^2\mathbf{I}_M)^{-1}\mathbf{x}(n) \tag{3}$$

where $(\cdot)^{-1}$ denotes the matrix inverse.

For further understanding, Fig. 1 presents the detection framework of the covariance matrix-based detection. Considering that only finite number of samples is available in practice, we replace the statistical covariance matrix by the sample covariance matrix, which is defined as

$$\mathbf{R_x}(N) = \frac{1}{N}\sum_{n=0}^{N-1} \mathbf{x}(n)\mathbf{x}^H(n), \tag{4}$$

where $\mathbf{R_x}(N)$ denotes the sample covariance matrix of N observation vectors. As shown in Fig. 1, the sample covariance matrix has many discriminative properties for detection, including the trace of the sample covariance matrix (denoted by $tr(\mathbf{R_x}(N))$) [8], the eigenvalues of the sample covariance matrix (denoted by $eig(\mathbf{R_x}(N))$) [9], [12], and the absolute values of elements of the sample covariance matrix (denoted by $abs(\mathbf{R_x}(N))$) [13]. During the detection framework, the SU terminal first obtains the sample covariance matrix from the observation vectors. Based on the sample covariance matrix, we can then calculate the test statistic $T_{\mathbf{R_x}(N)}$. Finally, we can make a decision by comparing $T_{\mathbf{R_x}(N)}$ with a threshold, which is preset to satisfy a desired probability of false alarm (PFA).

Based on the analysis above, the test statistic is of great importance for detection. In the following, we will use CNN to design a data-driven test statistic to further improve the detection performance.

III. Covariance Matrix-Aware CNN-based Detection Algorithm

Since CNN has powerful capability in extracting the features of matrix-shape data, we use CNN to extract the features of covariance matrix to design the test statistic. As shown in Fig. 2, we propose a covariance matrix-aware CNN-based detection scheme, which consists of offline training and online detection.

- Paper CNN 관련해서 structure 부분에서 convolution 연산을 적용하는 environment 관련해서 2부분 비교를 정리.

  - Paper #1

    - input signal

$$x(n) \xrightarrow{\text{FFT sampling}} \text{as} \quad \mathbb{X}_m = \left[x(mN_0), \cdots, x(mN_0 + N_{FFT} - 1) \right]$$

Transpose for $\hat{\mathbb{X}}_m$ set

$$\mathbb{X}_c = \begin{bmatrix} \hat{\tilde{X}}_c \\ \tilde{X}_{N_{FFT}/K_c} \end{bmatrix}$$

- estimation of delivered signal

$$\hat{P}_c = \frac{1}{K_c B} \sum_{n=0}^{B-1} \sum_{m=0}^{K_c-1} \left| \hat{\tilde{\mathbb{X}}}_c(m,n) \right|^2$$

- sampling된 noise channel 에 대해서도 동일하게 연산

- observation vector for multiple antennas

$$\begin{pmatrix} \mathbb{X}(0) \\ \mathbb{X}(1) \\ \vdots \\ \mathbb{X}(N-1) \end{pmatrix} \xrightarrow[\text{matrix}]{\text{sample - covariance}} \mathbb{R}_x(N)$$

observation vector

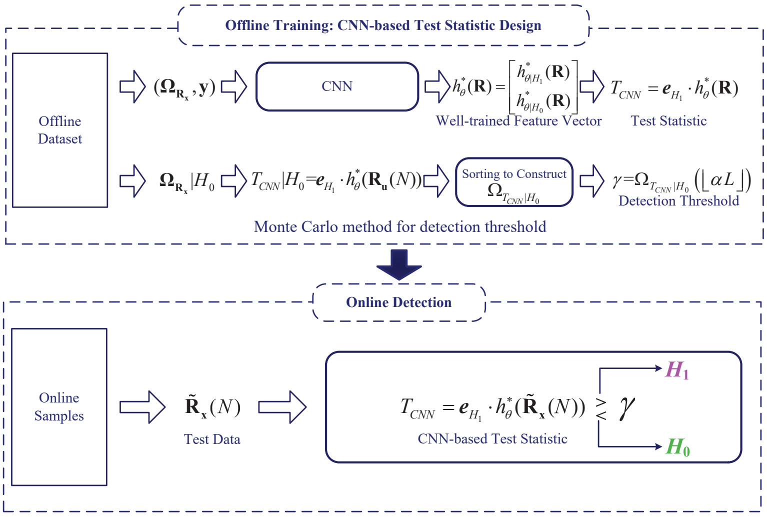

Fig. 2. Proposed covariance matrix-aware CNN-based detection scheme.

