Adversarial Examples Detection of Radio Signals Based on Multifeature Fusion

Dongwei Xu, Hao Yang, Chuntao Gu, Zhuangzhi Chen, Qi Xuan, *Member, IEEE*, Xiaoniu Yang

***Abstract*—In the field of deep learning, deep neural networks (DNNs) have shown good performance on classification applications. However, a DNN model is vulnerable to adversarial examples, which is formed by adding tiny perturbations on a normal example and can mislead the DNN model to make a wrong estimate during the prediction. In this paper, for adversarial attacks in radio signals field, we propose a novel adversarial example detection strategy based on multifeature fusion and provide a framework which includes generating adversarial examples, extracting the local intrinsic dimensionality (LID) features and the constellation diagram (CD) features, detecting adversarial examples. We obtain the output values of normal examples and adversarial examples in each layer of the model respectively, and then, calculate the LID features values of examples by the maximum likelihood estimate based on a certain neighborhood range. Meanwhile, we calculate the CD features values by the range feature and density feature of the constellation diagram distribution. Finally, a logistic regression classifier is trained based on multifeature fusion values to detect adversarial examples. The experimental results across two benchmark datasets demonstrate that the proposed multifeature fusion method could accurately detect adversarial examples of radio signals. The detection accuracy is up to 98.7% when the perturbation reached 10%.**

***Index Terms*—adversarial attacks, adversarial detection, deep learning, signal modulation classification**

I. Introduction

WITH the resurgence of research on artificial intelligence, deep neural networks (DNNs) have exhibited outstanding performances in various areas, such as image classification [1], speech recognition [2], and natural language processing [3]. In addition, DNNs are widely applied in radio signals regions. For example, Zha *et al.* [4] proposed a deep learning-based approach for signal recognition, which had an accuracy of up to 98% at 6 dB. Uppal *et al.* [5] built a robust radio frequency signal classifier that had better accuracy and lower computational requirements to classify different types of signal modulations in radio transmissions.

Whereas DNN models have achieved considerable success, researchers have found that they are very vulnerable to adversarial examples. Typically, the adversarial examples are very similar to the original ones but can mislead the model to make a wrong prediction. Goodfellow *et al.* [6] designed a model to effectively and quickly find a perturbation according to the gradient of the model, reducing the classification confidence to fool the model. Carlini *et al.* [7] proposed a general optimization strategy, which has smaller perturbation and more attack power. Therefore, it is highly significant and meaningful to detect such adversarial examples. Thus far, some methods of detection in the vision area have been proposed. Feinman *et al.* [8] proposed two methods, namely, kernel density estimates (KDEs) and Bayesian uncertainty estimates (BUEs), which have been used to detect adversarial examples in image examples. Imani *et al.* [9] proposed a multifidelity Bayesian optimization framework that significantly scales the learning process of a wide range of existing inverse reinforcement learning techniques. Zhu *et al.* [10] proposed a systematic dual-domain adversary defense countermeasure based on the conditional variational autoencoder and Bayesian network (BN). The composite and hierarchical BN detector was developed that can unite complementary domains from both features and residuals, and also can conduct double-check-based detection in each domain. Wang *et al.* [11] proposed adversarial posterior distillation which has good performance on downstream applications including anomaly detection, active learning, and defense against adversarial attacks.

However, all of the above methods were tested in vision domain, there are few approaches to detect adversarial examples in radio signals domain. In Ref. [12], median absolute deviation-based outlier detection, dimensionality reduction, and clustering detection methods were used to detect adversarial examples. There are great differences in data form and attribute between the vision and the radio signal domain. Compared to the visibility of vision, the feature of radio signals is abstract and complex. Radio signals have unique attributes such as phase, amplitude, frequency, and power. Therefore, designing an adversarial example detection method in the signal field is an innovative work, which plays an important role in improving the robustness of signal recognition.

In this paper, we propose an adversarial detection method

This work was supported in part by the National Natural Science Foundation of China under Grant (61903334, 61973273), in part by the Zhejiang Provincial Natural Science Foundation under Grant (LQ16E080011, LY21F030016, LR19F030001), and in part by the China Postdoctoral Science Foundation under Grant 2018M632501. (Corresponding author: Qi Xuan.)

Dongwei Xu, Hao Yang, Chuntao Gu, Zhuangzhi Chen, Qi Xuan are with the Institute of Cyberspace Security, Zhejiang University of Technology, Hangzhou 310023, China (e-mail: xuanqi@zjut.edu.cn).

