Signal Area Estimation based on Deep Learning

Mohammed M. Alammar
Department of Electrical Engineering and Electronics
University of Liverpool
Liverpool, United Kingdom
King Khalid University, Abha, Saudi Arabia
M.M.Alammar@liverpool.ac.uk

Miguel López-Benítez
Department of Electrical Engineering and Electronics
University of Liverpool
Liverpool, United Kingdom
ARIES Research Centre, Antonio de Nebrija University, Spain
M.Lopez-Benitez@liverpool.ac.uk

***Abstract*—In many practical application scenarios, radio communication signals are commonly represented as a spectrogram, which represents the signal strength measured at multiple discrete time instants and frequency points within a specific time interval and frequency band, respectively. In the context of spectrum occupancy measurements, the notion of Signal Area (SA) is defined as the rectangular region in the time-frequency domain where a signal is assumed to be present. Signal Area Estimation (SAE) is an important functionality in spectrum-aware wireless systems where spectrum usage monitoring is required. However, the conventional approaches to SAE have a limited estimation accuracy, in particular at low SNR. In this work, a novel technique for SAE is proposed using Deep Learning based on Artificial Neural Network (DL-ANN) for enhanced extraction of SA information from radio spectrograms. The performance of the proposed DL-ANN method is evaluated both with software simulations and hardware experiments, and the results are compared with several conventional methods from the literature, showing significant performance improvements. A key feature of the proposed method is the improvement in the SAE accuracy compared to other existing methods (in particular in the low SNR regime) and the capability to extract the location of the detected SAs automatically. Overall, the proposed technique is a promising solution for the automatic processing of radio spectrograms in spectrum-aware wireless systems.**

***Index Terms*—Spectrum awareness, signal area estimation, deep learning, artificial neural network.**

I. Introduction

In many practical application scenarios, radio communication signals are commonly represented as a spectrogram, which represents the signal strength measured at multiple discrete time instants and frequency points within a specific time interval and frequency band, respectively. Radio spectrograms are used for time-frequency signal analysis in spectrum-aware systems for many purposes, including automatic blind modulation classification (with heuristic algorithms [1]–[3] and convolutional neural networks [4]–[6]), radio technology identification [7], interference detection and mitigation [8], detection and localisation of radio events [9], radio signal denoising [10], extraction of frequency hopping signal parameters [11], [12], spectrum sensing [13], detection of radar signals [14] and characterisation of the Signal-to-Noise Ratio (SNR) and Doppler shift [15].

An important aspect in the processing of radio spectrograms is the region that each individual radio transmission or signal component occupies in the time-frequency domain within the spectrogram, which in this work is referred to as the Signal Area (SA). A SA is defined as a cluster of spectrogram points within a rectangular shape where a transmitted signal component is assumed to be present. Thus, each SA detected in a spectrogram precisely determines the occupied bandwidth and the start/end times of each individual radio transmission. The capability to obtain this information accurately from a radio spectrogram can be useful in many practical applications, which include spectrum surveillance (both for enforcement of spectrum regulations and gathering of signal intelligence in military applications), radio signal interception and identification, electronic warfare and radio environment spectral awareness (for instance, in databases for spectrum sharing systems). Consequently, the process of Signal Area Estimation (SAE), which entails determining the subsets of spectrogram points that belong to one or more SAs, is an important function is spectrum-aware wireless systems.

The interest of this work is in how to accurately determine the SAs present in a radio spectrogram obtained from radio spectrum measurements and extract the information about the coordinates of each SA automatically. Several methods have been proposed in the literature in order to achieve this end, however many of such methods have some limitations, which include one or more of the following issues: are based on heuristic principles, involve a number of configuration parameters that need to be tuned individually for each operation scenario, offer a poor performance in the low SNR regime, or are unable to extract automatically the coordinates of each SA present in a spectrogram (which is useful for automatic spectrogram processing in autonomous spectrum-aware wireless systems). To address these issues, this work proposes a novel approach for SAE based on the use of widely known and well-developed deep learning techniques.

The field of Artificial Intelligence (AI) has experienced a dramatic development associated with past and recent advances in the areas of Machine Learning (ML) methods in general and Deep Learning (DL) techniques in particular. DL is a group of ML techniques that allows computers to learn and discover complex patterns in large datasets automatically, taking inspiration from the human brain [16]. DL relies on Artificial Neural Networks (ANNs), which consist of layers of nodes (neurons) and synapses inspired by the human brain. A typical ANN is a simplified computation model with

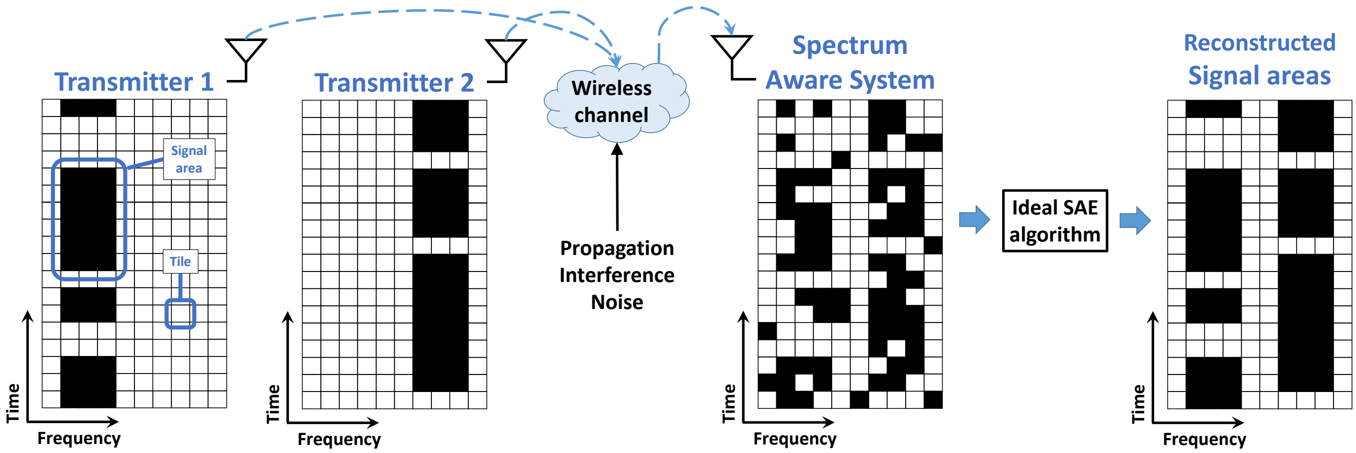


Fig. 1: Illustration of the concept of Signal Area (SA) and system model for Signal Area Estimation (SAE).

multiple interconnected layers of neurons [16]. The network is trained to produce useful predictions based on identification of patterns in training data. One area were DL techniques based on ANNs have gained widespread application is in communication systems [16]–[18]. However, the application of DL techniques to the particular problem of SAE has received very limited attention in the existing literature. In this context, this work proposes a novel approach for SAE that relies on the use of Deep Learning based on Artificial Neural Network (DL-ANN). The proposed method addresses the limitations of existing SAE methods in the literature as mentioned above. The performance of the proposed DL-ANN method for SAE is assessed both with software simulations and hardware experiments, showing that the proposed method provides an accurate SAE performance – with significant improvements in the low SNR regime – and offers the interesting feature of automatically extracting the coordinates of each SA detected in a radio spectrogram obtained from spectrum measurements.

The main contributions of this work are summarised below:

- A novel approach for SAE based on DL-ANN is proposed. The main advantage of the proposed method is its ability to overcome the main limitations of existing SAE methods. In particular, the proposed method can extract automatically the coordinates of each SA present in a spectrogram, which is a useful feature for automatic spectrogram processing in autonomous spectrum-aware wireless systems. Moreover, it does not require any human intervention for its configuration or operation and provides substantial performance improvements in the low SNR regime compared to existing SAE methods.
- In addition to optimising the network architecture dimensions and hyperparameter configuration, several variants of the proposed DL-ANN approach are investigated, including options with one and two input layers (including not only the binary spectrogram but also the estimated SNR at the receiver) as well as different amounts of available training data, in order to determine the most convenient design and optimisation of the proposed approach.
- Given the analogy of the SAE problem with image processing techniques for the detection of rectangles in noisy images, the performance of the proposed DL-ANN when combined with two image processing techniques (namely morphological operations and edge detection plus flood fill) is evaluated as well. These image processing techniques are combined with the proposed DL-ANN method as pre/post-processing stages.
- The performance of the proposed DL-ANN method is assessed both with software simulations and hardware experiments, and compared with relevant SAE methods from the literature. The analysis includes not only the estimation accuracy of the detected SAs and the computational cost but also the impact of the spectrogram resolution and channel fading. The obtained results show that the proposed method provides an accurate SAE performance with significant improvements over existing SAE methods, in particular in the low SNR regime.

The remainder of this work is organised as follows. First, Section II provides a description of the SAE problem addressed in this work and an overview of the existing SAE methods. Then, Section III presents the DL-ANN method proposed in this work based on the use of DL techniques. The evaluation methodology followed to assess the performance of the proposed method is outlined in Section IV, while the obtained simulation and experimental results are presented and discussed in Section V. Lastly, Section VI summarises the findings of this work and draws the main conclusions.

II. Signal Area Estimation

A. Problem Description and Formulation

The problem being addressed in this work is how to accurately identify the dimensions of each SA present in a spectrogram and automatically extract such information for later processing in spectrum-aware systems. In radio communication signals, the spectrogram is a discrete two-dimensional (time vs. frequency) representation of the power observed at various discrete time instants and discrete frequency points. These power levels that can be observed by means of spectrum measurements can be compared to a certain decision threshold (which can be set according to several criteria [19]) to produce a binary version of the spectrogram where the value of each point can be either zero (meaning not occupied, if the power level at that point is below the threshold) or one (meaning occupied, if the power observed at that point is greater than the threshold). Such binary matrix is the input information used by SAE methods to attempt to estimate the presence of SAs as accurately as possible. Each individual radio transmission or signal component that is present in a radio spectrogram is by definition contained inside a rectangular area within the spectrogram (the SA) that defines precisely the frequency bandwidth occupied by that signal as well as the start and end transmission instants[1]. The main aim of SAE methods is to identify the dimensions of each SA as accurately as possible. The process of SAE, however, is not straightforward given that the spectrum data observed by means of experimental spectrum measurements are a degraded version of the transmitted signals after having suffered impairments introduced by radio propagation, the receiver noise and other external sources of unwanted noise and interference (e.g., out-of-band transmissions, ambient noise or man-made noise) [22]–[27]. The aim of this work is to accurately determine the time-frequency region occupied by each radio transmission in a spectrogram (i.e., the SAs present in a radio spectrogram). The overall concept and system model are shown in Fig. 1.

SAE methods rely on the output of spectrum sensing (signal detection) techniques. While there is a strong connection between the two signal processing steps, it is important to make a clear differentiation between both techniques since the ultimate purpose of SAE is different from that of spectrum sensing. Signal detection (spectrum sensing) techniques are aimed at accurately detecting the instantaneous presence or absence of a signal in a certain frequency band and at a particular time instant. The final output of signal detection techniques is therefore a binary decision on whether a signal is believed to be present in a set of digital samples. In this work, a signal detection stage is applied to power spectrograms in order to decide the binary idle/busy state of each spectrogram point. This step is necessary as a pre-processing stage since it produces the binary spectrogram that in this work is fed as an input to the SAE method. Based on that binary spectrogram, the SAE method will then attempt to estimate as accurately as possible the number of SAs present in the spectrogram and their respective dimensions, which is the final output information provided by SAE methods. Because SAE methods rely on the output of spectrum sensing (signal detection) techniques, their accuracy is conditioned by the performance of the employed signal detection technique and hence they are also vulnerable to spectrum sensing errors (i.e., missed detections and false alarms). However, the impact of such errors has a different level of severity on the output of each signal processing stage. While sensing errors can make a signal detection method fail completely by providing an incorrect idle/busy output decision for a particular spectrogram point, SAE methods may still be able to perform their task reasonably well in the presence of sensing errors (i.e., spectrogram points detected in an incorrect state by the employed signal detection method). SAE is concerned with an accurate estimation of the overall SA, hence the accuracy of every individual spectrogram point is not relevant per se as long as the overall SA can be estimated accurately, which in many cases can be possible under a moderate number of individual sensing errors. This is because the aim of SAE methods is to establish the time-frequency region occupied by each SA rather than the instantaneous signal presence in each time-frequency point of the spectrum, which is the purpose of signal detection methods. Moreover, signal detection methods are usually aimed at providing real-time decisions on the instantaneous spectrum occupancy state and this information is normally used for short-term decisions (i.e., transmit or vacate the channel immediately), while SAE methods are usually not envisaged to be applied in real-time (which would not be possible due to the time span needed to capture the amount of data required to complete a spectrogram). SAE methods are typically employed for offline processing of spectrum occupancy data and its characterisation in a longer-term. The information obtained from SAE is typically useful for optimising spectrum and radio resource management in the longer-term. Therefore, in SAE the focus is on determining the spectrum occupancy pattern of spectrum users in the time-frequency domains and in a medium to long term. SAE methods rely on the output of signal detection methods but have a different purpose and objective.

