Classification of PC Baseband Signals from Wireless Egress

1st M. Ahmed Leghari
Computer Science and Engineering
Jacobs University Bremen
Bremen, Germany
m.leghari@jacobs-university.de

2nd Sina M. Pralle
Computer Science and Engineering
Jacobs University Bremen
Bremen, Germany
s.pralle@jacobs-university.de

3rd Sören F. Peik, *senior member, IEEE*
Electrical Engineering and Computer Science
Bremen University of Applied Sciences
Bremen, Germany
speik@hs-bremen.de

4th Sebastian Lütje
Development and TEMPEST
Aerospace Data Security GmbH
Bremen, Germany
sebastian.luetje@aerospace-datasecurity.de

5th Werner Henkel, *senior member, IEEE*
Computer Science and Engineering
Jacobs University Bremen
Bremen, Germany
werner.henkel@ieee.org

***Abstract*—Being related to so-called TEMPEST activities, we show some possibilities to classify standard signal emissions from a PC environment, such as Ethernet, USB, or HDMI. Hereto, we make use of time-frequency analysis followed by deep learning or alternatively, by looking into individual signal properties, such as symbol rates, special fixed patterns, common-mode presence, or amplitude histograms.**

We will see that the distinction of such standard signals can be very efficient, even in the case of 100Base-T and 1000Base-T Ethernet that are designed to use the same frequency range and hence have a very similarly looking time-frequency representation. Under noise-free conditions, 100 % correct classification is achieved.

***Index Terms*—TEMPEST, field probe, signal classification, neural network, deep learning**

I. Introduction

Our works belong to the so-called "TEMPEST" [1] activities. This is a United States government term related to limiting electric or electromagnetic radiation emanations from electronic equipment. The acronym stands for Transient Electromagnetic Pulse Emanation Standard. We are interested, how well near-field probing of circuitry would unveil the signal format and how easy it might be to synchronize and finally obtain data. Such kind of eavesdropping cannot easily be recognized, since one does not directly connect to on-board conductors or cables. Furthermore, signal components might be emitted before they are protected by encryption.

For our results in here, we simulated the corresponding signals and the probing and investigate some options, how to determine the corresponding signal format and its special characteristics. We focus on some very standard baseband signals that are used in PC environments, such as USB 2.0 [2], Ethernet 100Base-T and 1000Base-T [3], and HDMI [4].

Signal classification is, of course, since long a topic for armed forces, trying to eavesdrop communications without knowing the modulation format. Publications from such works are typically limited. Nevertheless, there is public literature covering the classification of modulation methods, less so based on near-field RF radiation measurements and also usually not covering baseband broadband signals, but rather narrowband modulated signals. Nevertheless, we give a short overview of some of the literature. Cyclostationary signal properties and cumulants are used in [5], [6]. The spectral correlation function is investigated in [7]. In [8], [9], Wavelets (Haar) and the Wigner-Ville distribution are applied.

For the final classification, typical machine-learning methods can be applied, such as SVN (support vector machine) [10]–[12], Random Forest [13] (seemingly not yet applied to modulation classification), or Neural Networks [9], [11], [14], [15].

Dimension reduction algorithms may support classification such as Independent Component Analysis [10], Principle Component Analysis, or Classical Multidimensional Scaling. For clustering, possibilities are k-means (vector quantization, Lloyd-Max quantizer, Linde-Buzo-Gray alg.) or so-called Hierarchical Agglomerative Cluster Analysis [16].

When sparse representations can be utilized, then Compressive Sensing methods come into play, such as in [9], [17], [18].

Our classification choices were two-fold. On the one hand, we went for a neural network-based one that can easily be extended to new signal formats and is based on time-frequency picture pattern recognition. After having selected a suitable time-frequency representation, the network choice is not specific for our application. On the other hand, we try to synchronize to the symbol clock of all signals and detect specific signal properties. All signals use twisted pairs, usually in a balanced fashion. To already mention some examples for specific signal properties, USB does not only have special headers after idle periods, it ends a block with a common mode component, which is very visible, since egress is, of course, bigger for common mode signals. HDMI has, of course, a

We gratefully acknowledge funding by the Federal Ministry for Economic Affairs and Energy (BMWi) under the ZIM program

frame structure with vertical and horizontal blanking intervals, control periods, and data islands protected with a BCH code. Ethernet 100Base-T and 1000Base-T not only use 2 or 4 twisted pairs, respectively, they are also using different signal values, namely 3 or 5, respectively, with different underlying line coding. Otherwise, the two are designed to use the same spectral width.

