# On Experimental Evaluation of Unsupervised Spectrum Sensing

Pastor David Chávez Muñoz, [1] , Julio Cesar Manco Vasquez[2], and César David Córdova Bernuy[3]

[1,2,3]*Pontificia Universidad Católica del Perú, Perú, dchavez@pucp.edu.pe, manco.jc@pucp.edu.pe, cdcordova@pucp.edu.pe*

Abstract – Spectrum sensing plays a key role in cognitive radio (CR) networks in order to determine the availability of unused frequency bands. So far, presumed models have been employed to conceive statistical tests such as eigenvalue-based detectors. Nevertheless, their detection performances are degraded as the accuracy of these models depart from real-world measurements. In this paper, we assess the performance of an unsupervised learning spectrum sensing (ULSS) detection through experimental evaluations. In this approach, model assumptions are no longer required, while avoiding labeled data often not available in practical CR scenarios. The ULSS consists of a two-stage training, where an unsupervised Gaussian mixture model (GMM) is employed to provide training data for a deep neural network (DNN). The experimental results shows that it outperforms model-based detectors by learning from real measurements.

Keywords – Experimental evaluation, spectrum sensing, unsupervised detection, software defined radio (SDR).

On Experimental Evaluation of Unsupervised Spectrum Sensing

Pastor David Chávez Muñoz, [1] , Julio Cesar Manco Vasquez[2] , and César David Córdova Bernuy[3]

[1,2,3] *Pontificia Universidad Católica del Perú, Perú, dchavez@pucp.edu.pe, manco.jc@pucp.edu.pe, cdcordova@pucp.edu.pe*

Abstract – Spectrum sensing plays a key role in cognitive radio (CR) networks in order to determine the availability of unused frequency bands. So far, presumed models have been employed to conceive statistical tests such as eigenvalue-based detectors. Nevertheless, their detection performances are degraded as the accuracy of these models depart from real-world measurements. In this paper, we assess the performance of an unsupervised learning spectrum sensing (ULSS) detection through experimental evaluations. In this approach, model assumptions are no longer required, while avoiding labeled data often not available in practical CR scenarios. The ULSS consists of a two-stage training, where an unsupervised Gaussian mixture model (GMM) is employed to provide training data for a deep neural network (DNN). The experimental results shows that it outperforms model-based detectors by learning from real measurements.

Keywords – Experimental evaluation, spectrum sensing, unsupervised detection, software defined radio (SDR).

I. Introduction

The increasing number of wireless devices requiring higher data rates has propitiated a higher demand of broadband wireless spectrum. This has limited the availability of spectral resources where spectrum allocation policy remains static. To this end, CR systems introduces dynamic spectrum allocation by exploiting the available frequency bands, also known as spectrum holes or white spaces [1], [2]. In these systems, licensed or primary users (PU) and unlicensed users or secondary users (SU) coexist over the same spectrum. In this regard, spectrum sensing plays a key role by detecting the spectrum occupancy of the PUs, so that detected holes are filled with transmissions from SUs [2].

Numerous spectrum sensing algorithms have been reported in the literature. Particularly, the vast majority of test statistics are obtained from the sample covariance matrix, where eigenvalues are probably the features most employed for these statistical tests. Unlike energy-based or matched filter detectors [3], they do not require prior knowledge about the PU (e.g., noise power) or synchronization with the sampled signals. Nevertheless, inaccuracies about signal or noise distribution, as well as assumed prior knowledge regarding the mathematical formulation of the underlying environment may lead to poor performance, when it does not represent the underlying CR scenario, for which several efforts have been carried out to validate these approaches [4], [5], [6].

Recently, DNN has been employed in numerous applications in the field of wireless communications, and naturally its success in tasks of classification has motivated its study for spectrum sensing. The reported supervised learning approaches requires of labeled data for training purposes to meet the required detection performance [7], [8], [9].