A. Offline Training: CNN-based Test Statistic Design

For offline training module, the labeled samples should be collected to build the offline dataset, which is denoted by:

$$(\mathbf{\Omega}_{\mathbf{R}_\mathbf{x}}, \mathbf{y}) = \{(\mathbf{R}_\mathbf{x}^{(1)}(N), \mathbf{y}^{(1)}), (\mathbf{R}_\mathbf{x}^{(2)}(N), \mathbf{y}^{(2)}), \\ \cdots, (\mathbf{R}_\mathbf{x}^{(K)}(N), \mathbf{y}^{(K)})\}, \tag{5}$$

where $\mathbf{\Omega}_{\mathbf{R}_\mathbf{x}}$ and $\mathbf{y}$ indicate the sample set and label set, respectively, and $(\mathbf{R}_\mathbf{x}^{(k)}(N), y^{(k)})$ represents the k-th ($k = 1, 2, \cdots, K$) labeled sample of offline dataset. In this paper, the hypotheses of H_1 and H_0 are denoted by $y^{(k)} = 1$ and $y^{(k)} = 0$, respectively.

Based on the offline dataset, we will use a CNN to design the test statistic. As shown in Fig. 2, taking the k-th labeled sample $(\mathbf{R}_\mathbf{x}^{(k)}(N), y^{(k)})$ as an example, we use the sample covariance matrix $\mathbf{R}_\mathbf{x}^{(k)}(N)$ as the input of CNN. After the non-linear transformation of the hidden layers, we finally use a softmax function to obtain the output of CNN: a normalized feature vector, which is defined as

$$h_\theta(\mathbf{R}_\mathbf{x}^{(k)}(N)) = \begin{bmatrix} h_{\theta|H_1}(\mathbf{R}_\mathbf{x}^{(k)}(N)) \\ h_{\theta|H_0}(\mathbf{R}_\mathbf{x}^{(k)}(N)) \end{bmatrix}, \tag{6}$$

where $h_\theta(\cdot)$ denotes the whole non-linear expression of CNN with model parameters θ, and $h_{\theta|H_i}(\cdot)$ represents the non-linear expression under hypothesis H_i. In this case, $h_{\theta|H_i}(\mathbf{R}_\mathbf{x}^{(k)}(N))$ denotes the extracted feature value of hypothesis H_i. According to the definition of softmax regression and spectrum sensing model, we have the conditional probability expressions under two hypotheses:

$$H_1 : P(y^{(k)} = 1|\mathbf{R}_\mathbf{x}^{(k)}(N); \theta) = h_{\theta|H_1}(\mathbf{R}_\mathbf{x}^{(k)}(N)), \\ H_0 : P(y^{(k)} = 0|\mathbf{R}_\mathbf{x}^{(k)}(N); \theta) = h_{\theta|H_0}(\mathbf{R}_\mathbf{x}^{(k)}(N)), \tag{7}$$

with

$$h_{\theta|H_1}(\mathbf{R}_\mathbf{x}^{(k)}(N)) + h_{\theta|H_0}(\mathbf{R}_\mathbf{x}^{(k)}(N)) = 1, \tag{8}$$

where $P(\cdot)$ denotes the probability. In this case, the goal of the offline training is to maximize the likelihood:

$$\begin{aligned} L(\theta) &= P(\mathbf{y}|\mathbf{\Omega}_\mathbf{R}; \theta) \\ &= \prod_{k=1}^{K} P(y^{(k)}|\mathbf{R}_\mathbf{x}^{(k)}(N); \theta) \\ &= \prod_{k=1}^{K} \left(h_{\theta|H_1}(\mathbf{R}_\mathbf{x}^{(k)}(N))\right)^{y^{(k)}} \left(h_{\theta|H_0}(\mathbf{R}_\mathbf{x}^{(k)}(N))\right)^{1-y^{(k)}}, \end{aligned} \tag{9}$$

or it can be rewritten as a form of log-likelihood:

$$\begin{aligned} l(\theta) &= \log L(\theta) \\ &= \sum_{k=1}^{K} y^{(k)} \log h_{\theta|H_1}(\mathbf{R}_\mathbf{x}^{(k)}(N)) \\ &\quad + (1 - y^{(k)}) \log(h_{\theta|H_0}(\mathbf{R}_\mathbf{x}^{(k)}(N))). \end{aligned} \tag{10}$$