B. Existing SAE Methods

The most simple form of SAE could be considered to be a simple Energy Detector (ED), which outputs the on/off state of each point in a spectrogram based on the comparison of each power level to a decision threshold. However, while simple and convenient, ED produces no rectangular estimation of the SAs in the spectrum and therefore cannot be considered a SAE method; nevertheless, ED will be used in this work as a useful reference baseline for comparison with other existing SAE methods (i.e., methods that can provide rectangular estimations of each SA detected in a radio spectrogram).

Several SAE methods have been proposed in the literature based on various approaches. The work reported in [28] presents a computer vision approach based on the application of a fixed threshold to the spectrogram in order to generate a binary image, which is also processed using morphological operations as an adaptive threshold approach to remove extraneous detections, and finally extracts the image blobs by grouping connected components and calculating their bounding boxes. Such method is modified in [29] by introducing an

[1]Certain radio emissions may not lead to rectangular SAs, in particular electromagnetic emissions from systems that are not intended for wireless communications, including some types of radars [14, Figs. 1 and 9], microwave ovens [20, Fig. 2b], and several sources of man-made noise [21, Fig. 4]. Such specific signal formats require a tailored study that is beyond the scope of this work, whose focus is on wireless communication signals, usually characterised in radio spectrograms by rectangularly-shaped SAs.

auto-thresholding method and a bi-directional self-organising network in order to reduce noise after thresholding. In [30], the use of a network based on a single shot multibox detector [31] is proposed for signal component extraction, which is further extended in [32] by introducing convolution layers in order to provide a more accurate detection at the expense of an increased complexity. A different approach based on the Mean-Shift Clustering (MSC) algorithm is suggested in [33], where each SA is determined based on the use of a scanning window whose dimensions are adjusted according to the expected bandwidth and transmission duration of the signal components to be detected. A Transmission Encapsulation based on the Connected Component Labelling (TECCL) method is proposed in [34], which performs clustering based on the connected component labelling algorithm [35] and estimates the SA of each cluster as its extreme dimensions (bounding box). This method can be implemented using contour tracing techniques [36] (see CT-SA in [37] for instance). A so-called Simple Signal Area (SSA) estimation method is proposed in [37], which performs a raster scan to find the first corner of each SA, followed by horizontal scanning to estimate the SA width and coarse/fine vertical scanning to estimate the SA height. Some variants to reduce the impact of false alarms are proposed in [38]–[40]. An approach based on mathematical morphology principles is proposed and evaluated in [41].

III. Proposed DL-ANN Method for SAE

A. Motivation

The basic problem in SAE is how to detect and estimate a rectangular grid representing a solid SA in a time-frequency matrix (spectrogram) of degraded power values (affected by noise and impaired by the propagation channel). To facilitate the problem, the continuous-domain power levels are thresholded in order to produce a binary matrix with zero/one values indicating the idle/busy state of each point in the power spectrogram (i.e., absent/present signal component). The problem of estimating a solid SA in such binary matrix has some analogies with the problem of recognition of patterns in a noisy black-and-white image. Therefore, image processing principles aimed at detecting objects in images can be employed to address the problem considered in this work by treating the spectrogram of power values as a greyscale image or its binary version as a black-and-white image (the latter case is the one considered in this work), where each spectrogram time-frequency point represents an image pixel. The problem of SAE then becomes the problem of detecting rectangular shapes (i.e., SAs) in a binary noisy image. This viewpoint offers a new perspective for SAE that enables the application of a broad range of tools from the field of image processing [42] to the problem of SAE. For example, the recent work reported in [43] has investigated the application of the Hough transform to detect the rectangular shapes of SAs in noisy radio spectrograms, while [44] has investigated a combined approach based on edge detection (used to estimate the edges of the potential SAs in a spectrogram) and flood fill (used to fill the area inside the detected edges to produce solid SAs) to address the same problem. While traditional image processing techniques can provide an interesting approach to the problem of SAE, more recent approaches from the field of computer vision and pattern recognition [45], many of which make use of AI/ML/DL techniques, provide more advanced methods to address the problem considered in this work from the point of view of object detection and recognition. This motivates this work to explore the suitability of using DL techniques to address the problem of SAE.

It is worth noting that, even though the use of DL techniques for object detection (e.g., see [46]), and in particular for the detection of rectangular objects in an image, is not new (e.g., see [47]) the problem of detecting rectangular SAs in a radio spectrogram in the context of SAE has some particular characteristics that require special consideration and hence a tailored study as the one presented in this work. Concretely, the problem of SAE has some specific properties, including the fact that the rectangular objects to be detected (i.e., the SAs) are aligned with the horizontal and vertical axes of the spectrogram (in other words, the SAs do not have any rotation) and moreover the SAs do not overlap among them (overlapping SAs would mean harmful interference between radio transmissions and most radio communication systems are indeed engineered in order to explicitly avoid such scenario). In general, these two features would mean that the problem of SAE might be seen as a simplified version of the general problem of detecting rectangular objects in a noisy image. However, the detection of SAs in a spectrogram is particularly challenging due to the degrading effects introduced by the radio propagation channel and the receiver noise. These two degrading effects will lead to the appearance of random false alarms (i.e., time-frequency points where a signal component is not present but its power level is observed above the detection threshold due to increased noise) and random missed detections (i.e., time-frequency points inside SAs that are observed below the detection threshold due to a power reduction caused by the radio propagation channel). These two types of degradations, as it will be shown, can distort significantly the SAs present in a radio spectrogram and their rectangular shapes to the extent that they may become undistinguishable from the background noise of the receiver and therefore unrecognisable, in particular when the radio signals are received at very low SNR (where SAE becomes extraordinarily challenging). This implies that addressing the problem of SAE from the point of view of detecting rectangular objects in an image requires a specific and tailored analysis, which motivates the study presented in this work.

B. Proposed SAE Approach with DL based on ANN

DL represents a category of ML methods that use a number of deep layers to transform and process raw input data in order to extract relevant features. DL models are commonly based on ANNs, in which model learning can be supervised, unsupervised, or semi-supervised [48]. ANNs are inspired by the working of the human brain to process information. With the advances in DL techniques, ANNs have gained widespread

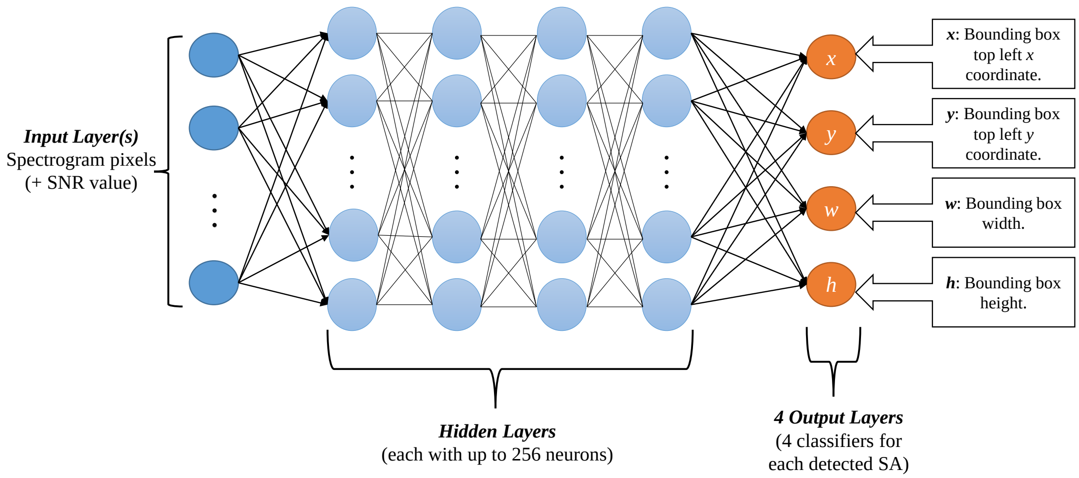


Fig. 2: DL-ANN network architecture for SAE.

application in digital image processing [48], [49]. However, the application of DL techniques in general, and ANNs in particular, to the specific problem of object detection in the context of SAE has received little attention. In this context, this work proposes a novel approach for SAE that relies on the use of a DL-ANN approach for SAE. The proposed approach relies on a standard Multi-Layer Perceptron (MLP) network for SAE, which is a class of feedforward ANN composed of multiple layers of perceptrons (neurons that use non-linear activation functions), including an input layer, one or more hidden layers, and a final output layer. MLPs utilise a supervised learning technique called backpropagation for training, where the network is trained using labelled input data to help recognise the corresponding correct output.

The proposed DL-ANN model is depicted in Fig. 2. As discussed earlier, the main input information is a binary spectrogram, which is composed of zero/one values indicating where a signal component is observed above a given power threshold. This binary spectrogram is obtained by thresholding a power spectrogram (obtained by the receiver from spectrum measurements) with a properly set decision threshold. Several methods to set such threshold in the context of SAE were investigated in [19], where it was concluded that a threshold set for a low Constant False Alarm Rate (CFAR), such as 0.01, provides a good performance in SAE. Two different scenarios are considered regarding the input information provided to the DL-ANN. In the first scenario, only the binary spectrogram obtained from thresholded spectrum measurements is provided as input information, while in the second scenario the SNR value at which the provided spectrogram was generated[2] is also fed to the network as a second input parameter. It is well-known that ANNs can learn better when additional input information is provided (assuming that the input parameters are independent), therefore the approach considered in this second scenario is investigated to determine whether it can provide better performance than the more simple approach proposed in the first scenario. The main drawback of this second approach in a practical system implementation is that the SNR needs to be estimated and any inaccuracies in such estimation may also affect the accuracy of the SAE process. However, this extra cost may be worth if it enables a more accurate SAE, which makes the consideration of this second approach interesting.

[2]This work assumes, for performance evaluation purposes and without loss of generality, that all the SAs in the same spectrogram have the same SNR. This approach will allow determining in a clear and unambiguous manner the relation between the performance of the proposed SAE method and the SNR of the signals present in the spectrogram.

The output information provided by the DL-ANN needs to identify unambiguously the location and dimensions of each SA detected in the input spectrogram. To this end, four output parameters are considered for each SA as illustrated in Fig. 2:

- x: Abscissa of the SA's top-left corner.
- y: Ordinate of the SA's top-left corner.
- w: SA's width.
- h: SA's height.

The tuple (x, y, w, h) univocally identifies the bounding box within which each SA is contained and therefore provides sufficient information to unambiguously characterise each detected SA. Notice that this output information means that the DL-ANN is able to automatically extract the relevant information related to each detected SA. The DL-ANN will detect automatically the number of SAs present in the spectrogram provided as input and will provide one tuple for each detected SA with the corresponding information of the bounding boxes.

The complete definition of the proposed DL-ANN involves the specification of the network architecture (number of hidden layers and neurons per layer), the tuning/optimisation of the network hyperparameters and the used training procedure. These aspects are discussed in more detail in the following section as part of the methodology followed in this study.

IV. Methodology

In order to train, validate and test the proposed DL-ANN method, spectrum occupancy data were generated by means

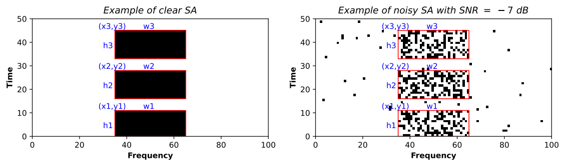


Fig. 3: Example of a randomly generated time/frequency test grid: (a) Clean test grid, (b) Test grid with noise (SNR = –7 dB).

of both software simulations and hardware experiments. This section provides details of the methodology employed in both cases as well as how the proposed DL-ANN was optimised.

A. Simulation Procedure

Simulations were performed following the same procedure employed in [19], [41], which is discussed below:

Step 1. *Creation of the time-frequency test grids:* In the first step, rectangular time-frequency grids (representing the measured spectrograms) were created using a predefined time-frequency resolution. In this work, a resolution of 50×100 points was selected, which can be considered as a medium spectrogram resolution [19]. The vertical value (50 points) represents the time resolution while the horizontal value (100 points) represents the frequency resolution of the spectrogram. The total frequency span of the spectrogram was divided into three channels with equal bandwidth (1/3 of the spectrogram width with 5% reserved for guardbands). However, for simplicity and clarity of graphical representation, only the central channel was assumed to carry actual data traffic[3], which was modelled as a sequence of on/off transmissions randomly drawn from exponential distributions with a minimum duration of 10/5 spectrogram points, respectively, and rate parameter 0.5 points^{-1} (since transmission durations are generated in terms of the number of spectrogram points, the rate parameter of the corresponding exponential distribution has units of points^{-1}). The bounding boxes of each generated SA were saved as labels for training. Fig. 3a shows a sample test grid containing three SAs and their corresponding bounding boxes.