Xiaoniu Yang is with the Institute of Cyberspace Security, Zhejiang University of Technology, Hangzhou 310023, China, and also with the Science and Technology on Communication Information Security Control Laboratory, Jiaxing 314033, China.

based on multifeature fusion for radio signal modulation types classification. We analyze the deep representations of examples on the DNN layers and calculate LID values. We calculate CD values through the distribution of the points in the constellation diagram. We develop a systematic detection framework with deep learning techniques including generating adversarial examples, extracting features and detecting adversarial examples. The experiment results show that the detection algorithm has good performance.

The rest of the paper is organized as follows. In Section II, we provide a description of the local intrinsic dimensionality method. The experiments performed to evaluate this method are described in Section III. Finally, the conclusion is summarized in Section IV.

II. Local Intrinsic Dimensionality

As an important attribute of the manifold and a vital parameter of manifold learning algorithms, the intrinsic dimensionality of the manifold has attracted extensive attention and has been applied in [13] [14]. Gionis *et al*. [15] proposed the local intrinsic dimensionality based on the geometric analysis technology and applied it to clustering. On the manifold, according to the number of neighbors, the local range of each point increases with an increase in the radius, which can provide the local correlation dimension of this point. Then, the local correlation dimension can be used as an estimate of the intrinsic dimension of this point.

For example, there was roundness in *d* uniformly distributed dimensions in the Euclidean space; when its radius changed from r_1 to r_2, the rate of size *S* change could be expressed as follows:

$$\frac{S_2}{S_1} = (\frac{r_2}{r_1})^d . \tag{1}$$

Then, *d* could be derived as follows:

$$d = \frac{\ln(S_2/S_1)}{\ln(r_2/r_1)} . \tag{2}$$

A. Local Intrinsic Dimensionality

Definition: *Assuming that there is a data example* $x \in \mathbf{X}$, *there exists variable* $R > 0$ *that denotes the distance from x to another example of data. If the cumulative distribution function* $F(r)$ *of R is positive, continuous, and differentiable at distance* $r > 0$, *the LID of x at distance r can be expressed as follows*:

$$lid_F(r) = \lim_{\varepsilon \to 0^+} \frac{\ln(F(r\cdot(1+\varepsilon))/F(r))}{\ln(1+\varepsilon)} . \tag{3}$$

Function $F(r)$ is similar to the size *S* in (1); the result of (3) can be calculated by applying the L Hospital Theory whenever the limit exists, which is given as follows:

$$lid_F(r) = \frac{r\cdot F'(r)}{F(r)} . \tag{4}$$

Therefore, when $r \to 0$, the local intrinsic dimensionality of *x* can be in turn written as the following limit:

$$lid_F = \lim_{r\to 0} lid_F(r) . \tag{5}$$

where lid_F expresses the rate of change of the function $F(r)$ when the distance *r* increases from 0 and can be assessed by the distance of *x* to its *k* nearest neighbors in the examples in [16].

B. Estimation of LID

Conceptually, when the surrounding of *x* is distributed uniformly in the submanifold, the value of lid_F is equal to the dimension of the submanifold. Nevertheless, the value of lid_F is not completely applicable when using a real dataset. In the extreme value theory, the distance of the k-nearest neighbors of the reference point can be regarded as the extreme events associated with the low tail of the distribution of distance. Under very reasonable assumptions, the tails of a continuous probability distribution converge to the generalized Pareto distribution (GPD), a form of power law distribution. For this, the researchers developed an estimator of LID using the maximum likelihood estimator (MLE), given a reference example $x \approx P$, where *P* represents the data distribution; the MLE of the LID at *x* is defined as follows:

$$lid(x) = -(\frac{1}{k}\sum_{i=1}^{k}\log\frac{r_i(x)}{r_k(x)})^{-1} . \tag{6}$$

Here, $r_i(x)$ is the distance between *x* and its *i-th* nearest neighbor within an example of points drawn from *P* and $r_k(x)$ is the maximum of the neighbor distances.

The working principle of the LID method is that there are differences in the data distribution between normal examples and its adversarial examples. And the method of LID can measure the distance between an example and its neighbor examples on principle. The experiment proved that the distance between normal examples and their neighbors is obviously different from the distance between adversarial examples and their neighbors. We use these LID features combined with CD features to realize the detection of adversarial examples. Specifically, we train a logistic regression classifier based on multifeature fusion to detect adversarial examples.