Based on this, we can then define the cost function of training:

$$\begin{aligned} J(\theta) &= -\tfrac{1}{K} l(\theta) \\ &= -\tfrac{1}{K} \sum_{k=1}^{K} y^{(k)} \log h_{\theta|H_1}(\mathbf{R}_\mathbf{x}^{(k)}(N)) \\ &\quad + (1 - y^{(k)}) \log(h_{\theta|H_0}(\mathbf{R}_\mathbf{x}^{(k)}(N))). \end{aligned} \tag{11}$$

Through minimizing the cost function (11), the goal of CNN is to obtain the optimal θ to maximize $P(\mathbf{y}|\mathbf{\Omega}_\mathbf{R}; \theta)$, that is,

$$\theta^* = \arg \max_\theta P(\mathbf{y}|\mathbf{\Omega}_\mathbf{R}; \theta), \tag{12}$$

where θ^* denotes the optimal θ. Hence, if the size of the training dataset is large enough, we can obtain the optimal $h_\theta(\cdot)$ under the maximum a posteriori probability (MAP) criterion.

Based on the analysis above, we then adopt a 7-layer CNN for offline training. The architecture of the CNN model is shown in Fig. 3, which consists of an input layer (S_0), two convolutional layers (C_1 and C_2), two pooling layers (S_1 and S_2), and two fully connected layers (F_1 and F_2). The sample covariance matrix is first sent to the input layer S_0, and after the non-linear operation of the hidden layers, we can obtain the feature vector at the layer of F_2, as defined in (6). The

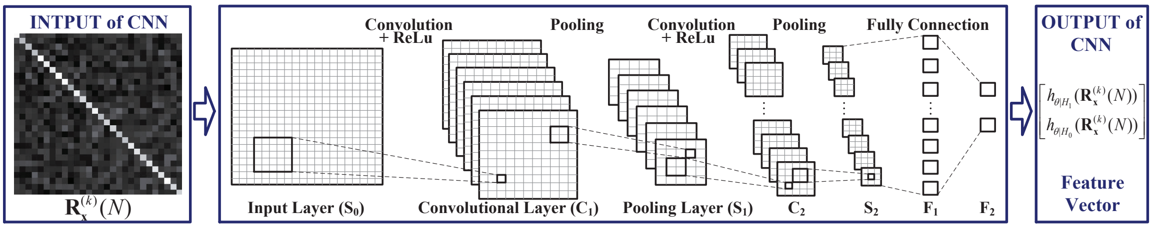

Fig. 3. CNN structure for test statistic design.

TABLE I
HYPERPARAMETERS OF THE PROPOSED CM-CNN METHOD

Input: Sample Covariance Matrix (Dimension: 28×28)	
CNN Layers	**Kernel Size**
S_0	Null
C_1 + ReLU	$20@(5 \times 5)$
S_1 (Max-Pooling)	2×2
C_2 + ReLU	$50@(5 \times 5)$
S_2 (Max-Pooling)	2×2
F_1	500×2450
F_2	2×500
Output: Feature Vector (Dimension: 2×1)	

detailed hyperparameters of the CNN used in our paper are summarized in Table I, where "ReLu" denotes the rectified linear units, and "Max-Pooling" indicates the sample-based discretization process by computing the maximum value of a particular feature over a region of the prior layer. Through the backpropagation algorithm, we can progressively update the model parameters, and finally obtain the well-trained CNN. Thus, the well-trained feature vector (as shown in Fig. 2) can be expressed as

$$h_\theta^*(\mathbf{R}) = \begin{bmatrix} h_{\theta|H_1}^*(\mathbf{R}) \\ h_{\theta|H_0}^*(\mathbf{R}) \end{bmatrix}, \tag{13}$$

where $h_\theta^*(\cdot)$ denotes the whole non-linear expression of the well-trained CNN, $h_{\theta|H_i}^*(\cdot)$ represents the non-linear expression of the well-trained CNN for hypothesis H_i, and $\mathbf{R}$ denotes the input matrix. Based on this, we can obtain the test statistic under an arbitrary hypothesis:

$$T_{H_i} = \boldsymbol{e}_{H_i} \cdot h_\theta^*(\mathbf{R}), \tag{14}$$

with the selection vector

$$\boldsymbol{e}_{H_i} = \begin{cases} [1, 0], & i = 1 \\ [0, 1], & i = 0 \end{cases}, \tag{15}$$

where T_{H_i} indicates the test statistic under hypothesis H_i.