The following sections are devoted to the two approaches. Beforehand, however, the signal properties are handled in some more detail in Section II-C. H-field probing is discussed in Section III. To prepare for the neural-network-based approach, Section V introduces a few time-frequency methods, followed by the neural network treatment in Section VI. The approach based on specific signal properties is described in Section VII, and we conclude with Section VIII.

II. Some PC-typical signals and its specific properties

A. Universal Serial Bus

USB signals were generated according to the Universal Serial Bus Specification Revision 2.0 [2]. Within this standard, two types of USB 2.0 were defined with different data rates. The full-speed USB signaling bit rate is 12 Mb/s, whereas the high-speed USB rate is 480 Mb/s.

Bits are transmitted over a twisted pair using D+ and D−, simultaneously, employing NRZI encoding. Consequently, a "0" is represented as a change in voltage level, while for transmission of a "1" the voltage level will remain constant. In order to transmit a long string of ones without losing clock synchronization, bit stuffing is implemented. The voltage level of the D+ and D− lines are categorized into three states for full-speed and high-speed USB.: J, K, and idle. The J state is defined as high voltage on D+ and low voltage on D- and K as the opposite.

Both signal types contain a start-of-packet followed by a synchronization sequence preceding every data packet, therefore allowing for clock synchronization at the receiver side. The start-of-packet is hereby defined as a switch from the idle state to the K state, which is also the first symbol of the synchronization pattern for both signal types. However, full-speed and high-speed USB synchronization patterns are of different lengths. The full-speed pattern is eight symbols long and made up of 3 KJ pairs followed by 2 Ks, whereas the high-speed pattern has a length of 32 symbols and contains 15 KJ pairs followed by 2 Ks . Furthermore, each packet contains an end-of-packet (EOP) sequence to allow the receiver to recognize the end. For full-speed, the EOP is differentiated from other signal parts by its common-mode part, as both lines are set to low voltage for two symbol periods before going to the J state. The high-speed EOP is recognizable by a state transition followed by 7 symbol periods of constant voltage. Consequently, these special signal characteristics allow for USB to be easily distinguishable.

B. High-Definition Multimedia Interface

For HDMI, according to the standard [4] there are a few interesting signal properties that can be used for classification. In HDMI there are two main types of encoding used depending on if audio or video data is being sent. The video data is encoded in a transition minimized way while the audio data is not. Preceding every data island or video data period a control period is sent containing a preamble indicating the type of data following. This preamble consists of eight identical control characters. Furthermore, the preamble is sent simultaneously on channels 1 and 2, where the repeating control character is different for the two channels. HDMI also has guard band areas that are also known patterns and could be used likewise. Lastly, in the HDMI cable there is also a clock signal being sent over its own wire pair. For our case, however, we assume not to have access to the clock signal wires.

C. Ethernet

100Base-T and 1000Base-T as defined in the IEEE standard [3] were used in this project. Although both systems operate at a symbol rate of 125 MBaud, their data transmission rates differ. 100Base-T transmits over two twisted pairs of type CAT-5 with a data transmission rate of 100 Mb/s in full-duplex. 1000Base-T, however, transmits over four twisted pairs of type CAT-5, where each pair is able to provide a data transmission rate of 250 Mb/s in each direction, simultaneously. A scrambler is used for both types of Ethernet. Furthermore, different encoding schemes are applied. 100Base-T is encoded using the 4B/5B code followed by MLT-3 encoding. This line code uses 3 voltage levels: -1,0,1. A "1" in the data is encoded as a transition to the next state, whereas a "0" causes the voltage level to stay constant. In 1000Base-T, the data bits are encoded using 8B1Q4 and 4D-PAM5 line coding, resulting in five voltage levels to be sent over four channels (twisted pairs), simultaneously.

100Base-T and 1000Base-T contain a start-of-stream delimiter and an end-of-stream delimiter. Additionally, 100Base-T sends a preamble prior to a data stream. Nevertheless, these are only used to mark the beginning and end of a stream for the receiver as the clock is transmitted on a separate line.