Step 2. *Addition of signal detection errors to the test grid:* For each test grid (spectrogram) generated in Step 1, random errors were generated and introduced to emulate the errors suffered by practical signal detection (spectrum sensing) methods. This step aims to emulate the degrading effects of the channel propagation (by introducing random misdetections inside SAs) and the receiver noise (by introducing random false alarms outside SAs). In this process, points inside SAs can change from busy to idle state with a random probability of misdetection (P_{md}) while points outside SAs can change from idle to busy state with a random probability of false alarm (P_{fa}). The value of these probabilities depends on the criterion employed to set the energy/power decision threshold. Following the findings of the study presented in [19], a CFAR criterion with $P_{fa} = 0.01$ was assumed, which results in an SNR-dependent value of P_{md}. The relation between the detection probability $P_d = 1 - P_{md}$ and the experienced SNR is given by [50, eq. (7)] for an Additive White Gaussian Noise (AWGN) channel and by [51, eq. (22)] for a Rayleigh fading channel[4]. As stated in footnote 2, all the SAs in the same spectrogram are assumed to have the same SNR and therefore are subject to the same probability of misdetection. For a detailed performance evaluation, an SNR range from –20 dB to +5 dB was considered, and simulations were repeated for each individual SNR value within that range in 1-dB increments. Fig. 3b shows how the spectrogram shown in Fig. 3a would be observed at the receiver according to the employed simulation procedure when the SNR at the receiver is –7 dB (at which $P_{md} \approx 0.61$), which would be the input spectrogram available for SAE.

Step 3. *Application of the considered SAE method:* This step involves the application of a SAE method to the test grid (spectrogram) obtained from Step 2 in order to attempt to estimate the SAs present in the spectrogram. The proposed DL-ANN method presented in Section III was applied in this step. For comparison purposes, the CT-SA and SSA methods presented in Section II-B were also evaluated and used as a reference benchmark; these two methods were selected as a benchmark for comparison due to the objectivity of their respective algorithm formulations and their reproducibility. A simple ED method was also included in the comparison for reference. The impact of using several techniques from the field of image processing along with the proposed DL-ANN method was also investigated, which will be further discussed in more detail in Section V.

Step 4. *Assessment of the SAE accuracy:* An ideal SAE method would produce in Step 3 an output spectrogram that would be identical to the one generated in Step 1, however in practice SAE methods are imperfect and therefore the estimated SAs will in general not be identical to the original ones. Therefore, this stage entails validating the accuracy of the considered SAE method by comparing the output of Step 3 to the original spectrogram generated in Step 1.

B. Experimental Platform

The obtained simulation results were validated against experimental results obtained with the hardware platform shown in Fig. 4, which was composed of a Signal Hound VSG25A vector signal generator (acting as the signal transmitter), a short coaxial cable along with a 20 dB attenuator (acting as the transmission channel), and a Tektronix RSA306B real-time spectrum analyser (acting as the signal receiver or spectrum monitoring system). To prevent transmission interference from nearby electronic devices, a wired connection was used. The transmitter and receiver were connected via USB to the same computer, where a Matlab control program was

[3]Notice that this choice is made simply for visual clarity in the graphical representation of the spectrograms and does not imply any assumption or any prior knowledge of the frequency extent of the signals to be measured. In this work, no prior information is assumed to be known about the time/frequency extent of the signals that may be present in a spectrogram.

[4]Note that the expressions provided in [50, eq. (7)] and [51, eq. (22)] assume real sampling. For complex sampling, the factors of 2 that appear in such equations should be replaced with 1.

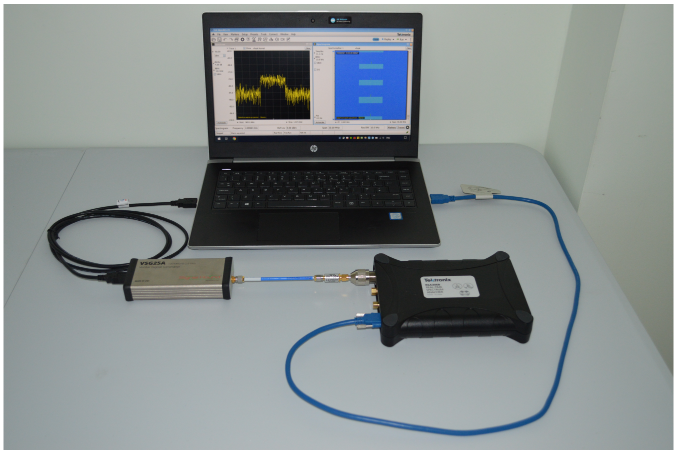


Fig. 4: Hardware prototype used in this work: vector signal generator (left), coaxial cable and attenuator (middle), and spectrum analyser (right).

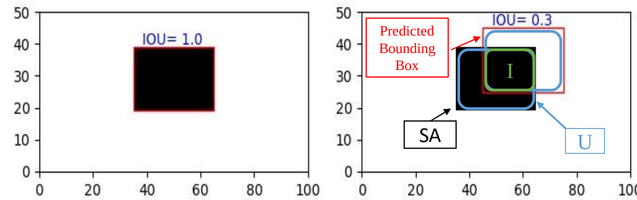


Fig. 5: Illustration of the IOU concept.

executed to coordinate the operation of both transmitter and receiver to ensure that the data were correctly synchronised for subsequent comparison later on. The control program was implemented using Matlab's Instrument Control Toolbox along with the libraries and Application Programming Interfaces (APIs) provided by the manufacturers of the vector signal generator and the spectrum analyser.

The settings of the experimental platform were configured to closely reflect the simulation environment. A multi-tone signal was generated at the transmitter with an OFDM-like spectral shape generated by 1001 unmodulated tones with random phase spaced at 10 kHz around a central frequency of 1 GHz, with a total signal bandwidth of 10 MHz. The centre frequency of the receiver was also configured to 1 GHz with a frequency span of 30 MHz (i.e., signal bandwidth was 1/3 of the frequency span). The relation between the transmission power configured at the signal generator and the SNR observed at the spectrum analyser was carefully calibrated to enable a fair comparison between simulation and experimental results.

C. Performance Metrics

Several performance metrics were used in this work both for DL-ANN training and optimisation as well as for evaluation of the final SAE accuracy, which are described below.

Mean Squared Error (MSE): The average squared difference between estimated and actual values is a popular metric commonly employed to assess the training of ANNs [52] and is also employed in this work. The MSE metric here employed is calculated based on the four output parameters of the proposed DL-ANN model shown in Fig. 2 as follows:

$$\text{MSE} = \frac{1}{4}\left((x-\tilde{x})^2 + (y-\tilde{y})^2 + (w-\tilde{w})^2 + (h-\tilde{h})^2\right)$$

where $m \in \{x, y, w, h\}$ denotes the true value of parameter m and $\tilde{m}$ represents its estimated value. The closer to zero the value of this metric, the more accurate the SAE.

Intersection over Union (IOU): This metric is commonly used to measure the accuracy of a model on a given dataset, especially in the context of object detection [53]. For each estimated SA, this metric is calculated based on the areas of the ground-truth and the estimated bounding boxes. In the particular case of SAE, the bounding box of each SA coincides with the SA edges and therefore the area of the bounding box equals the area of the SA itself. The IOU metric is defined as the quotient between I (the area of intersection of the actual and estimated bounding boxes) and U (the area of the union of the actual and estimated bounding boxes), as illustrated in Fig. 5. Mathematically, it can be calculated as IOU $= I/U$, where $I = I_w \cdot I_h$, with the width I_w and height I_h of the intersection area obtained as:

$$I_w = \min(x + w, \tilde{x} + \tilde{w}) - \max(x, \tilde{x}) \quad (1)$$

$$I_h = \min(y + h, \tilde{y} + \tilde{h}) - \max(y, \tilde{y}) \quad (2)$$

and the union area is obtained as $U = w \cdot h + \tilde{w} \cdot \tilde{h} - I$. The IOU metric takes values within the interval $[0, 1]$, with zero indicating the worst possible accuracy and one indicating a perfect SAE accuracy. The closer to one the value of this metric, the more accurate the SAE.

F1 score: The MSE and IOU metrics described above place the focus on the location and dimension of the estimated SAs with respect to the true SAs, but do not pay attention to what occurs outside those regions. It is worth noting that some SAE methods can *detect* SAs in regions where no true SA is present, in particular at low SNR, where a large number of missed detections and false alarms can be expected. To include the impact of these artefacts in the performance of SAE methods, the F1 score metric is also evaluated, which compares the output spectrogram to the ground-truth spectrogram on a point-by-point basis, thus taking into account the behaviour of the SAE method not only in the regions where SAs are present but also in the rest of the spectrogram. As opposed to other metrics that could be used for point-by-point comparison, the interest of the F1 score is that its calculation accounts for (and therefore is not biased by) the possible imbalance between the number of spectrogram points in the idle and busy states in the original test grid. The F1 score is defined as [54]:

$$F1 = \frac{2 \times TP}{2 \times TP + FP + FN}, \quad (3)$$

where TP, FP and FN represent the number of true positive, false positive and false negative detections, respectively [54].

The F1 score takes values within the interval $[0, 1]$, with zero indicating the worst possible accuracy and one indicating a perfect SAE accuracy. Therefore, the closer to one, the more accurate the SAE.

Computation time: The broad heterogeneity of SAE methods available in the literature and the inherent characteristics of the principles on which they rely makes it difficult to adopt a common definition of computational complexity that is applicable to all possible cases and enables a fair comparison among different SAE methods. This problem is resolved by assessing and comparing the computational cost or complexity of different methods by means of the computation time, measured as the time required to execute each SAE method in an actual processor. This is a common approach widely used in other SAE studies and is also employed in this work. Therefore, the computation time of each SAE method is also evaluated in this study. This measure is important because it affects the overall performance of the SAE method when it is practically implemented. For the proposed DL-ANN method, the computation time refers to the time required to execute the DL-ANN model once it has been trained. The training time, which is significantly longer than the execution time, is not included since in a practical system implementation the model would normally be trained only once and offline before it is deployed in a real implementation.

In this work, the MSE and IOU metrics are mainly used for model training and configuration (along with the F1 score as well), while the F1 score and the computation times are mainly used for assessment of the final SAE accuracy. As it will be shown, the F1 score provides a suitable metric that follows a similar trend as the MSE and IOU metrics while providing a more complete characterisation of the output spectrogram generated by SAE methods.

D. DL-ANN Model Training and Configuration

Raw Dataset Construction: Raw datasets were generated based on software simulations and hardware experiments as explained in Sections IV-A and IV-B, respectively. For each SNR value, a total of 60,000 independent test grids (i.e., original spectrograms) were generated and the corresponding features and labels were extracted in order to construct the required dataset, which was divided into separate subsets used for training (60%), validation (20%) and testing (20%). As explained in Section III-B, two different scenarios are considered regarding the input information provided to the DL-ANN model. In the first scenario only the binary spectrogram obtained from spectrum measurements is provided, while in the second scenario the SNR value at which the provided spectrogram was generated is also fed to the network as a second input parameter. The raw data generation process for both scenarios is illustrated in Figs. 6 and 7, respectively. The only difference is the fact that the SNR value at which each test grid (original spectrogram) is generated is not considered as an input feature in scenario 1 (Fig. 6) while it is considered as an input feature in scenario 2 (Fig. 7).

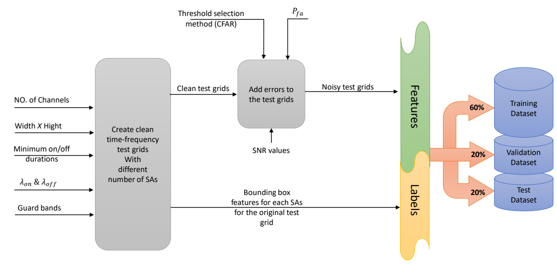


Fig. 6: Construction of DL-ANN raw dataset for scenario 1.

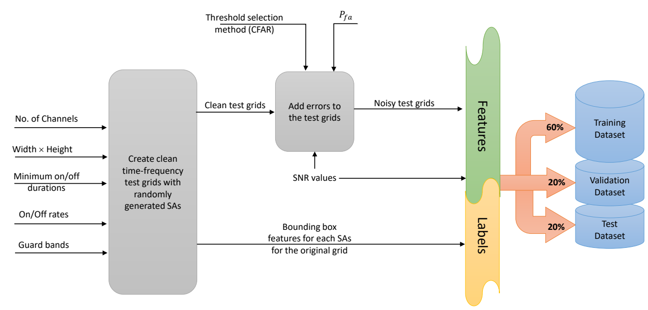


Fig. 7: Construction of DL-ANN raw dataset for scenario 2.