However, it is not suitable for practical CR scenarios, where the actual state of the PUs is not unknown, thus requiring a cooperation between PU and SUs. To address these issues, unsupervised learning techniques have been employed and reported in the literature [10], [11], [12]. These approaches do not require prior information, assumptions about the modeling of CR networks, or the availability of labeled data. Instead, there is need for validating these approaches in more real scenarios that allow its integration in existing and future wireless communication systems.

Software defined radio (SDR) platforms have shown to be an ideal vehicle to corroborate the feasibility of CR approaches [13], [14]. In this paper, an ULSS detection is assessed through experimental evaluations. This scheme employs the eigenvalues extracted from the sample covariance matrix of the received signal, and it is composed of two stages: a training stage and a real-time detection. During the training, a Gaussian mixture model (GMM) is used for the clustering of the collected eigenvalues in order to obtain a set of labeled data, after which it is utilized for training a DNN. Eventually, this DNN is used for real-time detection. Unlike [11], [10], eigenvalues features are considered for the detection and evaluation through experimental measurements. The preliminary results show that it obtains a close performance to that of an optimal Newman Pearson (NP) detector [15]. Moreover, it overcomes eigenvalue-based detectors commonly reported in the literature. Finally, it suggests the feasibility of unsupervised approaches motivating further research in this direction.

The rest of the paper is organized as follows: in Section II, the system model is presented. Next, model-based detectors and the examined unsupervised detection scheme are described in Section III and IV, respectively. The employed SDR platform is exposed in Section V, while the experimental results along with discussions are depicted in Section VI. Finally, the conclusions and future guidelines are presented in Section VII.

II. SYSTEM MODEL

Spectrum sensing can be formulated as a hypothesis test, where under a null hypothesis H0, it indicates the absence of the PU over a frequency channel, whereas under an alternative hypothesis H1, it points out that a PU is transmitting. Under these assumptions, the model of the received signal, $y[n] \in C$, at a SU can be expressed as follows:

$$H_0 : y[n] = w[n] \qquad (1)$$
$$H_1 : y[n] = h[n]s[n] + w[n] \qquad (2)$$

where n denotes the sampling instant, while h[n], s[n], and w[n] represent the channel gain, the transmitted signal, and the additive white Gaussian noise (AWGN), respectively. The decision between the H_0 and H_1 hypotheses, is often evaluated using the probabilities of detection PD and false alarm PF A. PD is the probability of deciding that a PU is present when it is true, i.e., p(H₁|H₁), while P_{FA} refers to the probability of reporting that a PU is present when it is indeed absent, p(H₁|H₀).

In the next sections, we describe the examined detectors. Two of them based on presumed models, i.e., the ratio of the arithmetic to geometric mean (AGM) of the eigenvalues [16], and an Hadamard detector [17], after which the ULSS detection is introduced.

III. MODEL-BASED DETECTORS

For a given SU, the receiver signal at the baseband is stacked depicted in Fig 1, where it can be observed that it is comprised of two main components during a training stage, i.e., GMM & expectation maximization (EM) algorithms, as well as a DNN. in an observation matrix. In doing so, the received samples are split into M vectors of length Ns, so that each of them are stacked in a M × N matrix as follows,

$$Y = \begin{bmatrix} y_1[1] \, y_1[2] \, y_1[3] & \cdots & y_1[N_s] \\ \vdots & \ddots & \vdots \\ y_M[1] \, y_M[2] \, y_M[3] & \cdots & y_M[N_s] \end{bmatrix} \qquad (3)$$

Then, a sample covariance matrix is computed as,

$$R = \frac{1}{N_s} Y Y^H \qquad (4)$$

It is worth mentioning that in the absence of a PU, i.e., under a H_0, the received samples are uncorrelated regardless the fading channel model, and the diagonal elements of the sample covariance matrix R tend to the noise variance, while the non-diagonal elements are approximately zero, as the number of samples Ns increases. In other words, $R \approx diag[\sigma_1^2, \ldots, \sigma_M^2]$, with σ_k^2 the noise variance of each row vector. On the other hand, the presence of a PU induces

some correlation and/or additional spatial structure in the R matrix.