III. H-field probing and properties

For picking up emissions from a system or circuit board, E- or H-field probes can be used. We mostly went for H-field probes which one can essentially see as a coil with just one turn. H-field probes are certainly preferred for low-ohmic connections like with our investigated signal types and are sensitive to current directions in contrast to E-field probes. Usually, they are the first choice to scan for egress from circuit boards or cables, possibly followed by also an E-field inspection. H-field probes are available in different diameters, where they are becoming more sensitive with growing diameter, but spatial resolution is becoming worse [19]–[21]. Hence, there is a trade-off between sensitivity and spatial resolution. Figure 1 shows a simple H-field probe close to the end of a terminated USB cable.

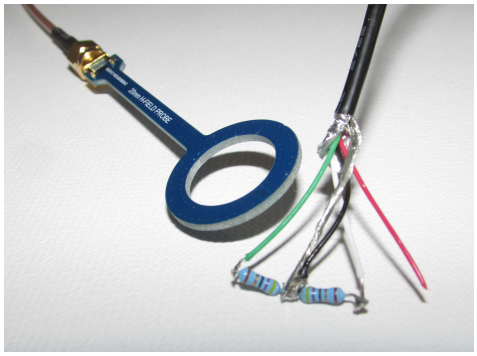

Fig. 1. H-field probe close to a USB-cable with termination

One can see the H-field probe as the secondary side of a transformer and the impedance-matrix entry of a transformer is just $j\omega M$, where M is the mutual induction. This means the phase would just be $\pm\pi/2$ (sign changes when the probe is turned around; an additional linear phase accounts for some additional delay) and the amplitude response grows linearly with ω. Practically, however, there are visible resonance effects at higher frequencies and the frequency response there depends a lot on the type of probe, and even worse, varies with the slightest position change.

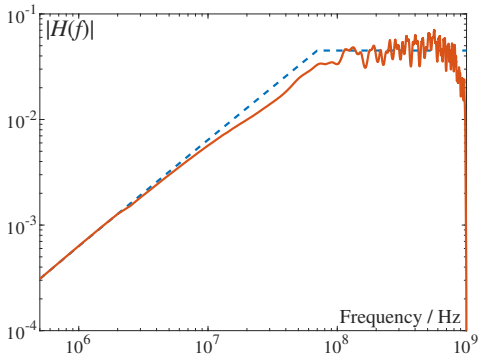

Fig. 2. Simple H-field probe close to a USB-cable with termination

In Fig. 2, one recognizes the almost ideal behavior for lower frequencies and the resonance effects at higher frequencies. One may idealize the transfer characteristic by the given piece-wise linear shape. For our studies, we also tried zero-forcing equalization with that given idealization, since it was clear that adjusting to the probe type and the varying properties dependent on the position of the probe is not feasible. Equalization with the idealized characteristic, however, was not showing a good performance. Hence, we kept the high-pass characteristic of the probe for our studies, despite of its variations.

IV. Time-Frequency Analysis Methods

For analyzing the signals, we looked at some different time-frequency analysis tools that are available. This allows us to not only be able to do some pre-processing on our data using these tools but also allows us to convert our data from time-series into pictorial representations. This is helpful for our neural network approach in particular as neural networks are remarkably good at image classification and this would make the problem into an image classification problem [22]. Let us briefly mention some of the time-frequency analysis tools that we considered.

A. Short Time Fourier Transform

One of the time-frequency analysis tools we considered is the Short-Time Fourier Transform. The short-time Fourier transform (STFT), is used to find the sinusoidal frequency and phase content of different local sections of the data as it changes in time [23]. Practically, this means that the longer signal is divided into different equal short sections. Then, the Fourier transform of these short sections is computed separately. This gives us the Fourier spectrum on each of the short sections. One then usually plots the so called waterfall plot in which this changing spectrum is plotted as a function of time. The formula for the continuous time STFT is a shown below where $w(\tau)$ is the window function used to cut the longer signal into sections.

$$\mathbf{STFT}\{x(t)\}(\tau,\omega) \equiv X(\tau,\omega) = \int_{-\infty}^{\infty} x(t)w(t-\tau)e^{-j\omega t}\,dt \tag{1}$$

B. Wavelet Transform

Similar to how in Fourier analysis, one decomposes a signal into sinusoidal waves of specific frequencies, in wavelet analysis, one decomposes a signal into shifted and scaled versions of a wavelet [24]. The continuous wavelet transform (CWT) of a function $x(t)$ at a scale $(a > 0)$ $a \in \mathbb{R}^{+*}$ and translational value $b \in \mathbb{R}$ is expressed by the integral

$$X_w(a,b) = \frac{1}{|a|^{1/2}} \int_{-\infty}^{\infty} x(t)\overline{\psi}\left(\frac{t-b}{a}\right) dt \tag{2}$$