Before using the datasets for training, validation or testing, they must be preprocessed in order to extract the relevant features and corresponding labels. Python is utilised to this end because of the availability of several tools and advanced DL libraries (e.g., TensorFlow [55], Keras [56] and PyTorch [57]) that help not just with the dataset preparation, but also with the construction, training and testing of the DL model. After this, the prepared dataset is ready for training of the DL-ANN model and its validation (which is important to ensure that the ANN can generalise to new data and avoid the over-fitting problem) before the final testing is carried out.

Hyperparameter Tuning and DL-ANN Optimisation: After the data preparation process, the DL-ANN model is trained to help it learn the optimum hyperparameter values. Table I shows a summary of the main hyperparameters considered and their selected values. Among the numerous hyperparameters that are amenable to optimisation, several hyperparameter settings were tested for the number of hidden layers, number of neurons per layer and batch size (with the selected values shown in bold font), while other relevant parameters were set to standard and commonly used values [58]. The approach used in this study was to optimise the first three hyperparameters shown in Table I based on the MSE as the performance metric as a function of the number of epochs. The hyperparameter optimisation was carried out based on a manual grid search method and the Adaptive Experimentation Platform (Ax). This approach was chosen because it allows an intelligent selection of properties in the search space when

TABLE I: Hyperparameter tuning settings

Hyperparameter	**Settings**
Number of hidden layers	[1, 2, 3, **4**]
Number of neurons	[16, 32, 64, 128, **256**]
Batch size	[5, 10, 15, 20, **25**]
Epoch	100
Optimiser	Adam
Learning rate	0.001
Dropout regularisation	0.4
Activation function	ReLU, Sigmoid (output)

there is no opportunity for explicit choice of properties [59]. The results of the optimisation process are shown in Fig. 8 and the corresponding optimised DL-ANN models for scenarios 1 and 2 are shown in Figs. 9 and 10, respectively. Notice that the number of hidden layers and their dimensions correspond to the selected values shown in boldface in Table I. The input information provided to the DL-ANN model is a binary spectrogram that, as discussed in Section IV-A, has a default size of 50×100 points, which corresponds to an input size of 5000 elements as shown in the input layer of Figs. 9 and 10. On the other hand, the output of the last layer (i.e., the final output of the DL-ANN model) has a total size of 80 elements, which corresponds to a matrix of 4×20 elements. This provides enough capacity to store information of the bounding boxes for up to 20 SAs, given that each bounding box is defined by a four-element tuple as discussed in Section III-B (see also Fig. 2). This was observed to be sufficient in all our experiments. The DL-ANN will detect automatically the number of SAs present in the spectrogram and fill the values of one tuple in the 4×20 matrix for each detected SA. The rest of positions of the output matrix will be filled with zeros. Counting the number of non-zero tuples in the output matrix provides the information about the number of detected SAs, while the values contained in the used positions of the output matrix provide the information of the corresponding bounding boxes for the detected SAs. Notice that if the output matrix is configured with a capacity that is not large enough, then the information of some SAs will be lost, however this can be easily found out by simply checking whether all the positions have been filled and increasing the size if needed.

Training, Validation and Testing Options: For the DL-ANN to provide accurate results, it first needs to be trained with labelled data. The labels for training are provided as a matrix similar to the one provided by the DL-ANN as an output, with one tuple (x, y, w, h) for each generated SA and zeros in the remaining empty positions. The labels were added to the training data set in the same order in which the SAs were generated and added to the spectrogram produced in Step 1 of Section IV-A, which in this work is in increasing order of frequency and time. As a result, the network tends to provide an output matrix where the detected SAs tend to appear in similar order, however this is not necessarily the only correct output (any permutation of the order of detected SAs is still valid). In any case, notice that the order in which the detected SAs are provided by the DL-ANN in the output matrix is irrelevant when it comes to the SAE accuracy – it is the accuracy of the parameters (x, y, w, h) estimated for each detected SA what counts.

For the DL-ANN model training and validation, four options were considered, the first three related to scenario 1 (one input

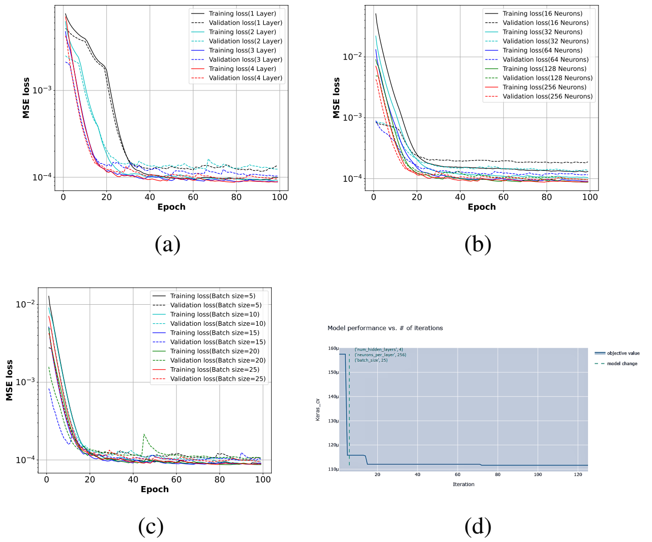


Fig. 8: DL-ANN hyperparameter optimisation based on MSE: (a) using different numbers of hidden layers (with 256 neurons per layer and 25 batch size), (b) using different numbers of neurons per layer (with 4 hidden layers and 25 batch size), (c) using different batch sizes (with 4 hidden layers and 256 neurons per layer), and (d) using the Adaptive Experimentation (Ax) platform.

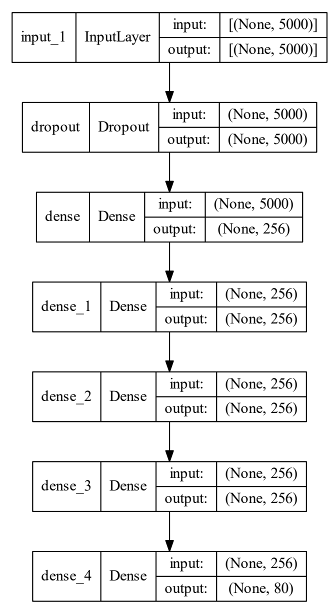


Fig. 9: Optimised DL-ANN model for scenario 1.

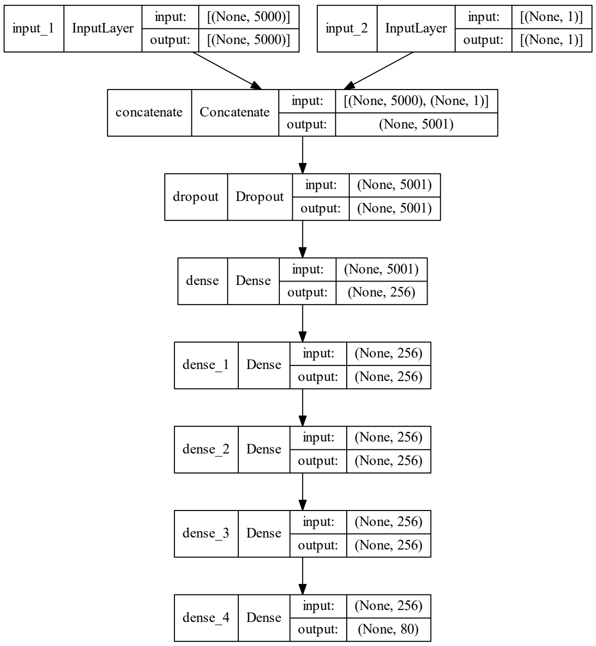


Fig. 10: Optimised DL-ANN model for scenario 2.

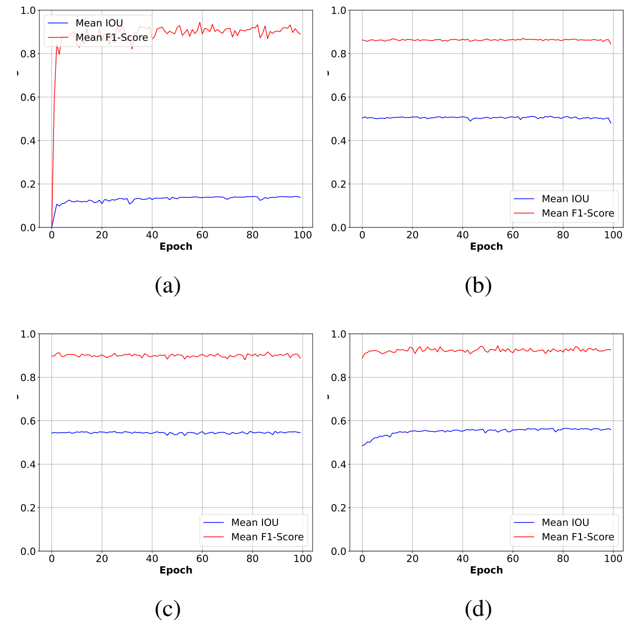


Fig. 11: DL-ANN training performance for different training options: (a) option 1, (b) option 2, (c) option 3, and (d) option 4.

layer for the input spectrogram) and the last one related to scenario 2 (with two input layers including the estimated SNR at the receiver):

- Option 1: Single input layer for input spectrograms generated at –5 dB SNR.
- Option 2: Single input layer for input spectrograms generated at –10 dB, –7 dB and –5 dB SNR.
- Option 3: Single input layer for input spectrograms generated at SNR values from –20 dB to +5 dB in 1-dB increments.
- Option 4: Same as Option 3 with a second input layer that provides the SNR at which the input spectrogram provided in the first input layer was obtained.

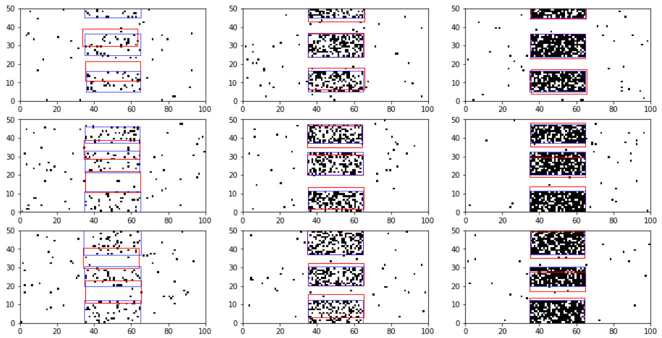
In training option 1, the DL-ANN model was trained based on input spectrograms generated at –5 dB SNR (which represents a case of low SNR), while in option 2 the model was trained using input spectrograms generated at three different SNR values, namely –10 dB, –7 dB and –5 dB (which represents a larger training data set but still a limited one compared to the full SNR operational range from –20 dB to +5 dB). Option 3 trains the model with data generated at all possible SNR values from –20 dB to +5 dB in 1-dB increments. These three options were considered to determine the degree to which the amount of data available for training can help the DL-ANN model produce accurate SAE outputs and is motivated by the fact that in some practical application scenarios it may not be possible to acquire data for training at all the SNR levels that may be experienced once the model is deployed in regular working conditions. Finally, option 4 is considered to determine the degree to which providing additional input information to the model (namely, the SNR value at which the input spectrogram was obtained) can help produce more accurate outputs. The results of the DL-ANN training for these four options can be seen in Fig. 11 in terms of the IOU and F1 score metrics. As it can be observed, the trends are similar for the IOU and F1 score. For simplicity, the remainder of the performance analysis will focus on the F1 score only, which is in line with previous work in [19], [41], [43], [44].

Once trained and validated, the DL-ANN model was tested using the same kind of data. However, in the testing stage only the input features (spectrogram only in options 1–3 and also the receiving SNR in option 4) were fed to the DL-ANN to predict its output, while labels were used to quantify the accuracy of the SAE result. It is worth noting that in all cases the input spectrograms employed in the testing stage were generated for all SNR values from –20 dB to +5 dB regardless of what data were used in the training/validation stage (including the testing of options 1 and 2). The results of this testing are presented and discussed in Section V.

V. Performance Evaluation

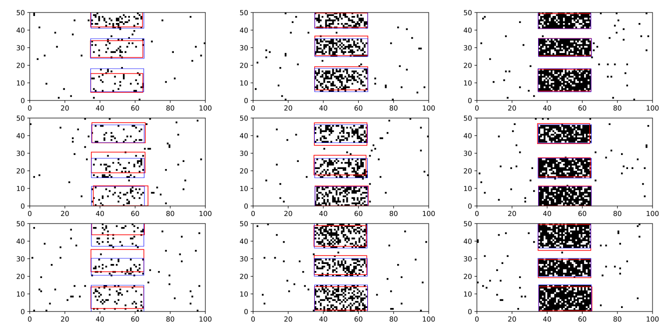
A. *Performance of DL-ANN*

In order to illustrate the output produced by the proposed DL-ANN model under the four different training options described in Section IV-D, Figs. 12–15 show some examples of the estimated SAs for various spectrograms under three different SNR values, namely –5 dB, –7 dB and –10 dB (notice that all these cases correspond to a low SNR regime, where SAE becomes more challenging). The figures show the spectrograms observed at the receiver along with the estimated SAs, which are depicted as red rectangles. Moreover, the



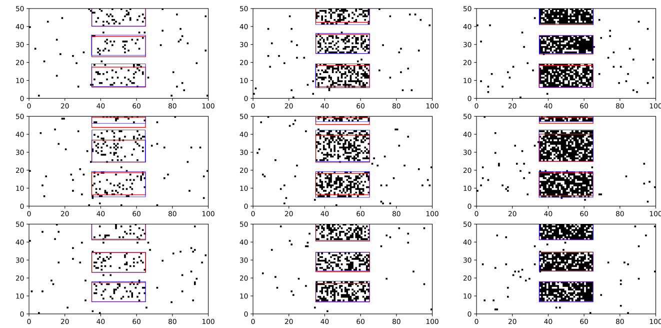
(a) (b) (c)

Fig. 12: Example of SAE for Option 1: (a) SNR = –10dB, (b) SNR = –7dB, and (c) SNR = –5dB.



