DeepTxFinder: Multiple Transmitter Localization by Deep Learning in Crowdsourced Spectrum Sensing

Anatolij Zubow*, Suzan Bayhan†, Piotr Gawłowicz*, Falko Dressler*
*Technische Universität Berlin, Germany
†University of Twente, The Netherlands
{zubow, gawlowicz, dressler}@tkn.tu-berlin.de, s.bayhan@utwente.nl

***Abstract*—As the radio spectrum has become the bottleneck resource with increasing volume of mobile data and ultra-dense network deployments, it is crucial to use spectrum more flexibly in time, space, and frequency dimensions. However, higher efficiency in spectrum usage facilitated by flexible spectrum allocation comes with a cost, namely the increased complexity of spectrum monitoring and management. Identifying the transmitters is at the interest of particularly spectrum enforcement authorities to ensure that spectrum is used as intended by the legitimate users of the spectrum. For a scalable, efficient, and highly-accurate operation, we propose a crowd-sensing based solution where sensing devices report their measured receive power levels to a central entity which later fuses the collected information for localizing an unknown number of transmitters. Our solution, referred to as DeepTxFinder, leverages deep learning to handle many sources of uncertainty in the operation environment: namely number of transmitters, their transmission power levels, and channel conditions (shadowing). Using deep-learning, DeepTxFinder distinguishes itself from the prior state-of-the art which requires knowledge of the number and transmission power of transmitters or require the transmitters to be well separated in space by tens to hundreds of meters making them ill-suited for application in expected ultra-dense deployment of small-cells. Moreover, we propose a tiling-based approach to increase the scalability of our proposal by reducing the computational complexity. Our simulation studies show that DeepTxFinder can provide a high detection accuracy even only by collecting data from a very small number of sensors. More specifically, with 1 %–2 % sensor density DeepTxFinder can estimate the number of transmitters and their locations with high probability which proves that sparse sensing is feasible.**

I. Introduction

One of the promises of the fifth generation of wireless networks (5G) is the flexibility in spectrum allocation and use. For example, in sharp contrast to traditional spectrum allocation in a nation-wide scale and for tens of years, more dynamic allocation of the spectrum at a finer granularity and for shorter time periods is considered [1]–[3]. Given that spectrum has become the bottleneck resource due to the increasing mobile data volumes and as a consequence of inefficiency of static spectrum allocation, this flexibility can facilitate realizing many benefits that 5G is expected to deliver. For example, private industrial networks can manage their services with guaranteed performance bounds, without depending on the traditional wireless network operators, using the spectrum license acquired for their operation area, e.g., an automotive factory. Recently in Finland and Sweden, a micro-operator has started to provide service for industrial private networks, e.g., shipping ports, using the leased spectrum from a mobile network operator.

On the other hand, the resulting fragmentation in the spectrum, both in temporal and spatial domain, leads to a significant challenge of spectrum management. Both the network operators and the spectrum enforcement authorities need to monitor spectrum usage to make sure that the spectrum is used as intended by the legitimate owners. In addition to intentional misuse of the spectrum, unintentional violation of spectrum rights might occur due to misconfigurations or system bugs. The latter might be experienced even more frequently due to the increasing trend toward software-defined radios (SDR) platforms. Current spectrum misuse detection schemes are laborious, operate reactively based on complaints, and requires an expert physically being present in the area of the interest [4]. However, with increasing spectrum flexibility, providing a scalable, efficient, and highly-accurate misuse detection scheme is crucial. Our goal in this paper is to develop a misuse detection scheme possessing these merits.

In this paper, we introduce deep-learning based transmitter finder, DeepTxFinder for short, which leverages the existing wireless devices to populate sensing data from the area of the interest and uses machine learning to predict the transmitters' locations. The transmitters who are localized and are not identified as the legitimate ones are then marked as *spectrum misusers*. A further closer look might be implemented by the regulatory body on these transmitter locations, which is out of scope of this paper. The crowd-sensing approach employed by DeepTxFinder offers scalability whereas deep learning approach ensures efficiency and high accuracy even under sparse spectrum sensing observations, i.e., small number of sensing devices. Since DeepTxFinder desires to be scalable, it does not require special hardware on the sensing devices and uses only received-signal-strength (RSS) information as the sensing report from a sensing device.