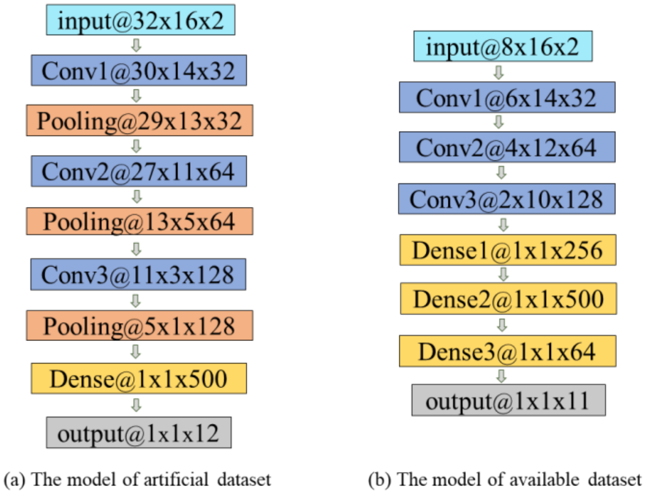

Fig. 1. Architecture of DNN model.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSII.2021.3095663, IEEE Transactions on Circuits and Systems II: Express Briefs

III. Experiments and Results Analysis

A. *Datasets and Experimental settings*

1) Datasets and Models

The experiment was conducted on two datasets: one of them is a publicly available dataset (DS1), which is the same as the dataset used in Ref. [17]. The other is an artificial dataset (DS2) that we generated, and details are as follows: DS2 contained 12 modulations, including BPSK, QPSK, 8PSK, OQPSK, 2FSK, 4FSK, 8FSK, 16QAM, 32QAM, 64QAM, 4PAM, and 8PAM. Each SNR of the modulations was in the range from −20 dB to 30 dB.

The complex number signal is generally used to represent the real number signal in the process of signal processing. The complex number signal can avoid the influence caused by the spectrum of the real number signal with conjugate symmetry. When the complex number signal is sampled, I and Q channels need to be sampled at the same time. So, each example in DS2 was split into I and Q signals, and each signal contained 512 points. We selected high SNR data whose SNR is above 10 dB for the experiment in these two datasets. High SNR data has less noise and more clear characteristics of radio signals.

For these datasets, two models were used. Fig. 1(a) shows the architecture of the DNN model used for DS2, which composed of three convolution layers and max-pooling layers, one regular densely connected layer, and one dropout layer, the parameters are set as [(32,3,3), (64,3,3), (128,3,3)], [(2,2), (2,2), (2,2)], [500], [0.5], respectively. It could achieve 82.62% classification accuracy on the test data of DS2. As shown in Fig. 1(b), three convolution layers and three regular densely connected layers were used for DS1, the parameters are set as [(32,3,3), (64,3,3), (128,3,3)], [256, 500, 64], respectively. The model prediction accuracy on the test data of DS1 is 79.21%. To match the input of the model, the input shape of the signal is modified without destroying the continuity of the signal as far as possible.

2) Experimental settings

The two datasets were divided respectively into a training data and a test data. The DNN model was trained using the training data of the dataset. Furthermore, the data of the valid examples (called normal examples) in the test data were selected by using the correct predictions of the model, which could generate adversarial examples by the adversarial attacks method [6][7][18]. Then, the LID features and CD features were combined as multifeature. Next, the values of multifeature fusion of normal examples and adversarial examples were split, respectively, into the training data and the test data. The logistic regression classifier trained by the training data of multifeature fusion was used as a detector to detect the test data of multifeature fusion; its accuracy (ACC) score showed the detection ability. The framework illustration is as follow Fig. 2.