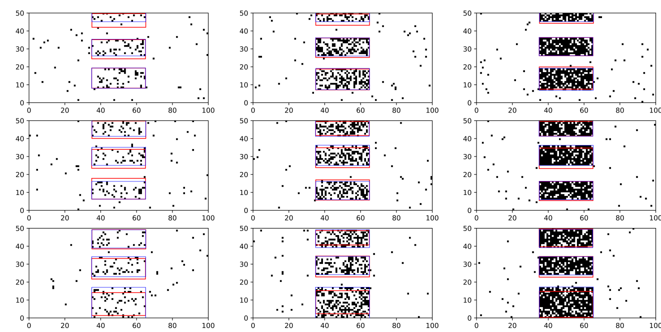
(a) (b) (c)

Fig. 13: Example of SAE for Option 2: (a) SNR = –10dB, (b) SNR = –7dB, and (c) SNR = –5dB.



(a) (b) (c)

Fig. 14: Example of SAE for Option 3: (a) SNR = –10dB, (b) SNR = –7dB, and (c) SNR = –5dB.

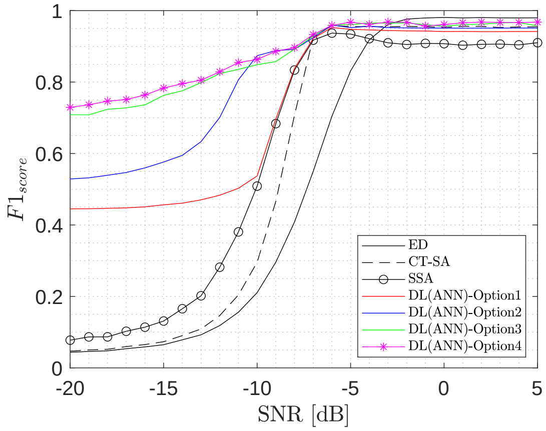


(a) (b) (c)

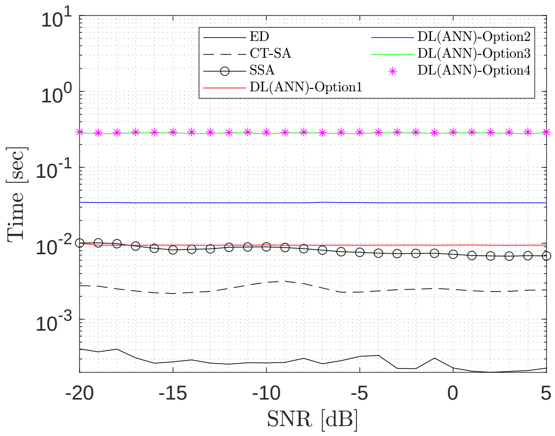
Fig. 15: Example of SAE for Option 4: (a) SNR = –10dB, (b) SNR = –7dB, and (c) SNR = –5dB.

original (ground truth) SAs are also shown in the figures as blue rectangles in order to clearly visualise the IOU.

The results shown in Fig. 12 for option 1, where the DL-ANN model is trained using only spectrograms observed at –5 dB SNR, suggest that the SAs in the received spectrogram can be detected with a reasonable degree of accuracy at the SNR level at which the model is trained (i.e., – 5 dB) and also at slightly lower SNRs (i.e., –7 dB), but fails to produce satisfactory results at a much lower SNRs (i.e., –10 dB). This suggests that the network can perform well when operating at the same SNR at which it was trained and also at slightly different SNR values, but fails to deliver satisfactory results when it operates at SNR values that are substantially different from those at which it was trained. To confirm this, the DL-ANN model was also trained at the three tested SNR values in option 2 and, as it can be appreciated in Fig. 13, in this case the model also provides a reasonable detection performance at lower SNR values (–10 dB) once it has been trained for those particular operating conditions. It is interesting to note that, even though the presence of SAs at –10 dB SNR is hardly recognisable for the human eye, the DL-ANN model can identify correctly the number of SAs and their locations with a remarkable level of accuracy once it has been trained with data observed at such low SNR level (see Fig. 13a). Moreover, it is also worth noting that training the network with additional data at lower SNR values such as –10 dB also makes the network perform better at higher SNR values such as –7 and –5 dB, as it can be appreciated by comparing Figs. 13b and 13c with Figs. 12b and 12c, respectively, which suggests that the network will experience an improved learning process from any training data even when operating at other SNR levels. The results in Fig. 14 correspond to option 3, where the model is trained with data generated at all the possible SNR values considered in this study (from –20 dB to +5 dB in 1-dB increments); in this particular case, there is no significant difference with respect to Fig. 13, since in both cases the network has been trained with data generated at the three SNR values shown in these examples. In Fig. 15, the network is also trained with spectrogram data generated at all the possible SNR values and, in addition to that, is also trained with a second input feature which is the actual SNR value at which the spectrogram is generated. This additional input information would in principle be expected to produce a more accurate detection of the SAs present in the spectrogram. However, as it can be observed by comparing Figs. 14 and 15, there does not seem to be a significant difference, which suggests that providing the SNR as a second input feature may not have a relevant impact on the model performance.



(a)



(b)

Fig. 16: Performance comparison of options 1–4 in terms of: (a) F1 score, and (b) computation time.

To verify the above statements and assess the performance of the DL-ANN model in a more quantitative manner, Fig. 16 shows the performance of the four considered training options in terms of the F1 score as a function of the SNR along with the corresponding computation time. The figure also includes the performance of the two selected benchmark methods (namely CT-SA and SSA) and the performance of a simple energy detector (ED), which cannot be considered as a SAE method in strict sense but provides an interesting baseline for comparison purposes (ED is the signal detection step needed to generate the binary spectrogram previous to the SAE process). Fig. 16a corroborates the observations made above based on Figs. 12–15. It can be noticed that the performance of the DL-ANN model when trained as in option 1 starts to degrade significantly for SNR values below –5 dB, which is the SNR at which the employed training data were generated. Similarly, when trained as in option 2, the performance starts to degrade significantly when the operating SNR falls below the lowest SNR used for training (i.e., –10 dB). When the DL-ANN model is trained as in option 3 with spectrogram data generated at all the SNR values at which the model operates, the performance is significantly improved over the whole SNR range and, even though the performance experiences a natural degradation as the SNR decreases, the degradation in this case is significantly less accentuated than in the other two training options. It is also worth noting that the training considered in option 4, where also the actual SNR value is provided as an input feature, provides only a marginal performance improvement with respect to the training based on option 3, where only spectrogram data (without actual SNR information) are used. Therefore, this suggests that the training in option 3 can be considered as a preferred option since it does not require the estimation of the receiving SNR, which would imply additional complexity in a practical system implementation. When the DL-ANN model is trained based on spectrograms generated at all the possible SNR values as in option 3, the proposed system provides an excellent performance over the whole SNR range, comparable to that attained with option 4, but with a much simpler system implementation (where the receiving SNR does not need to be estimated). Moreover, it is worth noting that the DL-ANN model trained as in option 3 outperforms both SAE benchmark methods (i.e., CT-SA and SSA) over the whole SNR range. While the SAE accuracy is similar at relatively high SNR values (at around –7 dB and above), there is a significant performance improvement attained with the DL-ANN method at lower SNR values. In particular, at SNR values as low as –20 dB, the DL-ANN model can provide an estimation accuracy (based on the F1 score) of about 73% (for option 3) while the benchmark methods would provide an estimation accuracy of 5–7% for the same SNR level. This significant performance improvement results in a noticeable extension of the SAE detection sensitivity (i.e., the ability to accurately detect SAs at lower SNR values). The price to be paid for this significant performance improvement, as observed in Fig. 16b, is a higher computational cost in terms of the computation time required to run the DL-ANN SAE method. Fig. 16b indeed shows the existence of a trade-off between SAE accuracy and required computation time, with the two benchmark methods (CT-SA and SSA) requiring the lowest computation time but also providing the lowest SAE accuracy, and the proposed DL-ANN method (training options 3 and 4) providing the best SAE accuracy at the expense of a higher computation time. However, with the availability of powerful processors nowadays (and presumably more powerful ones in the future), this is an affordable cost that is worth paying for the significant performance improvements that can be attained with the DL-ANN model, in particular in the low SNR regime.

The results shown in Fig. 16 correspond to an AWGN channel. The performance under Rayleigh fading is illustrated in Fig. 17. By comparing Figs. 16a and 17 it can be noticed that the impact of fading is in general a slight degradation of the SAE accuracy for all the considered methods, which is a consequence of the lower probability of correct signal detection at each spectrogram point due to fading. However, the degrading effect is more severe for the ED curve than

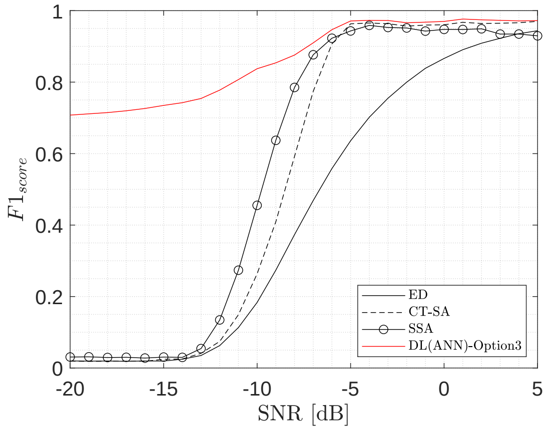


Fig. 17: Performance of the proposed and reference SAE methods in terms of F1 score vs. SNR under Rayleigh fading.

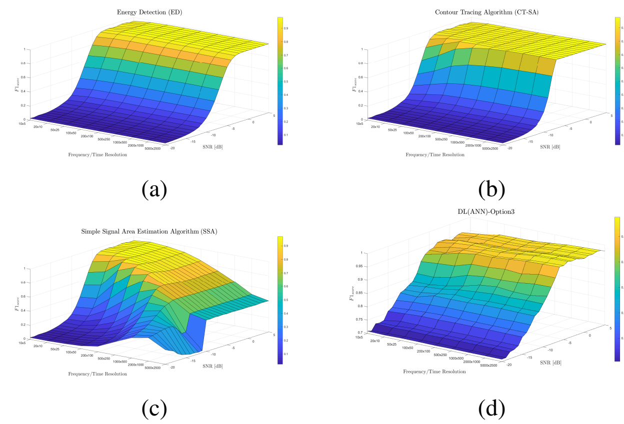


Fig. 18: Impact of the spectrogram resolution on the SAE accuracy for: (a) energy detection (ED), (b) CT-SA, (c) SSA, and (d) DL-ANN.

for the SAE methods. This can be explained by the fact that SAE methods, as opposed to ED, combine the states detected in various spectrogram points to estimate the present SAs, which can help overcome the degrading effects of a higher probability of misdetection at individual spectrogram points caused by fading. More importantly, it can also be observed that the proposed DL-ANN method outperforms the reference SAE methods not only in an ideal AWGN channel but also in a more realistic wireless communication channel that includes fading in addition to noise.

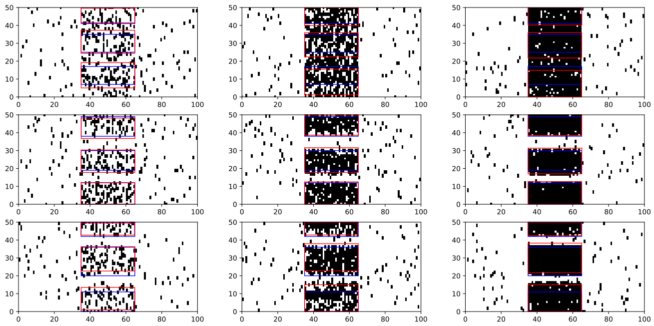
The results presented so far have been obtained for a spectrogram resolution of 50×100 points, which is the default resolution considered in this work unless otherwise stated (see Section IV). The impact of the spectrogram resolution on the performance of SAE methods was investigated in [19], where it was observed that modifying the spectrogram resolution does not have a significant impact on the SAE accuracy in general, except for the SSA method where it was observed that employing excessively high spectrogram resolutions may result in a degradation of the SAE accuracy (this could be because the SSA method may require some parameter optimisation procedure as a function of the employed spectrogram resolution, which has not been investigated and is beyond the scope of this research). This is illustrated in Fig. 18, where the impact of the spectrogram resolution on the SAE accuracy[5] is shown as a function of the SNR. As it can be appreciated, the performance of the proposed DL-ANN method is not significantly affected by the employed spectrogram resolution and, more importantly, the relative performance improvement of the proposed DL-ANN method with respect to the benchmark methods is maintained regardless of the considered resolution.