While there is a few works in the literature based on crowd-sourced transmitter localization [5]–[9], unfortunately, their application is limited to very specific, in fact unrealistic, cases where the transmitters are separated with a significant distance of tens or hundreds of meters and all the radio channel characteristics as well as the number of transmitters and their transmit power values are known at the localization entity. More generic frameworks such as BigSpec [10], similar to DeepTxFinder, strive for scalability as spectrum sensors generate a massive amount of data that has to be processed in near-real time. The closest works to ours are [11] and [12] which operate without knowing the number transmitters and their transmission powers. Our solution differs from these two works in that we design a deep-learning based approach. To the best of our knowledge, ours is the first study applying deep learning for multiple transmitter localization.

DeepTxFinder differs from prior work as follows:

- **Data-driven:** It takes a data-driven approach to localize an unknown number of transmitters which can transmit at different power levels and under different radio conditions even under strong shadowing. In the training phase, the neural network (NN) needs merely coarse information about channel propagation, namely the slope of distance-dependent pathloss. As a result, DeepTxFinder is very robust against uncertainties in shadowing and transmit power levels.
- **Zero knowledge:** DeepTxFinder does not require knowledge of the number of transmitters. Moreover, DeepTxFinder can distinguish between two transmitters even if they are very close to each other, e.g., five meters in our simulations. This is a significant improvement over the existing solutions such as [5] which can identify multiple transmitters only if they are separated by at least a few hundred meters. Note, that a robust operation under such dense settings is essential given the increasing density of deployments, e.g., ultra-dense networks.
- **Scalability:** DeepTxFinder works in very large environments, which is achieved by the adoption of a state-of-the-art solution to our problem. Specifically, DeepTxFinder divides the operation area into spatial tiles and processes them before fusing together. By tuning the number of tiles, DeepTxFinder can achieve the desired balance between the accuracy and its training and run-time complexity.

II. Related Work

RSS-based multiple transmitter localization: To keep sensing process simple, RSS-based crowdsensing schemes require the sensing entities to report only their perceived received signal strength values. Examples of this approach are SPLOT [6], Quasi-EM [5], and several others listed in [7]. SPLOT [6] identifies an unknown number of spectrum offenders using crowdsensing by first identifying possible transmitter locations that correspond to local maxima of spectrum measurements. As our performance analysis shows SPLOT might lead to high false alarms in the presence of complex radio propagation conditions, i.e., shadowing, due to many local maximas raising the question on the applicability of SPLOT to these cases. Quasi EM [5] is an iterative approach finding the transmitter locations with the knowledge of the exact number of transmitters and the transmission power. Ureten et al. [7] present an overview of single and multiple transmitter localization methods such as nearest neighbour or Kriging-based interpolation. Both approaches presented in [7] and Quasi EM [5] assume that the number of transmitters is known. Moreover, both approaches require certain transmitter model (e.g., channel propagation model) and knowledge of this model, e.g., the transmitters are separated by at least a distance of 300 meters as these works aim at localizing legacy transmitters which are base stations of a cellular network.

Similar to our work, [12]–[15] do not assume knowledge of the number of transmitters. [12] proposes to localize transmitters based on finding the most-likely hypothesis that would result in the observed signal distribution. A hypothesis represents a case where the transmitter is at a specific location and transmitting with a specific power level. While operating without channel knowledge is a merit of [12], hypotheses testing in [12] requires to build a model of all joint-probability distributions of observations at two different sensors for each hypothesis. Moreover, the number of hypothesis and thereby the complexity increases with the increase in number transmitters (exponentially) and considered area (linearly). Finally, Joneidi et al. [11] propose to use directional antennas for transmitter localization which requires more expensive sensing devices and is hard to apply to crowd-sensing environments due to the need of calibration of the sensing devices.