It has motivated the formulation of several hypothesis testing problems based on the structure of a covariance matrix under different assumptions. Moreover, the likelihood depends on unknown parameters, and the most typical approach to solve this kind of testing problems is the Gaussian likelihood ratio test (GLRT) [15]. In this regard, under an unknown noise variance, if we consider it to be independent and identically distributed (iid), the sample noise covariance $R \approx I \sigma^2$, and consequently an AGM detector can be obtained as follows,

$$T_{AGM} = \frac{\frac{1}{M} \sum_{k=1}^{M} \lambda_k}{\left(\prod_{k=1}^{M} \lambda_k \right)^{\frac{1}{M}}} \qquad (5)$$

where λ_k refers to the k-th eigenvalue of R. However, note that this case assumes the same noise variance σ^2 along the diagonal of R, which is hard to achieve with real measurements. A more generic diagonal noise covariance matrix under H_0 can take into account the uncertainties in the noise variance for most practical cases, e.g., under non-iid noise. Under this assumption, a Hadamard ratio can be derived from the GLRT which is given by,

$$T_{hadamard} = \frac{|R|}{\prod_{k=1}^{M} [R]_{k,k}} \qquad (6)$$

where $|.|$ refers to the determinant and $[R]_{k,k}$ to the (k, k) th element of the sample covariance matrix R.

IV. UNSUPERVISED DETECTION

A general overview of the examined ULSS detector is depicted in Fig 1, where it can be observed that it is comprised of two main components during a training stage, i.e., GMM & expectation maximization (EM) algorithms, as well as a DNN.
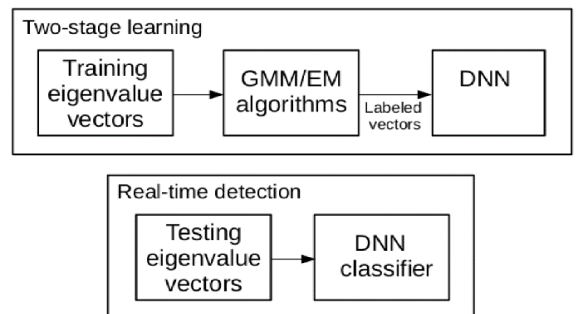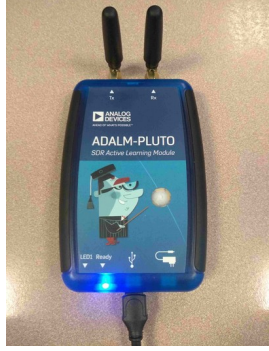

Fig. 1 Block diagram of the evaluated ULSS detector.

(a) PlutoSDR board used for the experimental evaluation. (b) Encoder and modulator to transmit television signals.

Fig. 2 Hardware employed for spectrum sensing

A. GMM/EM algorithms

The input vector to the ULSS detector is composed by the extracted eigenvalues of the sample covariance matrix, i.e., $\lambda = [\lambda_1, \lambda_2, \ldots, \lambda_M,]^T$, which is collected under H_0 and H_1 assumptions during the training stage. Let us consider that the probability density function (PDF) of the eigenvalues asymptotically follows a Gaussian distribution [18], [12], so that an eigenvalue vector follows a mixture of Gaussian distribution[1].

Based on this observation, we resort to an unsupervised clustering algorithm to determine the corresponding PU states. A Gaussian Mixture Model (GMM) assumes that the sample data is drawn from a mixture of multivariate Gaussian distribution as follows,

$$f\left(\lambda \vee \theta_{GMM}\right) = \sum_{k=1}^{K} \tau_k \phi\left(\lambda \vee \mu_k, \Sigma_k\right) \qquad (7)$$

where $\theta_{GMM} = [\tau_k, \mu_k, \Sigma_k]$ denotes the parameters defining the mixture of K Gaussian densities $\phi\left(\lambda \vee \mu_k, \Sigma_k\right) with k = [1, \ldots, K], \tau_k$ is the mixing coefficient satisfying $\Sigma_k \tau_k = 1$, while μ_k and Σ_k denotes the mean and covariance matrix of the Gaussian densities. Given a set of collected eigenvalue vectors S_λ, an EM algorithm [19] is employed to estimate θ_{GMM}, thus the clusters are identified. The membership of the eigenvalue vectors to these clusters is assigned by considering the distance between them in order to get their corresponding labels.