[5]Recall from Section IV-A (Step 4), that the SAE accuracy is evaluated by comparing the spectrograms obtained in Steps 1 and 3, both of which have a discrete domain. In a real system implementation, obtaining the spectrogram of Step 1 would require sampling and digitising the analogue radio-frequency signals and the selected sampling resolutions in the time and frequency domains would introduce some error component in the spectrogram of Step 1. Such error component is independent of the SAE method subsequently employed and therefore is not included in these results in order to provide a fair comparison. The results presented in this work quantify the difference between the spectrograms described in Steps 1 and 3 of Section IV-A; such error component is due solely to the employed SAE method.

B. Performance with Image Processing Techniques

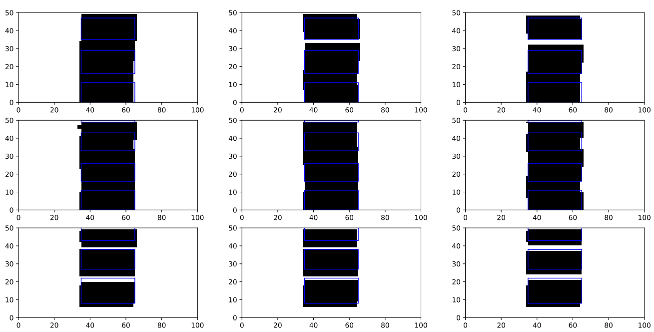
As discussed in Section III-A, the problem of SAE can be addressed from the point of view of image processing by looking at binary spectrograms as binary images where each time-frequency point represents an image pixel, which allows the application of image processing techniques. This section explores the performance of the proposed DL-ANN method when combined with some popular image processing techniques. In particular, two image processing techniques are considered here, namely morphological operations and a combination of edge detection and flood fill. These image processing techniques are combined with the proposed DL-ANN method by using them as pre/post-processing stages, where they can be applied only before (pre-processing), only after (post-processing) or both before and after (pre- and post-processing) the DL-ANN model is run.

Morphological Operations (MO) are carried out by moving a small (typically squared or rectangular) filter template referred to as *Structuring Element* (SE) over the binary image by centering it at every image pixel (based on the defined SE's origin, which is usually its geometric centre) and performing some logical operation between the SE pixels and the image pixels that fall within the SE template. The basic MOs include erosion, dilation, opening and closing [60], [61]. In morphological erosion (dilation) the image pixel at the centre of the SE is set to one if all (any) of the neighbouring pixels within the SE have the value one, and zero otherwise. Morphological erosion removes islands and small objects in the input image, so that only substantive objects remain and can be useful to remove false alarms in the input spectrogram, while morphological dilation has the opposite effect to erosion, it adds more pixels to the boundaries of existing regions, making objects more visible and reducing gaps between them, which can be useful to fill in missed detections within SAs. The other two morphological operations are obtained as a combination,



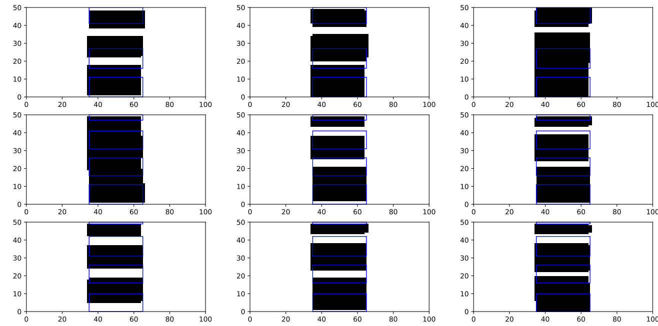
(a) (b) (c)

Fig. 19: Performance of dilation before DL-ANN: (a) SNR = –10dB, (b) SNR = –7dB, and (c) SNR = –5dB.



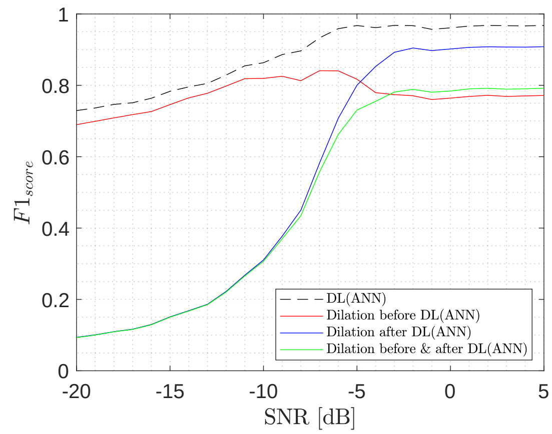
(a) (b) (c)

Fig. 20: Performance of dilation after DL-ANN: (a) SNR = –10dB, (b) SNR = –7dB, and (c) SNR = –5dB.

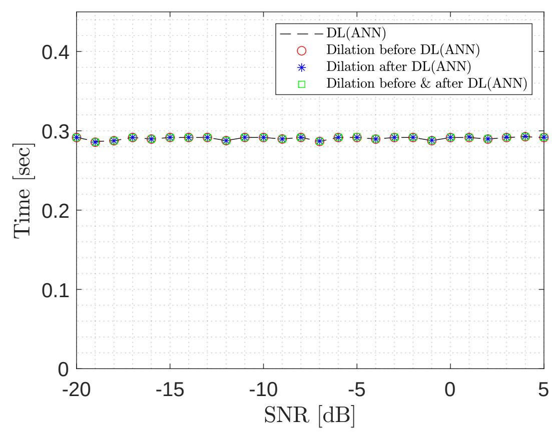


(a) (b) (c)

Fig. 21: Performance of dilation both before and after DL-ANN: (a) SNR = –10dB, (b) SNR = –7dB, and (c) SNR = –5dB.



(a)

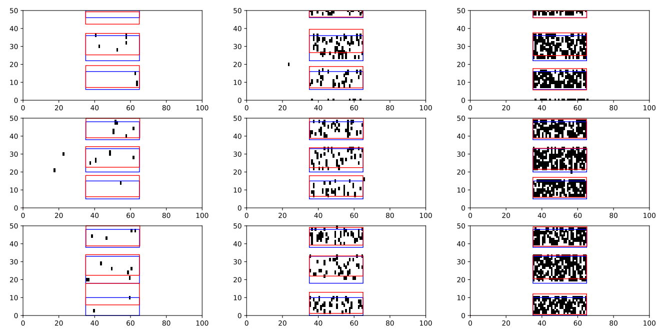


(b)

Fig. 22: Performance of DL-ANN combined with morphological dilation in terms of: (a) F1 score, and (b) computation time.

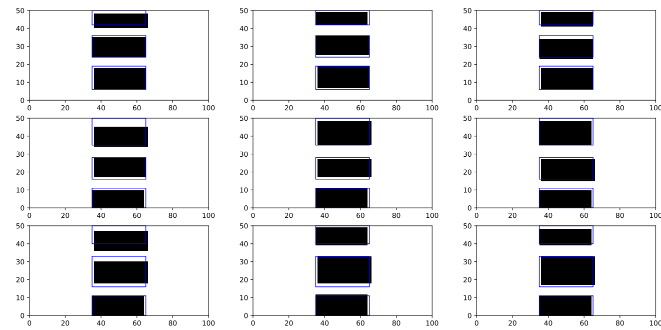
where morphological opening (closing) is obtained by first eroding (dilating) and then dilating (eroding) an image using the same SE for both operations. Morphological opening can remove small entities from an image while conserving the dimensions and proportions of larger objects almost unaltered, while morphological closing enlarges an image and then corrodes the expanded image, with the visual effect being the repletion of gaps in the image. The performance of the DL-ANN model is here shown when combined with dilation and opening, which are the two operations that can help fill gaps within SAs due to signal missed detections and therefore provide a more clear visualisation of the SAs in a spectrogram. Without loss of generality, the other two MOs (erosion and closing) are not shown here to avoid an excessively long analysis but similar conclusions can be reached.

The performance of the DL-ANN model when combined with the morphological dilation operation as a pre-, post, and both pre/post-processing technique is illustrated in Figs. 19, 20 and 21, respectively. Notice that the application of a MO after the DL-ANN model means that the information about the location and dimensions of the SAs detected by the DL-ANN (represented by red rectangles) is lost, hence such red rectangles are not shown in Figs. 20 and 21 (however these cases are also included in the study for completeness). As it can be appreciated, the overall effect of the dilation operation is to expand the regions where the presence of signal components is detected. This effectively fills gaps within genuine SAs created



(a) (b) (c)

Fig. 23: Performance of opening before DL-ANN: (a) SNR = –10dB, (b) SNR = –7dB, and (c) SNR = –5dB.



(a) (b) (c)

Fig. 24: Performance of opening after DL-ANN: (a) SNR = –10dB, (b) SNR = –7dB, and (c) SNR = –5dB.

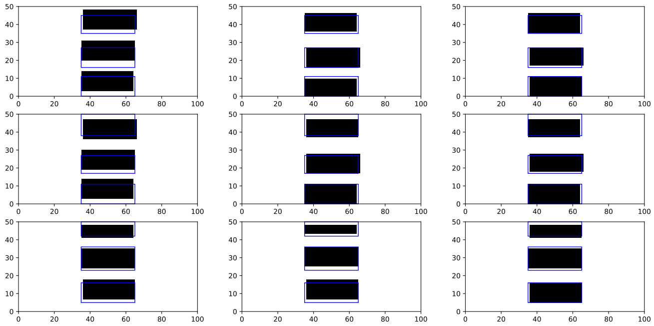
by signal missed detections, however it also magnifies these regions to the extent that the DL-ANN model overestimates the true dimensions of the SAs. The overall result, as shown in Fig. 22a, is an overall degradation of the detection accuracy. Notice that the introduction of the dilation step does not affect significantly the overall computation cost as indicated by Fig. 22b, which is due to a negligible computation time with respect to that required by the DL-ANN model. However, despite this negligible computational load, the introduction of this particular MO does not provide any performance improvement and therefore is of little utility. On the other hand, the performance of the DL-ANN model when combined with the morphological opening operation as a pre-, post, and both pre/post-processing technique is illustrated in Figs. 23, 24 and 25, respectively. The performance can be appreciated in more detail in Fig. 26. In this other case, it can be noted that the use of morphological opening as a pre-processing technique provides a very similar detection performance as the DL-ANN model alone without a noticeable variation in the required computational cost. The introduction of dilation as a pre-processing step can indeed result in a slightly better SAE accuracy at high SNR values (above around –7 dB) and in that SNR regime can be of some utility, even though the DL-ANN model on its own can achieve a very similar performance.

The second image processing technique explored here is a combination of edge detection (used to estimate the edges of the potential SAs in a spectrogram) and flood fill (used to fill the area inside the detected edges) [44]. In this approach, the Canny edge detector is used followed by a standard flood fill method [62], [63]. The motivation for using these image processing techniques is to attempt to enhance the recognisability of SAs (or fragments thereof) to help the DL-ANN recognise the SAs in the spectrogram. However, in this case, as observed in Figs. 27–30, the introduction of this image processing approach cannot improve the SAE accuracy already attained by the DL-ANN model itself, while resulting in slightly increased computation times. The performance of this combination of image processing techniques as a stand-alone SAE method (i.e., without DL-ANN) was investigated in [44], where it was shown that it can provide better accuracy than other previously proposed SAE methods. However, the comparison of Fig. 17 of [44] with the performance shown in this section for the proposed DL-ANN indicates that the DL-ANN method also performs better than edge detection plus flood fill when considered as stand-alone SAE methods, in particular in the low SNR regime.

In summary, the results obtained in this subsection suggest that image processing techniques, which may be suitable for their application to the SAE problem, do not improve the performance attained by the proposed DL-ANN method. The use of image processing techniques in the context of SAE has provided noticeable benefits and performance improvements in some previous studies [41], [43], [44]. However, the use of more advanced techniques such as DL can achieve by itself excellent levels of SAE estimation accuracy over the whole range of SNR values without the need of combining them with image processing techniques, which in this subsection have been shown to be of limited utility or, at best, provide very slight performance improvements in some cases.