Spectrum misuse detection: A second line of work to identify spectrum misuse without localizing the spectrum offenders suggests using *spectrum permits* [16], [17]. In this approach such as Gelato [18], each transmitter embeds a spectrum-use authorization in its transmitted signal so that the receivers (e.g., crowdsourced sensors or a managed infrastructure) decode the signal and verify the legitimacy of the transmitter. In case the receivers cannot verify, then the signal is marked as an illegitimate one. While spectrum permits are supposed to be for one-time use only, another approach [19], [20] in this spirit suggests that the spectrum enforcement authority assigns a unique identity to each transmitter which will be embedded in every signal the transmitter emits. While these schemes offer detecting the unauthorized transmitters, they do not localize the transmitter. Compared to spectrum permits implemented at physical layer, RSS based transmitter localization is lightweight, e.g., requires neither a permit distribution infrastructure nor permit detection at the sensing devices.

III. System Model

We assume a system as depicted in Figure 1 where a central node, e.g., located at the regulatory body or at a third-party service, is responsible of spectrum misuser identification. Consider M spectrum offenders (transmitters that do not have the spectrum rights) with unknown locations. We denote their 2D locations by $\mathbf{Q} = [Q_1, Q_2, \cdots, Q_M]$ where $Q_i = [x_i, y_i]$. We assume a stationary signal emitted by a misuser for a period long enough to be identified by spectrum sensing. We denote the transmission power for a transmitter by P_{tx}^i. Note that each transmitter might operate at a different power level. There are J spectrum sensing devices at locations denoted by $S = [S_1, \cdots, S_J]$ where $S_j = [x_j, y_j]$. We assume that both the transmitters and the spectrum devices are static during a single sensing round. Since spectrum sensors are crowd devices, they can be anywhere in the considered area of interest. Moreover, we assume that each sensing device is able

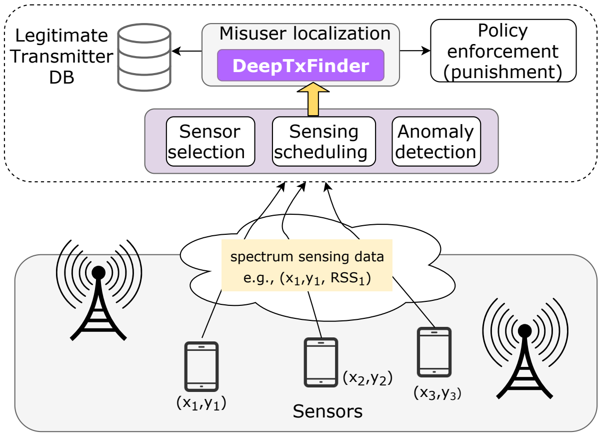


Figure 1. **Spectrum monitoring and misuse detection by crowdsourcing.** DeepTxFinder is part of the solution and its goal is to localize the transmitters according to the sensing data, e.g., received-signal-strength (RSS) measurements, collected from the crowdsensing sensors.

to measure its own location with high accuracy and share it with DeepTxFinder controller.[1]

We assume that the inexpensive sensing devices cannot separate the signal from the transmitters and instead observe the aggregated signal power from all transmitters. Let us denote the received signal power at S_j from transmitter i by $H_{i,j}$. The total signal strength received by S_j is equal to

$$H_j = \sum_{i=1}^{M} H_{i,j}, \tag{1}$$

where we define $H_{i,j}$ as

$$H_{i,j} = \mathrm{PL}(d(i,j)) + x_\sigma, \tag{2}$$

where $d(i,j)$ represents the euclidean distance between i and j, PL is the distance-dependent pathloss, and $x_\sigma \sim N(0,\sigma^2)$ is a normally distributed shadowing term with variance σ^2 which represents shadowing due to obstacles, e.g., trees.