Generally, we can use either T_{H_1} or T_{H_0} for detection. In this paper, we use T_{H_1} as the test statistic, and the CNN-based test statistic can be written as

$$T_{CNN} = \boldsymbol{e}_{H_1} \cdot h_\theta^*(\mathbf{R}). \tag{16}$$

By comparing T_{CNN} with a detection threshold γ, we can finally make a decision. Thus, the detection threshold γ needs to be predetermined before online detection. Note that the PFA is defined by

$$P_{fa} = P\{T_{CNN}|H_0 > \gamma\} \tag{17}$$

where

$$T_{CNN}|H_0 = \boldsymbol{e}_{H_1} \cdot h_\theta^*(\mathbf{R}_\mathbf{u}(N)) \tag{18}$$

denotes T_{CNN} under hypothesis H_0, that is, the T_{CNN} expression in (16) with

$$\mathbf{R} = \mathbf{R}_\mathbf{u}(N) = \frac{1}{N} \sum_{n=0}^{N-1} \mathbf{u}(n)\mathbf{u}^H(n), \tag{19}$$

where $\mathbf{R}_\mathbf{u}(N)$ is the noise sample covariance matrix.

Therefore, according to the definition of PFA, we can then use Monte Carlo method to set the detection thresholds for different probabilities of false alarm. As shown in Fig. 2, given the offline dataset under H_0: $\mathbf{\Omega}_{\mathbf{R_x}}|H_0 = \{\mathbf{R}_\mathbf{u}^{(1)}(N), \mathbf{R}_\mathbf{u}^{(2)}(N), \cdots, \mathbf{R}_\mathbf{u}^{(L)}(N)\}$, where L is the size of dataset, we can then obtain the values of $T_{CNN}|H_0$. By sorting these values in a descending order, we construct a set of $T_{CNN}|H_0$, denoted by $\mathbf{\Omega}_{T_{CNN}}|H_0$. Thus, the detection threshold with a desired PFA value α can be expressed by

$$\gamma = \mathbf{\Omega}_{T_{CNN}|H_0}(\lfloor \alpha L \rfloor), \tag{20}$$

where $\lfloor \cdot \rfloor$ denotes the round down to the nearest integer, and $\mathbf{\Omega}_{T_{CNN}|H_0}(l)$ indicates the l-th element of $\mathbf{\Omega}_{T_{CNN}|H_0}$.

B. Online Detection

When we perform online detection module, the multi-antenna system should collect N (unlabeled) online samples, as illustrated in Fig. 2. Based on these online samples, we first calculate the sample covariance matrix, and then send it to the well-trained CNN to obtain the test statistic T_{CNN}, as defined in (16). Finally, we can make a decision by comparing T_{CNN} with a detection threshold γ.

Note that the existing DL-based methods directly replace the whole detection system by an end-to-end neural network, which is not available to set a desired PFA. In contrast to them, we develop a more practical threshold-based detection mechanism, where the threshold is predetermined in offline training, and thus it is convenient to set a desired PFA by selecting the corresponding threshold.

C. Proposed Covariance Matrix-Aware CNN-based Detection Algorithm

According to the analysis above, we propose a novel covariance matrix-aware CNN-based detection algorithm, which is summarized in algorithm 1.

Algorithm 1: Covariance Matrix-Aware CNN-based (CM-CNN) Spectrum Sensing Algorithm.

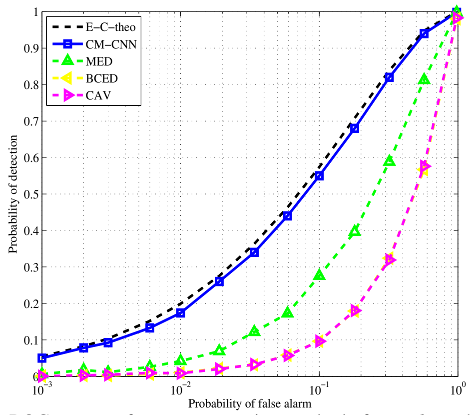

Fig. 4. ROC curves of spectrum sensing methods for *i.i.d.* model: M=28, N=100, SNR = -14dB.