B. DNN

The input to the DNN is given by $\lambda \in R^{M \times 1}$, and it is composed of N_L hidden layers. Each layer contains Q_l neurons with $\iota = [1, 2, \ldots, N_L]$, and the output vector at the ιth layer can be expressed as,

$$o_l = f_l\left(W_l i_l + b_l\right) \qquad (8)$$

where f_l is a non-linear function, $W_l \in R^{Q_l \times Q_{l-1}}$ is a matrix of weights, $i_l \in R^{Q_{l-1}}$ is the input vector, and $b_l \in R^{Q_l}$ is the bias vector. Note that the input vector i_l corresponds to the output of the previous layer, i.e. $i_l = o_{l-1}$. The evaluated DNN architecture is composed of fully connected (FC) layers, and the output is obtained after applying a *softmax* function. The DNN is defined by its parameters denoted by $\theta_{DNN} = \{W, B\}$, where W is a set of weight matrices $\{W1, W2, ..., W_{NL}\}$, and B is a set of bias vectors $\{b1, b2, ..., b_{NL}\}$. Then, during the training stage, a learning process aims to minimizing a loss function in order to estimate θ_{DNN} as follows,

$$L(\theta_{DNN}) = arg \min_{\theta_{DNN}}(o^P - o^T), \qquad (9)$$

where o^P denotes the predicted outputs, and o^T the true outputs obtained during the clustering process. Finally, once the DNN is trained, its detection performance is evaluated.

V. SDR PLATFORM DESCRIPTION

The SDR platform is composed of an ADALM-PLUTO active learning module (PlutoSDR as seen in Fig. 2a)[20]. It supports a radio frequency (RF) range between 325 MHz to 3.8 GHz, and up to a 20 MHz of instantaneous bandwith. Furthermore, it incorporates independent transmitter and receiver channels to support a full-duplex mode. In our platform, the PlutoSDR is connected via USB to a personal computer (PC) working on Linux Ubuntu 18.04 and running GNU Radio [21]. In addition, an encoder/modulator is also employed to generate a digital television signal following the integrated services digital broadcasting - terrestrial (ISDB-T) standard.

In order to emulate a CR scenario, the SDR board is configured to operate in a full-duplex mode. In this way, the transmitted and received channels allow us to emulate a PU and a SU, respectively. During a transmission, the in-phase and quadrature (IQ) samples to be transmitted are sent through

---

[1] It is worth to highlight that for a finite number of samples N_s, the PDF is very complex and not suitable for a simple mixture model. However it will be shown later, that under a Gaussian assumption, frequently used in literature, leads us to meaningful results.
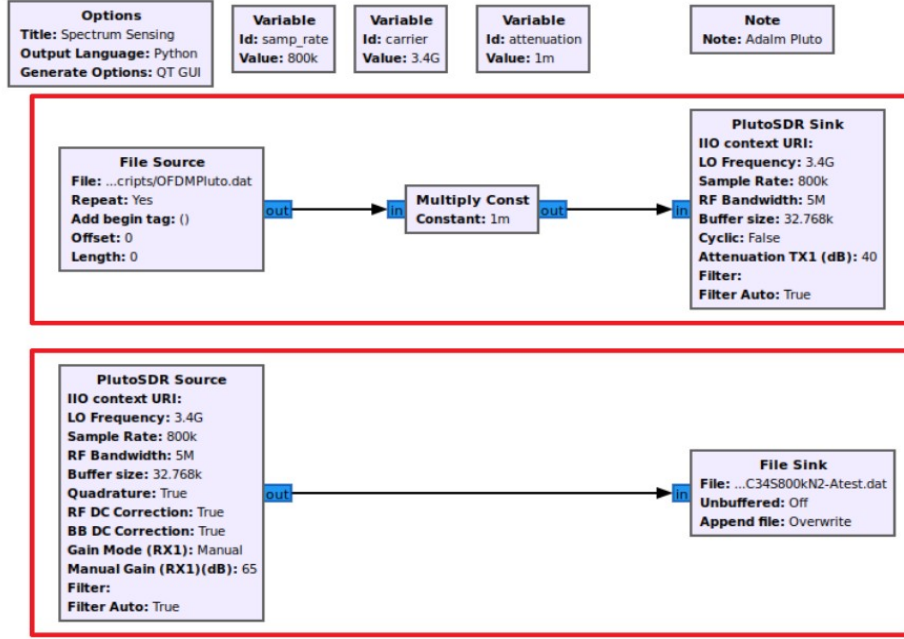
Fig. 3 GNU radio companion (GRC) design for the transmission and reception of signals using PlutoSDR.