The information collected at the controller is then the *observation vector* which is a vector of sensing outcome labeled with the sensor's location: $\mathcal{H} = [< H_j, x_j, y_j >]$. A transmitter localization scheme denoted by f can be described as $f(\mathcal{H}) = \mathbf{Q'}$ and $\mathbf{Q'} = [Q_0^{'}, \cdots, Q_k^{'}]$ where $Q_k^{'}$ denotes the location of the k$^{\text{th}}$ transmitter identified by the localization scheme. Note that f might not identify all the transmitters (referred to as *misdetection*) or might falsely detect the existence of a transmitter (referred to as *false alarms*). Comparing the ground truth $\mathbf{Q}$ with the predicted locations $\mathbf{Q'}$, we can calculate the following metrics [6], [21]:

- *Cardinality error (ϵ_c)*: It is the difference between the number of actual transmitters and the claimed ones independent of the accuracy of the locations. We report both absolute values for cardinality error, i.e., $\epsilon_c(\mathbf{Q},\mathbf{Q'}) = abs(|\mathbf{Q}| - |\mathbf{Q'}|)$ and the signed cardinality error to have more insights on false alarms and misdetections.
- *False alarm and detection probability*: False alarm is the probability that a transmitter location is a false alarm calculated as $\epsilon_{fa} = (|\mathbf{Q}| - |\mathbf{Q'}|)/|\mathbf{Q'}|$ when $|\mathbf{Q'}| > |\mathbf{Q}|$. Misdetection probability (ϵ_{md}) is calculated similarly: $\epsilon_{md} = (|\mathbf{Q}| - |\mathbf{Q'}|)/|\mathbf{Q}|$ when $|\mathbf{Q'}| \leqslant |\mathbf{Q}|$. Note that false alarms might be more tolerable compared to misdetections. However, false alarms should be minimized to save from laborious task of actually spotting the transmitter in the proximity of the reported locations.
- *Localization error (L_{err})*: It is the minimum error considering all the possible combinations of mapping between the actual and predicted locations. The error calculated between a transmitter's actual and predicted location is the root mean squared error of the distance between these two points. The localization error is then the minimum of the average error observed over all the mappings. Note that if the number of detected and actual locations do not match, we take all permutations of the size equal to the size of the smaller set.

IV. DeepTxFinder: Deep-Learning based Transmitter Finder

In this section, we describe our approach for the localization of multiple transmitters from the sensing data provided by the crowdsensing devices. Multiple transmitter localization becomes challenging for several reasons [6]. First, the number of transmitters is unknown. Moreover, the transmitters might transmit with different power levels. The observation at a sensor will be a superposition of signals from multiple transmitters where each signal will be affected by different channel propagation characteristics, i.e., different distance and shadowing conditions. Since the receivers cannot separate these signals, RSS-based multiple transmitter localization (MTL) becomes a hard problem. However, compared to other approaches such as time-of-arrival (TOA) or angle-of-arrival (AOA), RSS-based localization requires neither the knowledge of the signal structure as in TOA and AOA nor the multiple antennas as in AOA methods. Therefore, our goal is to design a low-cost and accurate MTL approach that can locate an unknown number of transmitters using only the power measurements provided by the crowd-sensing devices.

More formally, the MTL problem is to determine the location and transmitter power of existing transmitters using the observations collected from a set of sensors [14]. Since there are many uncertainties, we prefer to use a learning-based approach which can handle the complexities emerging as a result of the uncertainties in the number and operation power of transmitters and the channel characteristics between the transmitters and the sensing devices. In the next section, we provide more details on the design of DeepTxFinder.

A. Overview of DeepTxFinder

DeepTxFinder relies on supervised learning and takes a two-step approach as depicted in Figure 2. First, DeepTxFinder predicts the number of transmitters. Using the output from its

[1]For privacy preserving schemes, the sensor locations can include some noise. We do not consider this option, however we reserve the analysis for a future work as we believe it is critical to ensure privacy of the sensors.