C. Wigner-Ville Distribution

The Wigner-Ville Distribution also know as the Wigner-Ville Spectrum is functionally similar to a spectrogram. It gives better frequency and temporal resolution but this comes at the cost of introducing new artifacts [25]. For a signal $s(t)$ with analytic associate $x(t)$, the Wigner-Ville Distribution $W_x(t,\omega)$ is defined as

$$W_x(t,\omega) = \int_{-\infty}^{\infty} x(t+\tau/2)x^*(t-\tau/2)e^{-j\omega\tau}\,d\tau \tag{3}$$

V. Some Time-Frequency Analysis Signal Representations

Let us look at some time-frequency analysis signal representations of our chosen PC-typical signals. We show here the STFTs of some of the wave forms chosen.

We picked STFT as our pre-processing tool for the neural network approach as the pictures from the STFT seemed the most distinct between each other. Moreover, they also can be simplified into small, low-resolution pictures for simplification while still remaining easily distinguishable. We went for one-sided, two-dimensional, gray-scale versions of the STFTs. One of these is shown in Fig. 8.

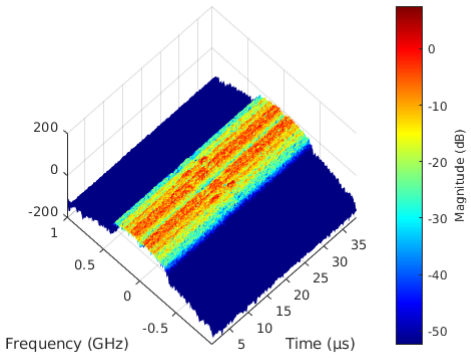

Fig. 3. STFT HDMI

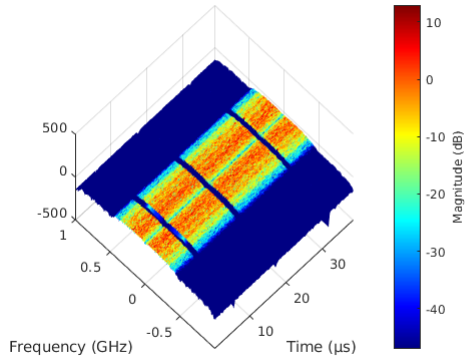

Fig. 4. STFT USB

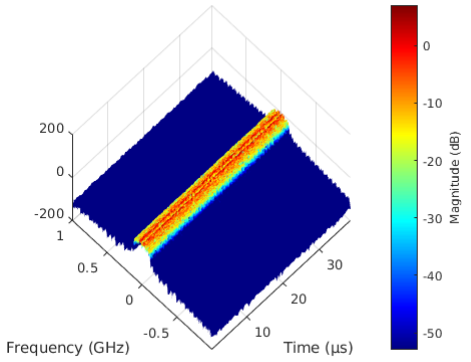

Fig. 5. STFT Eth100

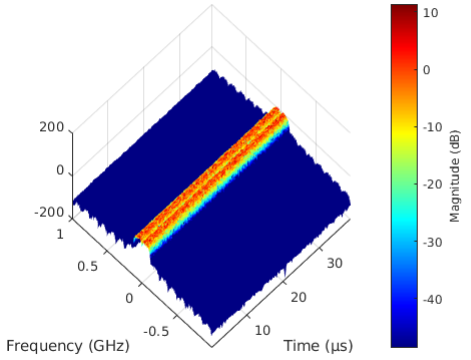

Fig. 6. STFT Eth1000

VI. Neural Network classification

We selected a Deep Convolution Neural Network (CNN). They are typically used for image recognition and classification. The neural network was trained on STFT pictures of the stimulated signals. Our network has around 50 layers which has a total of 7,164,349 different parameters. Out of these 7,155,549 are trainable, leaving 8,800 non-trainable parameters.

Our Neural Network is able to achieve 100 % accuracy for simulated noise free data, even for newly simulated data which the network has not seen beforehand. When we consider data with added noise we get some loss in accuracy. A plot of the accuracy percentage vs. the SNR is shown in Fig. 9. The procedure is still able to work well at very high noise levels. It needs an SNR of worse than -7 dB to get down to an accuracy lower than 65%. We have not yet compared to simple waveform training, but expect superior performance with our preprocessing.