the USB controller to the PlutoSDR. Then, these samples go through a digital-to-analog converter (DAC), after which they are upconverted to RF. At the receiver side, the same processing chain is repeated in reverse order. It starts by down-converting the receiver RF signal, then the IQ analog signals are subsequently digitized by an analog-to-digital converter (ADC), so that the samples can be retrieved at the host computer using the USB controller.

A GNU Radio Companion (GRC) design that implements this setup is depicted in Fig. 3, where the transmitter and receiver channels have been highlighted in red boxes. It allows to upload files containing the IQ samples for the transmissions, while saving in files the measurements of the receiver channel. This setup facilitates the experimental evaluations enabling the simultaneous transmission and reception of signals, thus avoiding the introduction of synchronization mechanisms. Moreover, it allows to control the PU and SU from a PC running a single application. Finally, the collected measurements will allow us to apply an off-line processing.

VI. EXPERIMENTAL RESULTS

In this section, we evaluate the ULSS detector using the aforementioned SDR platform. The experiments are conducted under a stationary environment, and the experimental results are obtained with 1000 Monte-Carlo trials for each realization. In our CR scenario, the PU transmits an orthogonal frequency division multiplexing (OFDM) waveform conveying 16 QAM symbols and employing a carrier frequency of 3.4 GHz. The control of the signal-to-noise ratio (SNR) is carried out by modifying the power transmission of the PU, while the SU establishes a number of samples $N_s=50$ per each sensing period.

ULSS training

In Fig. 4, a scatter plot of the λ eigenvalue vectors obtained for a measured SNR of 0.24 dB is shown. It can be appreciated that two clusters of vectors corresponding to the PU states (H_0 and H_1) can be identified, and the variance of each cluster determines the shape of the distribution, where some outliers are also taken into account. Based on these measurements, a GMM allows us to estimate the parameters θ corresponding to these clusters.
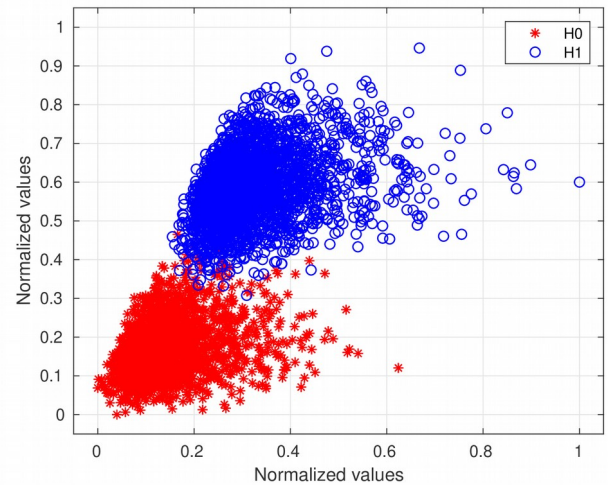