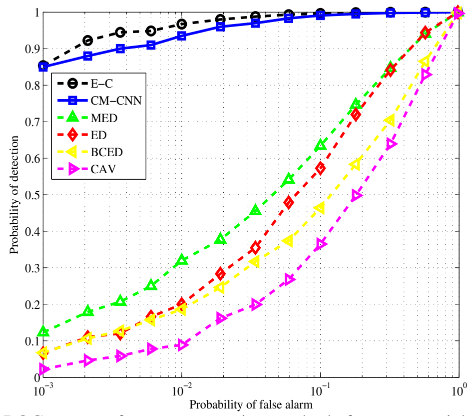

Fig. 5. ROC curves of spectrum sensing methods for exponential correlation model with ρ=0.7 : M=28, N=100, SNR = -14dB.

Step 1: **Data preprocessing.** Divide all the labeled samples into K groups: $\{(\mathbf{X}^{(1)}, y^{(1)}), (\mathbf{X}^{(2)}, y^{(2)}), \cdots, (\mathbf{X}^{(K)}, y^{(K)})\}$, and each of them is expressed by

$$\begin{aligned}\mathbf{X}^{(k)} &= \mathbf{x}(n)_{n=(k-1)N}^{kN-1} \\ &= \{\mathbf{x}((k-1)N), \mathbf{x}((k-1)N+1), \cdots, \mathbf{x}(kN-1)\},\end{aligned} \tag{21}$$

where $k = 1, 2, \cdots, K$, $N = N_s/K$ is the size of each group, and N_s is the total number of temporal samples for training.

Step 2: **Compute the sample covariance matrix of each group.** The k-th ($1 \le k \le K$) sample covariance matrix can be calculated by:

$$\mathbf{R}_{\mathbf{x}}^{(k)}(N) = \frac{1}{N}\mathbf{X}^{(k)}[\mathbf{X}^{(k)}]^H = \frac{1}{N}\sum_{n=(k-1)N}^{kN-1} \mathbf{x}(n)\mathbf{x}^H(n). \tag{22}$$

Step 3: **Offline training for the CNN-based test statistic.** Based on the offline training dataset $(\mathbf{\Omega_R}, \mathbf{y}) = \{(\mathbf{R}_{\mathbf{x}}^{(1)}(N), y^{(1)}), (\mathbf{R}_{\mathbf{x}}^{(2)}(N), y^{(2)}), \cdots, (\mathbf{R}_{\mathbf{x}}^{(K)}(N), y^{(K)})\}$, we operate training process with the cost function (11) through backpropagation algorithm, and finally obtain the CNN-based test statistic T_{CNN}, as defined in (16).

Step 4: **Search the detection threshold.** Given a desired PFA value α, we then find the detection threshold γ using Monte Carlo method, as described in (20).

Step 5: **Calculate the online test statistic.** Given N temporal samples for testing, we then calculate the sample covariance matrix, denoted by $\tilde{\mathbf{R}}_{\mathbf{x}}(N)$. Send $\tilde{\mathbf{R}}_{\mathbf{x}}(N)$ to the well-trained CNN and we can finally obtain the online test statistic:

$$T_{CNN} = \boldsymbol{e}_{H_1} \cdot h_\theta^*(\tilde{\mathbf{R}}_{\mathbf{x}}(N)). \tag{23}$$

Step 6: **Online decision.** If $T_{CNN} > \gamma$, signal exists (H_1 decision); otherwise, signal does not exist (H_0 decision).

IV. Numerical Results

This section provides some simulation results based on the multi-antenna CR system. In the simulations, we assume the PU signals are received by a SU which is equipped with a 28-element antenna array ($M = 28$ in formula (1)). Without loss of generality, we use two signal models to construct the received PU signal samples [7], [18]:

(a) *Independent and identically distributed (i.i.d.) model*: signal sample vector $\mathbf{s}(n)$ is formulated as an *i.i.d.* Gaussian random vector with zero mean and covariance matrix $\mathbf{R_s} = \mathbf{I}_M$, where $\mathbf{I}_M$ denotes the identity matrix of order M;

(b) *Exponential correlation model*: signal samples vector $\mathbf{s}(n)$ is formulated as an exponential correlated Gaussian random vector with zero mean and covariance matrix $\mathbf{R_s}$ with $(\mathbf{R_s})_{p,q} = \rho^{|p-q|}$, where $0 < \rho < 1$ is correlation coefficient and $(\cdot)_{p,q}$ denotes the p-th row, q-th column element.