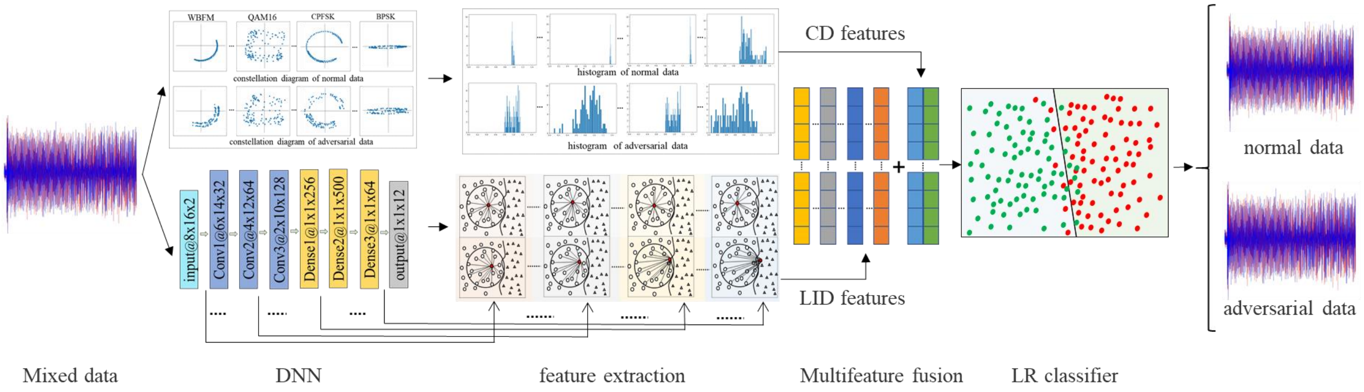

Fig. 2. The framework illustration of adversarial examples detection.

B. *Adversarial Example Generation*

Adversarial examples are generally similar to normal examples but can fool the DNN model of classification to make a wrong prediction. In deep learning, the methods of adversarial attacks from radio signals are almost migrated from vision domain, i.e., they are not very different between vision and radio signals domain. In this paper, we used different attack methods to generate adversarial examples.

1) Fast Gradient Sign Method

The fast gradient sign method (FGSM) directly added a small step ε to a normal example according to the gradient direction with the input normal example in [6]. The adversarial example $x^{'}$ can be defined as follows:

$$x^{'} = x + \varepsilon \cdot sign(\nabla_x L(x, y)) \,. \tag{7}$$

where $\nabla_x L(x, y)$ is the first-order derivative of the loss function corresponding to inputs *x* and *y*. Furthermore, *sign* () is the sign function.

2) Basic Iterative Method

The basic iterative method (BIM) in [18] is one of the many extensions of FGSM, implying that it achieves an attack for an example through multiple iterations of the FGSM method.

$$x^{'}_0 = x, x^{'}_{n+1} = Clip_\delta\{x^{'}_n + \varepsilon \cdot sign(\nabla_x L(x, y))\} \,. \tag{8}$$

where *n* is the total number of iterations and *Clip*{} is the clipping operator function used to limit the examples to a given range.

In this study, two methods based on BIM were used. BIM-a expresses for each example the first-time step of the attack. BIM-b is the adversarial example of the last step of iterative attacks.

3) Carlini and Wagner Attacks

Carlini and Wagner (CW) [7] is an optimized attack. The loss function within an unconstrained optimization formulation can be expressed as:

$$L(x^{'}, t) = \max(\max_{i \neq t}\{Z(x^{'})_i\} - Z(x^{'})_t, -k) \tag{9}$$

$$\min(\| x^{'}(w) - x \|_2^2 + c \cdot L(x^{'}(w), t)) \qquad (10)$$

where $Z(x')_i$ denotes the output logits of the pre-softmax layer for class i; t denotes the target label; k is the confidence constant that controls the degree of attack; w is an auxiliary variable; c is a binary search selection constant.

C. Baselines

In Ref. [8], two detection methods were proposed, namely, kernel density estimation (KDE) and Bayesian uncertainty estimation (BUE), which were used to detect adversarial examples in the image examples. In this study, we used the experimental results of these two datasets obtained by the BUE and KDE detection methods as the baselines when the perturbation reached 10%.

Table I RESULTS OF LID AND BASELINES

Dataset	Method	FGSM	BIM-a	BIM-b	CW
DS1	BUE	49.73	49.97	50.16	50.21
	CD	62.74	72.14	68.11	61.89
	KDE	65.67	95.04	81.65	76.82
	LID	93.94	97.40	96.62	88.36
	LID+CD	**94.62**	**98.01**	**97.33**	**89.41**
DS2	BUE	44.08	51.28	42.29	47.96
	CD	51.29	55.71	52.15	54.83
	KDE	57.59	51.54	56.91	52.73
	LID	**92.74**	80.01	95.17	85.42
	LID+CD	**92.74**	**80.73**	**96.81**	**86.37**

Table I shows the comparison of the proposed method with the baselines. We observed that our LID+CD estimation method achieved the best performance.