Fig. 4: Eigenvalues extracted from the covariance matrices obtained under a null and alternative hypotheses, H_0 and H_1, for a measured SNR = 0.24 dB on average.
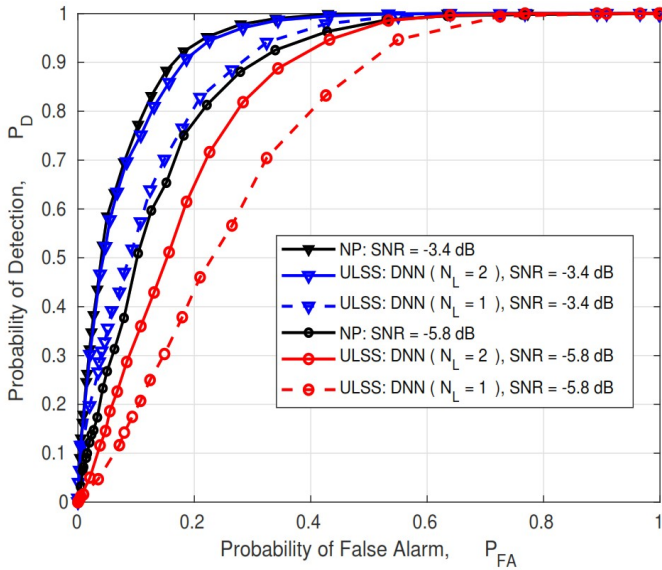
Fig. 5: ROCs curves corresponding to the NP and ULSS detectors for an SNR = −3.4 dB and SNR = −5.8 dB.

Fig. 6: ROC curves for the T_{AGM}, $T_{hadamard}$ and ULSS dectectors, for SNRs = −3.4 dB and −5.8 dB.

In doing so, it was observed that for a higher measured SNR, the clusters are well defined and separated, and consequently the obtained labeled data are often correct. Nevertheless, a trained DNN with this set of data provides poor results when tested for lower SNRs.

On the other hand, if the clusters are not well defined, it will not only be hard for the clustering process, but also it will introduce mislabeled data degrading the DNN performance as it learns from these errors. This problem is overcome by collecting measurements around or above an SNR of 0 dB. It allows the DNN to work for both, higher and lower SNR, and we select the data training shown in Fig. 4 for the current experimental evaluation. Regarding the DNN, it is trained for two architectures, a DNN1 containing a single FC layer, NL = 1, with $Q_1 = 16$ neurons, and another one DNN2 with two FC layers, NL = 2, each of them containing $Q_1 = 16$ and $Q_2 = 16$ neurons, respectively.

Performance evaluation

The performance of the detectors is studied by comparing their receiving operating characteristic (ROC) curves. In Fig. 5, we show the obtained performance of the ULSS detector when employing the aforementioned DNN architectures. It was observed that a DNN2 provides a better accuracy over a DNN1 and without causing overfitting. Furthermore, the ULSS attains a close performance to that exhibited by a Newman-Pearson (NP) detector for a measured SNR of −3.4 dB, while a slightly degradation is shown for an SNR of −5.8 dB.