The noise model is the same as the assumption in Section II. Each point in the simulation results is obtained by averaging 10000 Monte Carlo realizations, and each realization is operated based on the number of temporal samples $N = 100$. Since our proposed CM-CNN detection algorithm carries out the spectrum sensing by using the sample covariance matrix, we compare it with six existing covariance matrix-based methods, including the optimal E-C detection method, MED method, ED method, BCED method, and CAV detection method.

First, we evaluate the performance of these detection methods under *i.i.d.* model, as defined in model (a). In the simulation, we first set the detection thresholds for different values of PFA, and then use these thresholds to make decisions; finally, we can obtain the probability of detection (PD) values under different values of PFA based on Monte Carlo realizations. Note that it is easy to obtain the closed form theoretical PD expression of the optimal E-C detection method under *i.i.d.* model, and thus we plot it as the optimal theoretical bound, which is denoted by "E-C-theo". Based on the analysis above, Fig. 4 plots the Receiver Operating Characteristics (ROC) curves of different spectrum sensing algorithms to present the values of PD under different PFA values. It can be observed that these conventional methods, including the MAE, BCED, and CAV methods, present the validity of detection, while they still have large gaps with the optimal E-C detection method. Different from these model-driven detection methods, our proposed CM-CNN method achieves the best detection

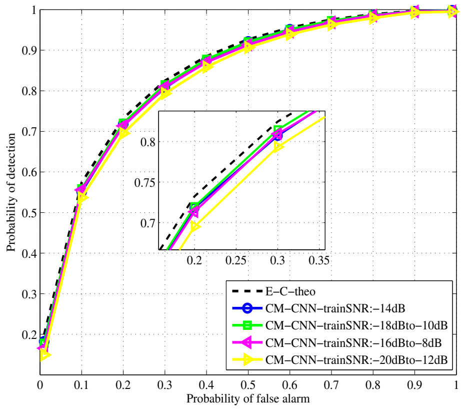

Fig. 6. ROC curves of the proposed CM-CNN method with different training SNR ranges: $M = 28$, $N = 100$, SNR = -14dB.

performance, which is very close to the optimal theoretical bound of E-C detection method. It is because the proposed method learns more discriminative features from the sample covariance matrix.

Next, we present the ROC curves under exponential correlation model (as defined in model (b)) in Fig. 5. Similar to the results under *i.i.d.* model, the proposed CM-CNN method presents outstanding detection performance, and the ROC curve of CM-CNN gets very close to the ROC curve of the optimal E-C detection method. In addition, when comparing Fig. 5 with Fig. 4, we can find that the detection probability under the exponential correlation model is dramatically higher than that under the *i.i.d.* model. For example, in Fig. 5, the PD of the proposed CM-CNN method exceeds the MAE detection method by 0.75 under the correlated model with $\rho = 0.7$. The reason is that, in correlated model, the difference between two sample covariance matrices under each hypothesis is larger than that in *i.i.d.* model. Therefore, the correlation of signals contributes to the detecting of PU signals from the noise.

Note that the test statistic of the proposed method depends on the SNRs of training set, we turn to evaluate the SNR-robustness of the proposed method. Through varying SNR ranges, we use samples under different SNRs to calculate the sample covariance matrices as training dataset, and plot the ROC curves at SNR=-14dB with $i.i.d.$ signal model in Fig. 6. It is shown that the CM-CNN methods with different training SNR ranges almost keep the same detection performance as the CM-CNN method with a single training SNR of -14dB. The present results highlight that the test statistic of the proposed method is robust to SNR as long as the SNR range of the offline training is large enough.

V. Conclusion

This paper proposes a smart spectrum sensing method using DL technology. In contrast to the conventional model-driven spectrum sensing methods, we use CNN to design a data-driven test statistic by exploring the distinguishable features from the sample covariance matrix, and propose a covariance matrix-aware CNN detection scheme which consists of offline training and online detection. The offline module is operated to train the optimal test statistic under MAP criterion, and the online module then uses the well-trained test statistic to make decisions. Since the CNN has powerful ability in extracting features of the matrix-shaped data, more discriminative features for detection can be exploited. Extensive simulation results have shown that the proposed CM-CNN detection method performs well whether the PU signals are independent or correlated, and its detection performance is nearly the same as that of the optimal E-C detection method.