D. Results Analysis

1) LID Features and CD Features

Fig. 3 shows the situation of LID value of the min–max normalization examples on the layers of the DNN model. The picture on the left is based on DS1 and that on the right is based on DS2. With an increase in the depth of the layer of the DNN model, the gap of LID value between adversarial examples (red, blue, green and cyan lines) and normal examples (black line) became increasingly distinct. This provided a favorable proof of the feasibility of our detection method. Therefore, we could use LID value of the examples to train a classifier to detect adversarial examples.

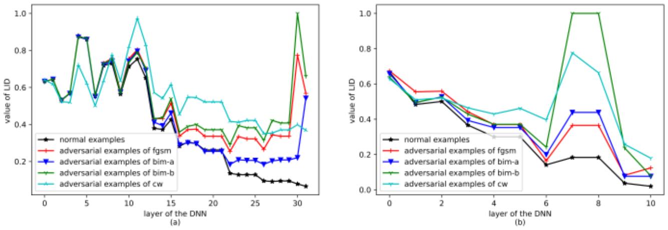

Fig. 3. Value of LID of normalized examples on the layers of the DNN model.

The values I channel and Q channel of the modulated signal are regarded as the coordinate value of a point in a two-dimensional coordinate system. The constellations diagram of some modulation types is shown in Fig. 4. The first line represents the constellations diagram of normal examples and the second line represents the constellations diagram of corresponding adversarial examples. For example, the distribution of constellation diagram points of a normal example is relatively dense and close, but the distribution of constellation diagram points of an adversarial example is looser. The perturbation added to the normal example by the adversarial attack changes the I channel and Q channel values, so the adversarial example is less dense in the constellation diagram. Therefore, we use the density value and the distance value as features to detect adversarial examples. By combining LID features and CD features, the multifeature detection method achieves better detection results.

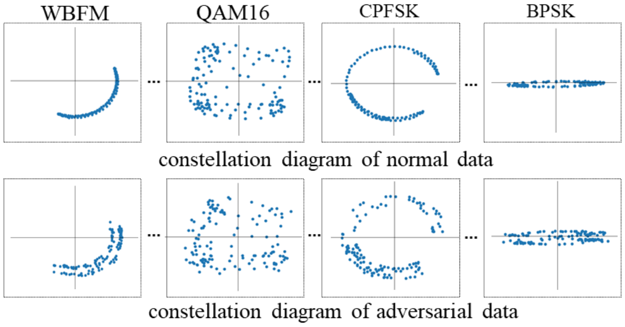

Fig. 4. Constellation diagram of partially modulation types signals in DS1.

2) Influence of Different Perturbations on Accuracy

In this paper, the magnitude of the perturbation could be controlled by setting the ε during the generation of adversarial examples. In addition, the relative amount of the L2 norm was used to measure the rate of change of the example before and after an attack. Fig. 5 shows the detection results (red, blue, green and cyan lines) of the detector and the accuracy of model classifier (black line) on the adversarial examples under different perturbations. The picture on the left is based on DS1 and that on the right is based on DS2.We can see that with an increase in the norm, the accuracy of the detector tended to be stable and the accuracy of the model classifier decreased. When the perturbation is less than 10%, the performance of this method is slightly decreased. The reason is that the perturbation is very small and the feature is not obvious between normal examples and adversarial examples.

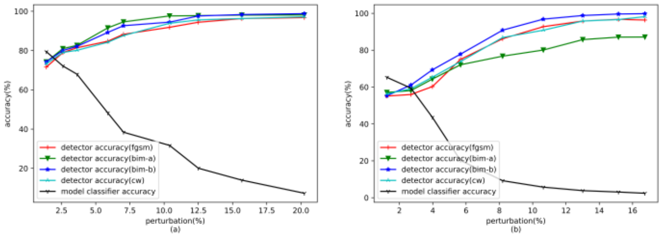

Fig. 5. Influence of different perturbations on accuracy.

IV. CONCLUSION

In this paper, we proposed an adversarial example detection method for radio signal modulation classification on the basis of multifeature fusion. Through our method, we could easily obtain the value of LID for each layer of the DNN model and the value of the constellation diagram features. In addition, we showed the comparison of the proposed method with the baselines and found that our method achieved the best performance. This work could enhance the security of radio signals demodulation process, which is of considerable significance to the application of deep learning in radio signal processing.