The detection performance of based-model detectors against the unsupervised detection is also compared taking into account a DNN2. In Fig. 6, it can be appreciated that the
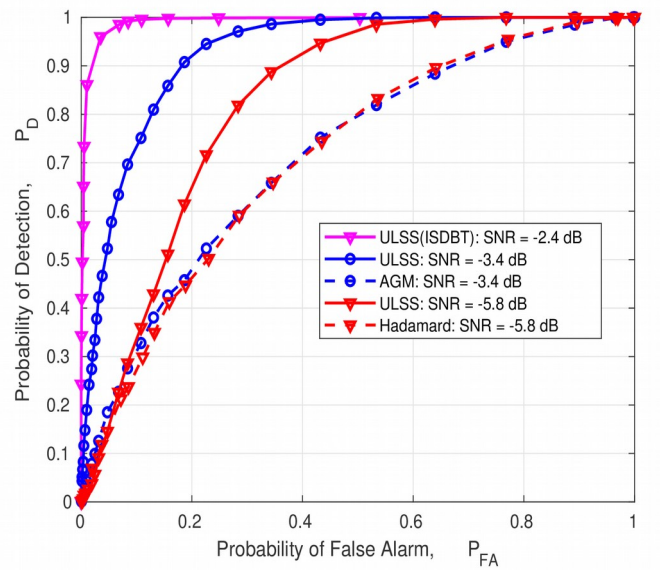
T_{AGM} detector is severely degraded for an SNR of −3.4 dB, which can be explained because it is deduced from a GLRT in which the noise is assumed to be iid. A $T_{hadamard}$ detector is also compared for an SNR of −5.8 dB. Although it is more robust against uncertainties in the noise variance, one of the key advantages of learning-based detector is that it can adapt to the actual data distribution, thus obtaining better results.

In Table I, the obtained detection gain of the ULSS detector over the model-based detection is reported for a $P_{FA} = 0.1$ and $P_{FA} = 0.01$. It can be observed that ULSS provides a significant gain for an SNR = -3.4 dB, while for an SNR = -5.8 dB the gain decreases. Note that the training data was collected around an SNR of 0 dB, thus suggesting us to further improve the training stage.

Table I: Detection gain

ULSS detection gain over model-based detection	SNR = -3.4dB	SNR = -5.8dB
P_{FA} = 0.1	42.4%	5.8%
P_{FA} = 0.01	14%	0%

Moreover, an ISDB-T signal is evaluated using the ULSS detector, where it can be observed that for an SNR of −2.4 dB, it reaches a $P_D \geq 0.9$ with a $P_{FA} = 0.1$ without resorting to labeled data.

Finally, the complexity in terms of real multiplications and sums for the T_{AGM} and $T_{hadamard}$ requires at least one sum and one multiplication[2], while the number of multiplications

---

[2]   Note that other operations such as the square root or the division have not been considered.

and sums for a DNN can be computed as $C_{mult/sums} = \sum_{l=1}^{N_L+1} Q_{l-1} Q_l$, where Q_0 and Q_{N_L+1} correspond to the sizes of the input and output vectors. Then, for the DNN2 with Q_1 = 16 and Q_2 =2 neurons, $C_{mult/sums}$ is equal to 2x16+16x2+2x2 = 68. Thus, a total of 68 multiplications and 68 sums are required.

VII. CONCLUSIONS

In this paper, we have evaluated the performance of two model-based detectors and an unsupervised learning detector through experimental evaluations. A practical setup for an SDR platform is presented for experimental measurements. The study shows that the unsupervised detection obtains a significant gain in comparison to GLRT detectors based on assumptions concerning the noise distribution. Ultimately, these preliminary results confirm the feasibility of ULSS approaches for learning and adapting to the observed data distribution under a more realistic environment, suggesting us the development of novel unsupervised approaches exploiting relevant features of a PU signal.

REFERENCES

[1] J. Mitola and G. Q. Maguire, "Cognitive Radio: making software radios more personal," *IEEE Personal Communications*, vol. 6, no. 4, pp. 13–18, 1999.

[2] S.Haykin,"Cognitive Radio: Brain-Empowered Wireless Communications," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 2, pp. 201–220, 2005.

[3] U. Salama, P. L. Sarker, and A. Chakrabarty, "Enhanced Energy Detection using Matched Filter for Spectrum Sensing in Cognitive Radio Networks," in *2018 Joint 7th International Conference on Informatics, Electronics Vision (ICIEV) and 2018 2nd International Conference on Imaging, Vision Pattern Recognition (icIVPR)*, pp. 185–190, 2018.

[4] A. Nafkha, B. Aziz, M. Naoues, and A. Kliks, "Cyclostationarity-based versus eigenvalues-based algorithms for spectrum sensing in cognitive radio systems: Experimental evaluation using GNU radio and USRP," in *IEEE 11th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, pp. 310–315, 2015.