VII. Signal-specific analysis

In this section, we describe the individual characteristics of the investigated signals. As these characteristics are unique to a signal, they allow for identification of the signal type based on the rates, synchronization patterns, or encoding scheme.

A. Universal Serial Bus Synchronization Pattern

Preceding every data packet, a synchronization sequence is sent, as described in Section II-A. In a case of direct access to the wires, a signal can be identified as USB by comparing the results of a correlation with the synchronization sequences. Due to the significant difference in the length of the synchronization sequences, it is possible to distinguish between full-speed and high-speed USB.

The sync pattern does not contain a common-mode component, which can also be used for distinction. Nevertheless, although altered by the transfer characteristic of the probe, the synchronization pattern is visible in the measurements, and hence, USB is also identifiable by its synchronization sequence. The predominant characteristic of both synchronization patterns is the switch from idle to K state followed by a period of alternating states, thus voltage levels. These predominant factors allow to determine the position of the synchronization sequence. The start of the sequence is located by finding the position of the switch from idle to the K state. Afterwards, the maxima and minima of the signal are determined for each symbol period following the start of the sequence, according to the length of the synchronization pattern. By comparing the positions of positive and negative peaks of the synchronization pattern, one can determine the approximate rate of the signal, which allows for the identification of the signal.

Using the synchronization pattern to approximate the rate

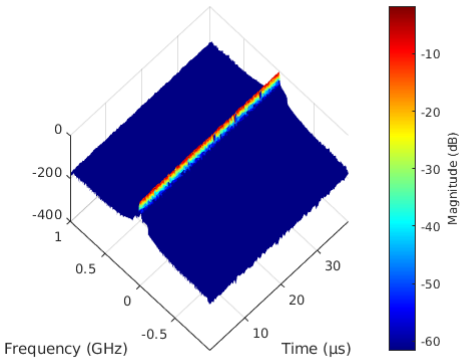

Fig. 7. STFT USB Full

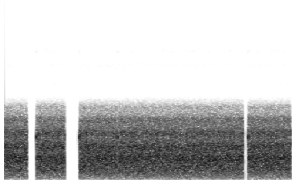

Fig. 8. STFT USB High

makes this approach quite robust, as the synchronization sequence is quite unique to the USB signals and the difference in the length of the sequence for the two types of USB, as well as the difference in the rate and hence symbol period are quite significant.

B. High-Definition Multimedia Interface Preamble

As explained in Section II-B, depending on the succeeding period, different preambles are sent. If there is direct access to the wires, the correlation method used for USB is also applicable to HDMI. Due to the relatively short length of the preamble, the pattern can also occur in the data resulting in correlation peaks at wrong locations. Consequently, it is necessary to compare the correlation results of channels 1 and 2 as the preamble is sent simultaneously on both channels. This is only possible for direct access to the separate channels.

Considering measurements done by the probe, the start of a preamble is not as easily located as for USB because there is no significant change like the switch from idle to a voltage state. This is due to the fact that the preamble is solely used to indicate the type and start of a data island or video data period as there is no requirement for clock synchronization at the receiver side because the clock is transmitted over a separate wire. Consequently, the approach of determining the rate as used for USB is not possible. Nevertheless, one can approximate the rate of the signal by identifying a rising edge and the succeeding minimum and maximum and computing the differences in time between them. If the measurement of the signal contains enough samples, this allows for an estimation of the rate which can thus identify the signal to

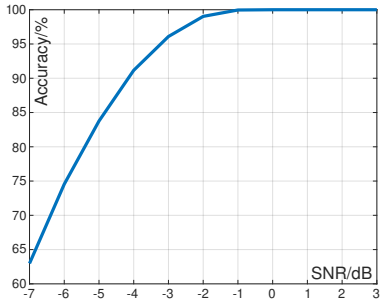

Fig. 9. Performance of neural network over different SNRs

be HDMI. However, this method is not as accurate as using a synchronization pattern to compute the rate.

C. Ethernet Amplitude Histograms

For 100Base-T it is technically possible to locate the start-of-stream delimiter (SSD) followed by the preamble if there is direct access to the signal. Synchronization is done by identifying the rising edges of the signal and matching the clock to those. This is followed by MLT-3 decoding and descrambling. Using the same approach as for USB, when there is direct access, a correlator is used to identify the SSD followed by the preamble, thus identifying the signal as 100Base-T Ethernet.