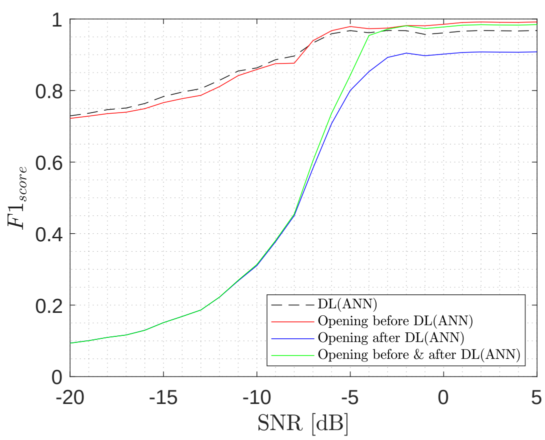
C. Experimental Validation

The results presented so far have been obtained based on software simulations. Simulations are a convenient and efficient way to explore the performance of the SAE approaches considered in this work under a broad range of operating conditions, however an experimental validation is required to provide a more convincing case showing the potential benefits that the proposed DL-ANN method can bring in a practical system implementation. Fig. 31 presents a comparison of the simulation and experimental performance results obtained for the DL-ANN model under the four considered training options, while Fig. 32 also includes the variants based on the combination with image processing techniques. As it can be appreciated, the obtained simulation results match very closely with their experimental counterparts, thus confirming the conclusions derived from the analysis presented above and corroborating the performance improvements that the proposed DL-ANN method for SAE can achieve in a practical system

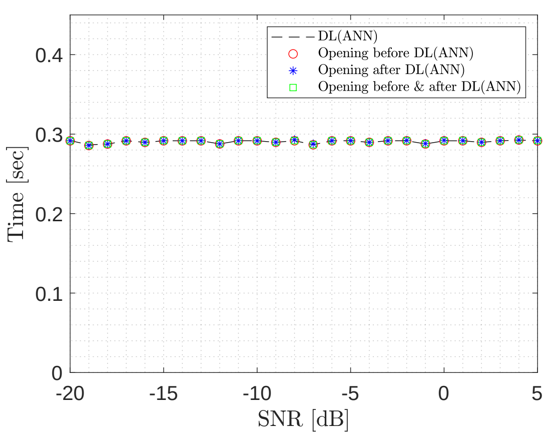


(a) (b) (c)

Fig. 25: Performance of opening both before and after DL-ANN: (a) SNR = –10dB, (b) SNR = –7dB, and (c) SNR = –5dB.



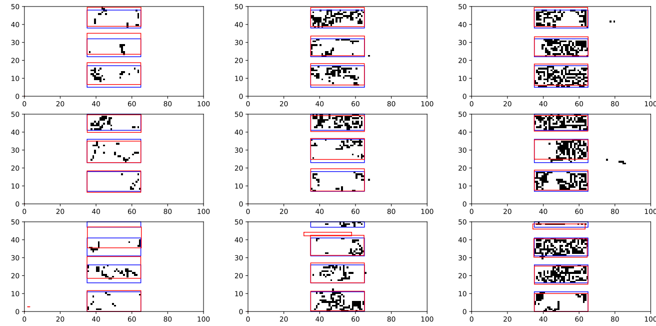
(a)



(b)

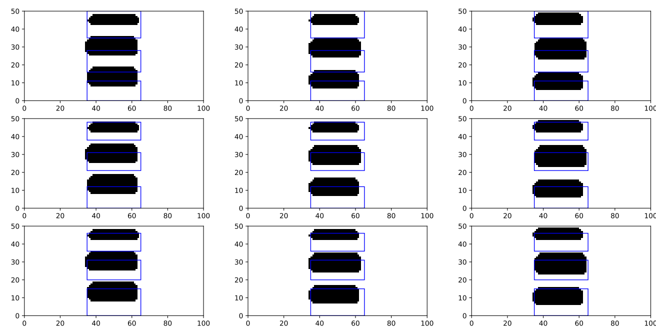
Fig. 26: Performance of DL-ANN combined with morphological opening in terms of: (a) F1 score, and (b) computation time.

implementation. As it can be noticed, the proposed DL-ANN approach can provide significant SAE accuracy improvements compared to other SAE methods from the literature, in particular when considering the low SNR regime.



(a) (b) (c)

Fig. 27: Performance of edge detection and flood fill before DL-ANN: (a) SNR = –10dB, (b) SNR = –7dB, and (c) SNR = –5dB.

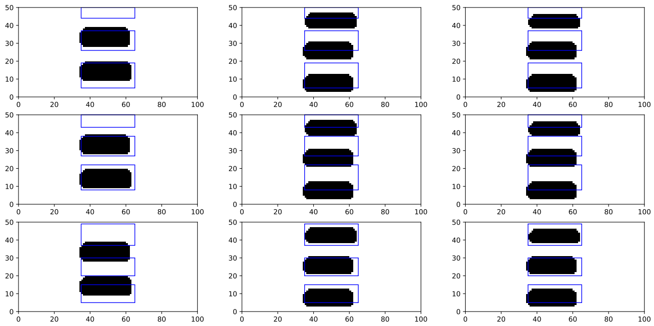


(a) (b) (c)

Fig. 28: Performance of edge detection and flood fill after DL-ANN: (a) SNR = –10dB, (b) SNR = –7dB, and (c) SNR = –5dB.

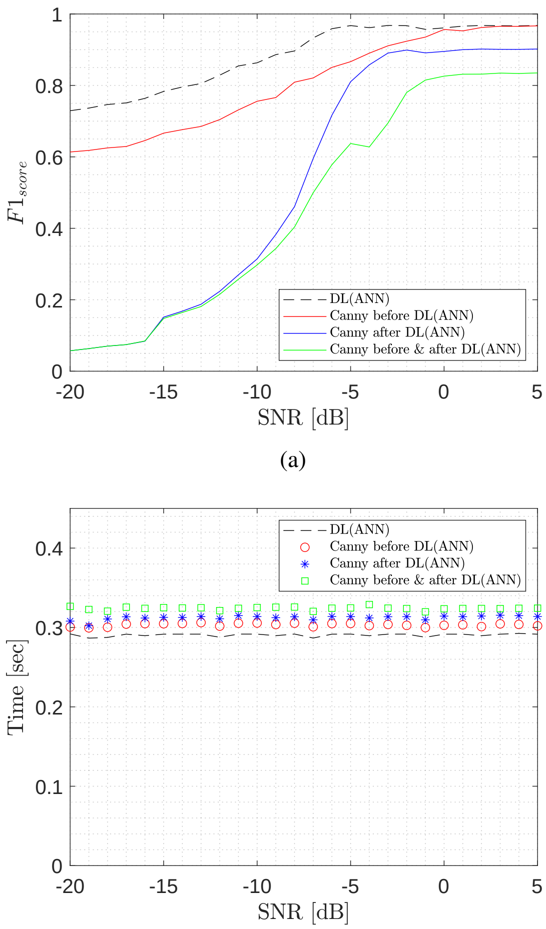
VI. Conclusion

In this work, a novel technique for Signal Area Estimation (SAE) has been proposed using Deep Learning based on Artificial Neural Network (DL-ANN) for enhanced extraction of Signal Area (SA) information from radio spectrograms. The proposed DL-ANN method has shown overall an excellent performance over the whole range of SNR levels, with significant improvements in particular in the low SNR regime (e.g., 73% estimation accuracy at –20 dB SNR compared to 5–7% for reference methods from the literature). The performance of the proposed DL-ANN model when combined with image processing techniques, which has been proven to be a suitable approach in previous studies, has been explored as well. The obtained results have shown that the use of DL techniques in the context of SAE, such as the proposed DL-ANN model, can achieve excellent levels of SAE accuracy without the need of assistance from image processing techniques, which in this particular case are of rather limited utility or, at best, provide very slight performance improvements in some cases. The obtained simulation results have been compared with experimental results obtained with a hardware platform specifically designed and implemented to this end,



(a) (b) (c)

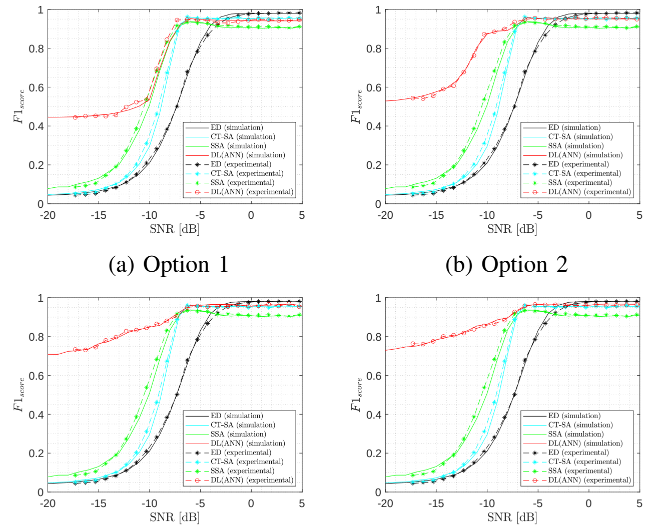
Fig. 29: Performance of edge detection & flood fill before and after DL-ANN: (a) SNR = –10dB, (b) SNR = –7dB, and (c) SNR = –5dB.



(a)

(b)

Fig. 30: Performance of DL-ANN combined with edge detection and flood fill in terms of: (a) F1 score, and (b) computation time.



(a) Option 1 (b) Option 2

(c) Option 3 (d) Option 4

Fig. 31: Experimental validation of the performance of the proposed DL-ANN method and the reference benchmark methods (ED, CT-SA, SSA) in terms of the F1 score as a function of the SNR for the four different training options considered in this work.

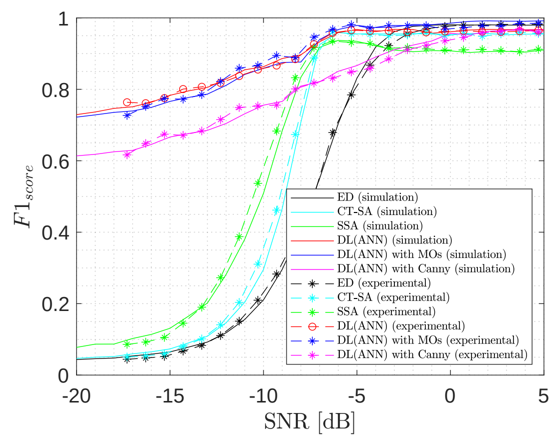


Fig. 32: Experimental validation of the performance of the proposed DL-ANN method (including the variants based on image processing techniques) and the reference benchmark methods (ED, CT-SA, SSA) in terms of the F1 score as a function of the SNR (for option 3).

thus corroborating the performance improvements that the proposed DL-ANN method for SAE can achieve in a practical system implementation. It is also worth noting that, in addition to the significant SAE accuracy improvements compared to other SAE methods from the literature (in particular in the low SNR regime), a key feature of the proposed method is the capability to extract the location and dimensions of the detected SAs automatically. Overall, the proposed technique is a promising solution for SAE and the automatic processing of radio spectrograms in spectrum-aware wireless systems.

The use of Deep Learning (DL) techniques for the detection of signals in various contexts has been receiving an increasing level of attention recently. Some DL-related techniques such as object detection methods have been successfully applied to signal detection related problems (e.g., see [46]) and the investigation of more advanced DL methods for the problem of SAE estimation (e.g., object detection-based SAE) is a promising research direction that is suggested as future work.

REFERENCES

[1] Y. Jie, Y. Chenzhou, and Z. Yue, "Modulation classification based on spectrogram," *Journal of Systems Engineering and Electronics*, vol. 16, no. 3, pp. 475–488, Sep. 2005.

[2] T. J. Lynn and A. Z. Sha'ameri, "Automatic analysis and classification of digital modulation signals using spectogram time-frequency analysis," in *Proc. Int'l. Symp. Commun. and Information Technol. (ISCIT 2007)*, Oct. 2007, pp. 916–920.

[3] A. Shklyaeva, P. Kovar, and D. Kubanek, "Classification of digital modulations mainly used in mobile radio networks by means of spectrogram analysis," in *Proc. 12th IFIP Int'l. Conf. Personal Wireless Commun. (PWC 2007)*, Sep. 2007, pp. 341–348.

[4] S. Jeong, U. Lee, and S. C. Kim, "Spectrogram-based automatic modulation recognition using convolutional neural network," in *Proc. 10th Int'l. Conf. Ubiquitous Future Netw. (ICUFN 2018)*, Jul. 2018, pp. 1–3.

[5] Y. Zeng, M. Zhang, F. Han, Y. Gong, and J. Zhang, "Spectrum analysis and convolutional neural network for automatic modulation recognition," *IEEE Wireless Comms. Letters*, vol. 8, no. 3, pp. 929–932, Jun. 2019.

[6] W. Jiang, X. Wu, Y. Wang, B. Chen, W. Feng, and Y. Jin, "Time-frequency-analysis-based blind modulation classification for multiple-antenna systems," *Sensors*, vol. 21, no. 231, pp. 1–19, Jan. 2021.

[7] F. A. Bhatti, M. J. Khan, A. Selim, and F. Paisana, "Shared spectrum monitoring using deep learning," *IEEE Tran. Cognitive Commun. and Networking*, vol. 7, no. 4, pp. 1171–1185, Dec. 2021.