[5] C. I. M. Althaf and S. C. Prema, "Covariance and eigenvalue based spectrum sensing using USRP in real environment," in *2018 10th International Conference on Communication Systems Networks (COMSNETS)*, pp. 414–417, 2018.

[6] A. Kumar, A. S. Khan, N. Modanwal, and S. Saha, "Experimental Studies on Energy and Eigenvalue Based Spectrum Sensing Algorithms Using USRP Devices in OFDM Systems," *Radio Science*, vol. 55, no. 8, p. e2019RS006973, 2020.

[7] J. Gao, X. Yi, C. Zhong, X. Chen, and Z. Zhang, "Deep Learning for Spectrum Sensing," *IEEE Wireless Communications Letters*, vol. 8, no. 6, pp. 1727–1730, 2019.

[8] J. Xie, J. Fang, C. Liu, and X. Li, "Deep Learning-Based Spectrum Sensing in Cognitive Radio: A CNN-LSTM Approach," *IEEE Communications Letters*, vol. 24, no. 10, pp. 2196–2200, 2020.

[9] Z. Chen, Y.-Q. Xu, H. Wang, and D. Guo, "Deep STFT-CNN for Spectrum Sensing in Cognitive Radio," *IEEE Communications Letters*, vol. 25, no. 3, pp. 864–868, 2021.

[10] J. Xie, J. Fang, C. Liu, and L. Yang, "Unsupervised Deep Spectrum Sensing: A Variational Auto-Encoder Based Approach," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 5, pp. 5307–5319, 2020.

[11] N. A. Khalek and W. Hamouda, "Unsupervised two-stage learning framework for cooperative spectrum sensing," in *IEEE International Conference on Communications (ICC)*, pp. 1–6, 2021.

[12] S. Majumder, "A Gaussian mixture model method for eigenvalue-based spectrum sensing with uncalibrated multiple antennas," *Signal Processing*, vol. 192, p. 108404, 2022.

[13] A. M. Wyglinski, D. P. Orofino, M. N. Ettus, and T. W. Rondeau, "Revolutionizing software defined radio: case studies in hardware, software, and education," *IEEE Communications Magazine*, vol. 54, no. 1, pp. 68–75, 2016.

[14] J. Manco-Vasquez, J. G. Teran, J. Perez-Arriaga, J. Ibañez, and I. Santamaria, "Experimental evaluation of multiantenna spectrum sensing detectors using a cognitive radio testbed," in *Proc. of IEEE International Symposium on Signals, Systems, and Electronics (ISSSE)*, pp. 1–5, 2012.

[15] S. Kay, *Fundamentals of Statistical Signal Processing: Detection Theory*. Upper Sadle River, NJ: Prentice Hall, 1998.

[16] R. Zhang, T. J. Lim, Y.-C. Liang, and Y. Zeng, "Multi-antenna based spectrum sensing for cognitive radios: A GLRT approach," *IEEE Transactions on Communications*, vol. 58, no. 1, pp. 84–88, 2010.

[17] A. Leshem and A.-J. van der Veen, "Multichannel detection of Gaussian signals with uncalibrated receivers," *IEEE Signal Processing Letters*, vol. 8, no. 4, pp. 120–122, 2001.

[18] M. Jin, Q. Guo, Y. Li, J. Xi, and D. Huang, "Spectrum Sensing Using Multiple Large Eigenvalues and Its Performance Analysis," *IEEE Internet of Things Journal*, vol. 6, no. 1, pp. 776–789, 2019.

[19] C. M. Bishop, *Pattern recognition and machine learning*. Berlin, Heidelberg: Springer-Verlag, 2006.

[20] "*ADALM-PLUTO*." https://www.analog.com/en/design-center/evaluation-hardware-and-software/evaluation-boards-kits/adalm-pluto.html. Accessed: 2021-10.

[21] "*GNU Radio*." https://www.gnuradio.org/. Accessed: 2021-10.