This approach to detect those sequences in 100Base-T is not successful for measurements done by the probe due to the influence of the probe and other factors resulting in noise causing the decoding and descrambling to become inaccurate. Furthermore, the line code of the 1000Base-T as described in Section II-C requires direct access to all four wire pairs for decoding. Consequently, it is not feasible to identify Ethernet 100Base-T and 1000Base-T by locating the SSD. Nevertheless, the approximation of the rate by identifying rising edges and the minima and maxima in the following two symbol periods and computing the differences between them, similarly to the USB approach, is possible. However, this is not as accurate as for USB because there is no synchronization pattern used to determine the rate. Due to the significant difference of the rates of the signals, this is, however, enough to identify the signal to be one of the two Ethernet types. This allows for synchronizing by locating the rising edges and matching a reference clock to them. While the rate of 100Base-T and 1000Base-T is the same, their encoding differs in the number of voltage levels used. To compute their amplitude histograms, for the detection instant, synchronization is essential, which also means the differentiation (HP characteristic) and additional delay caused by the probe and connection have to be taken into account.

One can differentiate the amplitude histograms of the two types of Ethernet by determining Gaussian mixture models using the Aike and Bayesian Information Criteria which can be seen in figures 10 and 11. While the three levels of the encoding scheme for 100Base-T are more common, the five levels of the 1000Base-T encoding make it distinguishable from other signals.

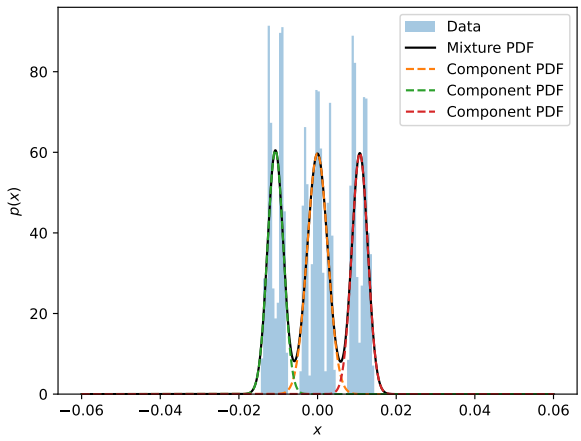

Fig. 10. 100Base-T amplitude histogram including Gaussian mixture models

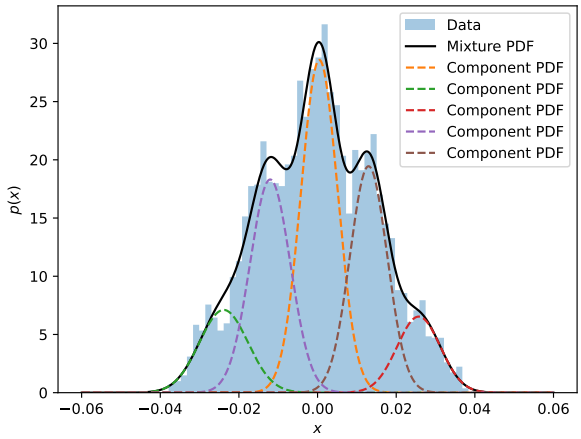

Fig. 11. 1000Base-T amplitude histogram including Gaussian mixture models

VIII. Conclusions

We found that typical baseband PC signal emissions measured by an H-field probe can easily be classified and major blocks can be localized using time-frequency preprocessing and a standard convolutional neural network applied to resolution-reduced pictures. Likewise, characteristic signal properties can be used to recognize them and synchronize to them.

For many signals this does not appear too surprising. However, it actually is when looking at spectra and time-frequency representations of Ethernet 100Base-T and 1000Base-T. For a human eye, they are looking essentially the same, but the neural network does not lead to any false detection. We are currently working on measured data and will soon present results in a further publication. Accuracy looks very promising there, as well.

References

[1] "An introduction to TEMPEST." https://www.sans.org/reading-room/whitepapers/privacy/introduction-tempest-981. Accessed: Oct. 13, 2017.

[2] Compaq et al., "Universal serial bus specification, rev. 2.0," 2000.

[3] IEEE Standards Association, "IEEE standard for Ethernet 802.3," *IEEE Computer Society*, 2018.

[4] Hitachi et al, HDMI Licensing, LLC, "High-definition multimedia interface specification," 2006.