[8] J. M. Tarongi and A. Camps, "Radio frequency interference detection and mitigation algorithms based on spectrogram analysis," *Algorithms*, vol. 4, no. 4, pp. 239–261, Oct. 2011.

[9] T. J. O'Shea, T. Roy, and T. Erpek, "Spectral detection and localization of radio events with learned convolutional neural features," in *Proc. 25th European Signal Process. Conf. (EUSIPCO 2017)*, Aug. 2017, pp. 331–335.

[10] E. Almazrouei, G. Gianini, N. Almoosa, and E. Damiani, "A deep learning approach to radio signal denoising," in *Proc. IEEE Wireless Comms. and Networking Conf. (WCNC 2019)*, Apr. 2019, pp. 1–8.

[11] X. Mankun, P. Xijian, L. Tianyun, and X. Mantian, "A new time-frequency spectrogram analysis of FH signals by image enhancement and mathematical morphology," in *Proc. 4th Int'l. Conf. Image and Graphics (ICIG 2007)*, Aug. 2007, pp. 610–615.

[12] Y. He, Y. Su, Y. Chen, Y. Yu, and X. Yang, "Double window spectrogram difference method: A blind estimation of frequency-hopping signal for battlefield communication environment," in *Proc. 24th Asia-Pacific Conference on Communications (APCC 2018)*, Nov. 2018, pp. 439–443.

[13] J. Kim and J. G. Andrews, "Sensitive white space detection with spectral covariance sensing," *IEEE Trans. Wireless Commun.*, vol. 9, no. 9, pp. 2945–2955, Sep. 2010.

[14] W. M. Lees, A. Wunderlich, P. J. Jeavons, P. D. Hale, and M. R. Souryal, "Deep learning classification of 3.5-GHz band spectrograms with applications to spectrum sensing," *IEEE Trans. Cognitive Commun. and Networking*, vol. 5, no. 2, pp. 224–236, Jun. 2019.

[15] S. Kojima, K. Maruta, Y. Feng, C.-J. Ahn, and V. Tarokh, "CNN based joint SNR and Doppler shift classification using spectrogram images for adaptive modulation and coding," *IEEE Trans. Commun.*, vol. 69, no. 8, pp. 5152–5167, Aug. 2021.

[16] K. Igwe, O. Oyedum, A. Aibinu, M. Ajewole, and A. Moses, "Application of artificial neural network modeling techniques to signal strength computation," *Heliyon*, vol. 7, no. 3, p. e06047, Mar. 2021.

[17] K. Yang, Z. Huang, X. Wang, and F. Wang, "An SNR estimation technique based on deep learning," *Electronics*, vol. 8, no. 10, p. 1139, Oct. 2019.

[18] O. H. Toma and M. López-Benítez, "Traffic learning: A deep learning approach for obtaining accurate statistical information of the channel traffic in spectrum sharing systems," *IEEE Access*, vol. 9, pp. 124 324–124 336, Sep. 2021.

[19] M. M. Alammar and M. López-Benítez, "Evaluation of the impact of thresholding and frequency/time resolution on signal area estimation methods," in *2021 IEEE 93rd Vehicular Technology Conference (VTC2021-Spring)*. IEEE, 2021, pp. 1–7.

[20] H. Li, M. Syed, Y.-D. Yao, and T. Kamakaris, "Spectrum sharing in an ISM band: Outage performance of a hybrid DS/FH spread spectrum system with beamforming," *EURASIP Journal on Advances in Signal Processing*, vol. 2009, no. 834527, pp. 1–11.

[21] M. López-Benítez and F. Casadevall, "On the spectrum occupancy perception of cognitive radio terminals in realistic scenarios," in *Proceedings of the 2nd IAPR International Workshop on Cognitive Information Processing (CIP 2010)*, Jun. 2010, pp. 99–104.

[22] E. N. Skomal, *Man-made radio noise*. Van Nostrand Reinhold, 1978.

[23] R. J. Matheson, "Measurements of electromagnetic noise radiated from automotive ignition systems," National Telecommunications and Information Administration (NTIA), Tech. Rep. 80-54, Nov. 1980.

[24] A. D. Spaulding and R. T. Disney, "Man-made radio noise, part 1: Estimates for business, residential, and rural areas," Office of Telecommunications (OT), Tech. Rep. 74-38, Jun. 1974.

[25] R. J. Achatz and R. A. Dalke, "Man-made noise power measurements at VHF and UHF frequencies," National Telecommunications and Information Administration (NTIA), Tech. Rep. 02-390, Dec. 2001.

[26] R. Dalke, R. Achatz, Y. Lo, P. Papazian, and G. Hufford, "Measurement and analysis of man-made noise in VHF and UHF bands," in *Proc. Wireless Communications Conf.*, Aug. 1997, pp. 229–233.

[27] P. Constantinou, D. Apostolakis, and M. Katsikis, "Man made noise measurements," in *Proc. 41st IEEE Vehicular Technology Conf. (VTC 1991)*, May 1991, pp. 475–476.

[28] C. Smith, Q. R. Black, and M. Magee, "Computer vision for improved single-sensor spectrum sensing," in *Proc. Sensor Signal Processing for Defence (SSPD 2012)*. IET, Sep. 2012, pp. 1–5.

[29] S. Phonsri, S. S. Mukherjee, and M. Sellathurai, "Computer vision and bi-directional neural network for extraction of communications signal from noisy spectrogram," in *Proc. IEEE Conf. Antenna Measurements and Applications (CAMA 2015)*. IEEE, Dec. 2015, pp. 1–4.

[30] X. Zha, H. Peng, X. Qin, G. Li, and S. Yang, "A deep learning framework for signal detection and modulation classification," *Sensors*, vol. 19, no. 18, pp. 1–21, Sep. 2019.

[31] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: single shot multibox detector," in *Proc. 14th European Conf. Computer Vision (ECCV 2016)*. Springer, Oct. 2016, pp. 21–37.

[32] W. Li, K. Wang, and L. You, "A deep convolutional network for multitype signal detection and classification in spectrogram," *Mathematical Problems in Engineering*, vol. 2020, pp. 1–16, Sep. 2020.

[33] Y. Zhou, Y. Feng, V. Tarokh, V. Gintautas, J. McClelland, and D. Garagic, "Multi-level mean-shift clustering for single-channel radio frequency signal separation," in *Proc. IEEE 29th Int'l. Works. Machine Learning for Signal Processing (MLSP 2019)*. IEEE, Oct. 2019, pp. 1–6.

[34] J. Kokkoniemi and J. Lehtomäki, "Spectrum occupancy measurements and analysis methods on the 2.45 GHz ISM band," in *2012 7th international ICST conference on cognitive radio oriented wireless networks and communications (CROWNCOM)*. IEEE, Jun. 2012, pp. 285–290.

[35] L. D. Stefano and A. Bulgarelli, "A simple and efficient connected components labeling algorithm," in *Proc. 10th Int'l. Conf. Image Analysis and Processing (ICIAP 1999)*, Sep. 1999, pp. 1–6.

[36] D. W. Capson, "Performance comparisons of contour extraction algorithms," *IEEE Trans. Instrum. and Measurement*, vol. IM-35, no. 4, pp. 409–417, Dec. 1986.

[37] K. Umebayashi, H. Iwata, J. J. Lehtomäki, and M. López-Benítez, "Study on simple signal area estimation for efficient spectrum measurements," in *Proc. 26th European Conf. Networks and Commun. (EuCNC 2017)*, Jun. 2017, pp. 1–5.

[38] K. Umebayashi, K. Moriwaki, R. Mizuchi, H. Iwata, S. Tiiro, J. J. Lehtomäki, M. López-Benítez, and Y. Suzuki, "Simple primary user signal area estimation for spectrum measurement," *IEICE Trans. Commun.*, vol. E99-B, no. 2, pp. 523–532, Feb. 2016.

[39] R. Mizuchi, K. Umebayashi, J. Lehtomäki, and M. López-Benítez, "A study on FFT-ED based signal area estimation for spectrum awareness," in *Proc. Int'l. Workshop on Smart Wireless Commun. (SmartCom 2016)*, IEICE Tech. Rep., vol. 116, no. 29, SR2016-9, pp. 27-34, May 2016.

[40] R. Mizuchi, K. Umebayashi, J. J. Lehtomäki, and M. López-Benítez, "A study on false alarm cancellation for spectrum usage measurements," in *Proc. IEEE Wireless Commun. and Networking Conf. (WCNC 2017), 3rd Int'l. Works. Smart Spectrum (IWSS 2017)*, Mar. 2017, pp. 1–6.

[41] M. M. Alammar and M. López-Benítez, "A minesweeper algorithm for improved signal area estimation in spectrum aware systems," in *Proc. IEEE 28th Int'l. Conf. Telecommun. (ICT 2021)*, Jun. 2021, pp. 1–6.

[42] H. Zhou, J. Wu, and J. Zhang, *Digital image processing: part I*. Bookboon, 2010.

[43] M. M. Alammar and M. López-Benítez, "Automatic extraction of signal areas from radio spectrograms based on the Hough transform," in

Proc. IEEE Int. Symp. on a World of Wireless, Mobile and Multimedia Networks (WoWMoM 2022), Jun. 2022, pp. 1–10.

[44] ——, "Enhanced signal area estimation based on edge detection and flood fill," *IEEE Access*, vol. 10, pp. 47 179–47 194, 2022.

[45] S. E. Umbaugh, *Digital image processing and analysis: human and computer vision applications with CVIPtools*. CRC press, 2010.

[46] S. Kayraklik, Y. Alagöz, and A. F. Coşkun, "Application of object detection approaches on the wideband sensing problem," in *Proc. IEEE International Black Sea Conference on Communications and Networking (BlackSeaCom 2022)*, Jun. 2022, pp. 341–346.

[47] M.-C. Su and C.-H. Hung, "A neural-network-based approach to detecting rectangular objects," *Neurocomputing*, vol. 71, no. 1-3, pp. 270–283, 2007.

[48] Y. H. Hu and J.-N. Hwang, *Handbook of Neural Network Signal Processing*, 1st ed. CRC Press, oct 2018.

[49] P. Alapuranen and J. Schroeder, "Complex artificial neural network with applications to wireless communications," *Digital Signal Processing*, vol. 119, p. 103194, Dec. 2021.

[50] M. López-Benítez and F. Casadevall, "Improved energy detection spectrum sensing for cognitive radio," *IET Communications*, vol. 6, no. 8, pp. 785–796, May 2012.

[51] ——, "Versatile, accurate, and analytically tractable approximation for the Gaussian Q-function," *IEEE Transactions on Communications*, vol. 59, no. 4, pp. 917–922, Apr. 2011.

[52] K.-L. Du and M. N. S. Swamy, *Recurrent Neural Networks*. London: Springer London, Sep. 2019, pp. 351–371.

[53] X. Tong, B. Sun, J. Wei, Z. Zuo, and S. Su, "Eaau-net: Enhanced asymmetric attention u-net for infrared small target detection," *Remote Sensing*, vol. 13, no. 16, p. 3200, Aug. 2021.

[54] D. Powers, "Evaluation: From precision, recall and F-measure to ROC, informedness, markedness and correlation," *Journal of Machine Learning Technologies*, vol. 2, pp. 37–63, Jan. 2011.

[55] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. J. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Józefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. G. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. A. Tucker, V. Vanhoucke, V. Vasudevan, F. B. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, "Tensorflow: Large-scale machine learning on heterogeneous distributedm systems," *CoRR*, vol. abs/1603.04467, 2016. [Online]. Available: http://arxiv.org/abs/1603.04467

[56] F. Chollet, "Keras, github," 2015. [Online]. Available: https://github.com/fchollet/keras

[57] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Eds., vol. 32. Curran Associates, Inc., 2019, pp. 8026–8037.

[58] S. F. M. Radzi, M. K. A. Karim, M. I. Saripan, M. A. A. Rahman, I. N. C. Isa, and M. J. Ibahim, "Hyperparameter tuning and pipeline optimization via grid search method and tree-based automl in breast cancer prediction," *Journal of Personalized Medicine*, vol. 11, no. 10, p. 978, Sep. 2021.

[59] E. Bakshy, L. Dworkin, B. Karrer, K. Kashin, B. Letham, A. Murthy, and S. Singh, "AE: A domain-agnostic platform for adaptive experimentation," in *Conference on Neural Information Processing Systems*, 2018, pp. 1–8.

[60] H. J. A. M. Heumans, "Mathematical morphology: A modern approach in image processing based on algebra and geometry," *Society for Industrial and Applied Mathematics*, vol. 37, no. 1, pp. 1–36, Mar. 1995.

[61] P. Soille, *Morphological Image Analysis: Principles and Applications*, 2nd ed. Springer-Verlag, 1999.

[62] R. C. González and R. E. Woods, *Digital Image Processing*. Prentice Hall, 2002.

[63] J. Parker, *Algorithms for Image Processing and Computer Vision*, 2nd ed. Wiley, 2011.