REFERENCES

[1] Krizhevsky A, Sutskever I, Hinton G E. “Imagenet Classification with Deep Convolutional Neural Networks,” *Communications of the ACM*, vol. 60, no. 6, pp.84-90. May.2017. doi: https://doi.org/1.0.1145/3065386.

[2] L. Li, D. Wang, Y. Chen, Y. Shi, Z. Tang and T. F. Zheng, “Deep Factorization for Speech Signal,” *2018 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2018, pp. 5094-5098, doi: 10.1109/ICASSP.2018.8462169.

[3] Y. Liao and Y. Wang, “Some Experiences on Applying Deep Learning to Speech Signal and Natural Language Processing,” *2018 World Symposium on Digital Intelligence for Systems and Machines*, 2018, pp. 83-94, doi: 10.1109/DISA.2018.8490638.

[4] X. Zha, X. Qin, Y. Zhou and H. Peng, “Power of Deep Learning for Amplitude-phase Signal Modulation Recognition,” *2019 IEEE 8th Joint International Information Technology and Artificial Intelligence Conference*, 2019, pp. 454-458, doi: 10.1109/ITAIC.2019.8785607.

[5] A. J. Uppal, M. Hegarty, W. Haftel, P. A. Sallee, H. Brown Cribbs and H. H. Huang, “High-Performance Deep Learning Classification for Radio Signals,” *2019 53rd Asilomar Conference on Signals, Systems, and Computers*, 2019, pp. 1026-1029, doi: 10.1109/IEEECONF44664.2019.9048897.

[6] Goodfellow I J, Shlens J, Szegedy C. “Explaining and harnessing adversarial examples,” 2014. arXiv:1412.6572.

[7] N. Carlini and D. Wagner, “Towards Evaluating the Robustness of Neural Networks,” *2017 IEEE Symposium on Security and Privacy*, 2017, pp. 39-57, doi: 10.1109/SP.2017.49.

[8] Grosse K, Manoharan P, Papernot N, et al. “On the (Statistical) Detection of Adversarial Examples,” 2017. arXiv:1702.06280.

[9] M. Imani and S. F. Ghoreishi, “Scalable Inverse Reinforcement Learning Through Multifidelity Bayesian Optimization,” *IEEE Transactions on Neural Networks and Learning Systems*, doi: 10.1109/TNNLS.2021.3051012.

[10] J. Zhu, G. Peng and D. Wang, “Dual-Domain-Based Adversarial Defense with Conditional VAE and Bayesian Network,” *IEEE Transactions on Industrial Informatics*, vol. 17, no. 1, pp. 596-605, Jan. 2021, doi: 10.1109/TII.2020.2964154.

[11] Wang K C, Vicol P, Lucas J, *et al*. “Adversarial Distillation of Bayesian neural Network Posteriors,” *International Conference on Machine Learning*. PMLR, pp. 5190-5199, 2018.

[12] S. Kokalj-Filipovic, R. Miller and G. Vanhoy, “Adversarial Examples in RF Deep Learning: Detection and Physical Robustness,” *2019 IEEE Global Conference on Signal and Information Processing*, 2019, pp. 1-5, doi: 10.1109/GlobalSIP45357.2019.8969138.

[13] J. A. Costa and A. O. Hero, “Geodesic Entropic Graphs for Dimension and Entropy Estimation in Manifold Learning,” *IEEE Transactions on Signal Processing*, vol. 52, no. 8, pp. 2210-2221, Aug. 2004, doi: 10.1109/TSP. 2004.831130.

[14] Houle M E. “Local Intrinsic Dimensionality I: an Extreme-value-theoretic Foundation for Similarity Applications,” *International Conference on Similarity Search and Applications*. Springer, Cham, pp. 64-79. 2017.

[15] Gionis A, Hinneburg A, Papadimitriou S, et al. “Dimension Induced Clustering,” *Proceedings of the 11th ACM SIGKDD International Conference on Knowledge Discovery in Data Mining*, pp. 51-60, 2005.

[16] Amsaleg L, Chelly O, Furon T, et al. “Estimating Local Intrinsic Dimensionality,” *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. pp. 29-38, 2015.

[17] O’Shea T J, Corgan J, Clancy T C. “Convolutional Radio Modulation Recognition Networks,” *International Conference on Engineering Applications of Neural Networks*. Springer, Cham, pp. 213-226, 2016.

[18] A. Kurakin, I. J. Goodfellow, and S. Bengio, “Adversarial Machine Learning at Scale,” 2017. arXiv:1611.01236.