VI. Acknowledgments

This work was supported in part by National Natural Science Foundation of China under Grants 61801082, 61631005, 61571100, and 61801101.

References

[1] J. Lunden, V. Koivunen, and H. V. Poor, "Spectrum exploration and exploitation for cognitive radio: Recent advances," *IEEE Signal Process. Mag.*, vol. 32, no. 3, pp. 123–140, Apr. 2015.

[2] Y. Ma, Y. Gao, Y.-C. Liang, and S. Cui, "Reliable and efficient sub-nyquist wideband spectrum sensing in cooperative cognitive radio networks," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 10, pp. 2750–2762, Oct. 2016.

[3] L. Zhang, M. Xiao, G. Wu, M. Alam, Y.-C. Liang, and S. Li, "A survey of advanced techniques for spectrum sharing in 5G networks," *IEEE Wireless Commun.*, vol. 24, no. 5, pp. 44–51, 2017.

[4] L. Zhang, Y.-C. Liang, and M. Xiao, "Spectrum sharing for internet of things: a survey," *IEEE Wireless Commun.*, 2018, DOI: 10.1109/MWC.2018.1800259.

[5] A. Ali and W. Hamouda, "Advances on spectrum sensing for cognitive radio networks: Theory and applications," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 2, pp. 1277–1304, 2nd Quart. 2017.

[6] Y.-C. Liang, K.-C. Chen, G. Y. Li, and P. Mähönen, "Cognitive radio networking and communications: An overview," *IEEE Trans. Veh. Tech.*, vol. 60, no. 7, pp. 3386–3407, Sept. 2011.

[7] S. M. Kay, *Fundamentals of statistical signal processing, volume II: Detection Theory*. Englewood Cliffs, NJ, USA, Prentice Hall, 1998.

[8] F. F. Digham, M.-S. Alouini, and M. K. Simon, "On the energy detection of unknown signals over fading channels," *IEEE Trans. Commun.*, vol. 55, no. 1, pp. 21–24, Jan. 2007.

[9] Y. Zeng, C. L. Koh, and Y.-C. Liang, "Maximum eigenvalue detection: Theory and application," in *Proc. IEEE Int. Conf. Commun. (ICC)*, pp. 4160–4164, Beijing, China, May 2008.

[10] C. Liu, H. Li, J. Wang, and M. Jin, "Optimal eigenvalue weighting detection for multi-antenna cognitive radio networks," *IEEE Trans. Wireless Commun.*, vol. 16, no. 4, pp. 2083–2096, Apr. 2017.

[11] R. Tandra and A. Sahai, "SNR walls for signal detection," *IEEE J. Sel. Topics Signal Proc.*, vol. 2, no. 1, pp. 4–17, Feb. 2008.

[12] Y. Zeng, Y.-C. Liang, and R. Zhang, "Blindly combined energy detection for spectrum sensing in cognitive radio," *IEEE Signal Process. Lett.*, vol. 15, pp. 649–652, Oct. 2008.

[13] Y. Zeng and Y.-C. Liang, "Spectrum-sensing algorithms for cognitive radio based on statistical covariances," *IEEE Trans. Veh. Tech.*, vol. 58, no. 4, pp. 1804–1815, May 2009.

[14] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, May 2015.

[15] Y.-J. Tang, Q.-Y. Zhang, and W. Lin, "Artificial neural network based spectrum sensing method for cognitive radio," in *Proc. IEEE Int. Conf. Wireless Commun. Netw. Mobile Comput. (WiCOM)*, Chengdu, China, Sept. 2010.

[16] M. R. Vyas, D. Patel, and M. Lopez-Benitez, "Artificial neural network based hybrid spectrum sensing scheme for cognitive radio," in *Proc. IEEE Int. Symp. Pers. Indoor Mobile Radio Commun. (PIMRC)*, Montreal, QC, Canada, Oct. 2017.

[17] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[18] V. A. Aalo, "Performance of maximal-ratio diversity systems in a correlated nakagami-fading environment," *IEEE Trans. Commun.*, vol. 43, no. 8, pp. 2360–2369, Aug. 1995.