[5] G. Giannakis, "Cyclostationary signal analysis," in *The Digital Signal Processing Handbook, Electrical Engineering Handbook* (D. B. W. Vijay K. Madisetti, ed.), pp. 62–17, CRC Press LLC, 1997.

[6] M. Mühlhaus, *Automatische Modulationsartenerkennung in MIMO-Systemen*. Forschungsber. Institut für Nachrichtentechnik, KIT, Karlsruhe, Inst. für Nachrichtentechnik, 2014.

[7] T. Nawaz, L. Marcenaro, and C. S. Regazzoni, "Stealthy jammer detection algorithm for wide-band radios: A physical layer approach," in *STWiMob*, pp. 79–83, Oct 2017.

[8] O. Dobre, A. Abdi, Y. Bar-Ness, and W. Su, "Survey of automatic modulation classification techniques: classical approaches and new trends," *IET Communications*, vol. 1, pp. 137–156(19), April 2007.

[9] L. Zhou and H. Man, "Distributed automatic modulation classification based on cyclic feature via compressive sensing," in *IEEE MILCOM*, pp. 40–45, Nov 2013.

[10] Y. Liu, "Modulation classification of MIMO-OFDM signals by independent component analysis and support vector machines." https://arxiv.org/pdf/1307.4430. ArXiv 2013.

[11] L. Han, H. Xue, F. Gao, and Z. Li, "Low complexity automatic modulation classification based on order statistics," in *IEEE VTC*, pp. 1–5, Sept 2016.

[12] L. Han, F. Gao, Z. Li, and O. A. Dobre, "Low complexity automatic modulation classification based on order-statistics," *IEEE Tr. on Wireless Communications*, vol. 16, pp. 400–411, Jan 2017.

[13] L. Breiman, "Random forests," *Machine Learning*, vol. 45, pp. 5–32, Oct 2001.

[14] A. Ali and F. Yangyu, "k-sparse autoencoder-based automatic modulation classification with low complexity," *IEEE Commun. Letters*, vol. 21, pp. 2162–2165, Oct 2017.

[15] T. Nawaz, D. Campo, M. O. Mughal, L. Marcenaro, and C. S. Regazzoni, "Jammer detection algorithm for wide-band radios using spectral correlation and neural networks," in *IWCMC*, pp. 246–251, June 2017.

[16] A. Swami and B. Sadler, "Modulation classification via hierarchical agglomerative cluster analysis," in *First IEEE Signal Processing Workshop on Signal Processing Advances in Wireless Communications*, pp. 141–144, April 1997.

[17] Z. Sun, S. Wang, and X. Chen, "Feature-based digital modulation recognition using compressive sampling," *Mobile Information Systems*, 2016.

[18] T. Nawaz, L. Marcenaro, and C. S. Regazzoni, "Defense against jamming attacks in wide-band radios using cyclic spectral analysis and compressed sensing," in *ICUFN*, pp. 874–879, July 2017.

[19] D. Baudry, C. Arcambal, A. Louis, B. Mazari, and P. Eudeline, "Applications of the near-field techniques in EMC investigations," *IEEE Tr. on Electromagnetic Compatibility*, vol. 49, pp. 485–493, Aug 2007.

[20] Y. Liu and B. Ravelo, "Fully time-domain scanning of EM near-field radiated by RF circuits," *Progress In Electromagnetics Research*, vol. B57, pp. 21–46, January 2014.

[21] G. S. Matthias Spang, Manfred Albach, "Calibration of probes for EMC near-field scanning," in *CST User Group Meeting 2010*, 2010.

[22] Yilin Sun ,Edward A Ball, "Automatic modulation classification using techniques from image classification," 2022.

[23] L. S. E. Sejdić, I. Djurović, "Fractional Fourier transform as a signal processing tool: an overview of recent developments," *Digital Signal Processing: A Review Journal*, vol. 19, no. 2, pp. 153–183, 2009.

[24] L. Debnath and J.-P. Antoine, "Wavelet transforms and their applications," *Physics Today - PHYS TODAY*, vol. 56, pp. 68–68, 04 2003.

[25] C. Chioncel, P. Chioncel, N. Gillich, and O. Tirian, "Wigner Ville distribution in signal processing, using Scilab environment," *Analele Universitatii 'Eftimie Murgu' Resita*, vol. XVIII, pp. 101–106, 01 2011.