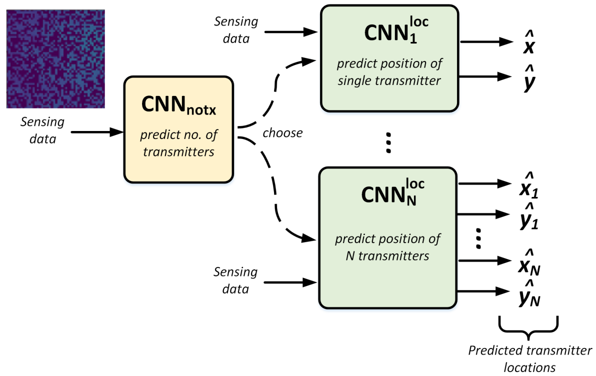


Figure 2. **DeepTxFinder architecture**: two step approach. First CNN is used to detect the number of transmitters, while second CNN estimates actual 2D locations of that many transmitters.

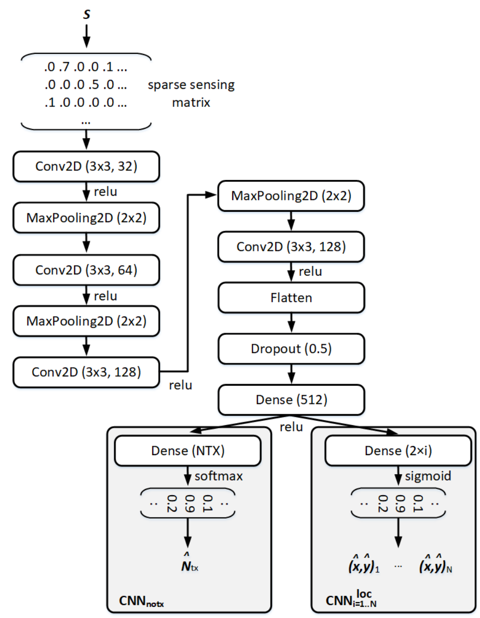


Figure 3. DeepTxFinder CNNs.

first step, DeepTxFinder predicts the locations of *that* many transmitters. For the sake of scalability, DeepTxFinder divides the operation area into smaller grids, e.g., $1\,\mathrm{m} \times 1\,\mathrm{m}$ squares, and then processes the sensing data (represented in power in dBm) to map them to the grids. We refer to this sensing data as *sensing matrix* with entities $\Delta = [\Delta_{o,p}]_{m\times m}$ where $\Delta_{o,p}$ is the receive power in dBm observed in the (o,p)-grid. Since there may not be any sensing data for some of the grid cells, DeepTxFinder sets the sensing data for such grids to a value below the noise floor of the sensing device, e.g., $-100\,\mathrm{dBm}$. Next, the sensing data is normalized to interval $[0,1]$ where 0 represents the cells with missing sensing data: $\Delta_{o,p} = \frac{\Delta_{o,p}+100}{\max \Delta}$.

The sensing matrix is then fed into the first convolutional neural network (CNN) depicted in Figure 2. The output of the first CNN is the number of transmitters which is used to select the respective second CNN. As DeepTxFinder aims at identifying an unknown number of transmitters, we include in the second step $N+1$ many CNNs where N is the maximum number of transmitters expected to be in the operation area. Note that one can calculate N by accounting for the transmitter coverage area at the considered operation spectrum as well as using some historical data on the number of transmitters in an area of interest. Each CNN corresponds to a scenario with a particular number of transmitters, e.g., 0 transmitters to N transmitters. The input to second CNNs is also the sensing matrix while the output is a set of two-dimensional (2D) locations. Comparing the set of these locations with the ground-truth, i.e., the actual 2D locations of the transmitters, we assess the accuracy of DeepTxFinder using the metrics introduced in Section III.

Figure 3 provides a closer look to the design of the CNNs used in DeepTxFinder. We selected CNNs as they are already successfully applied to analyzing visual imagery and we believe our MTL problem is similar. Next, we explain briefly each layer and the rationale of using that layer:

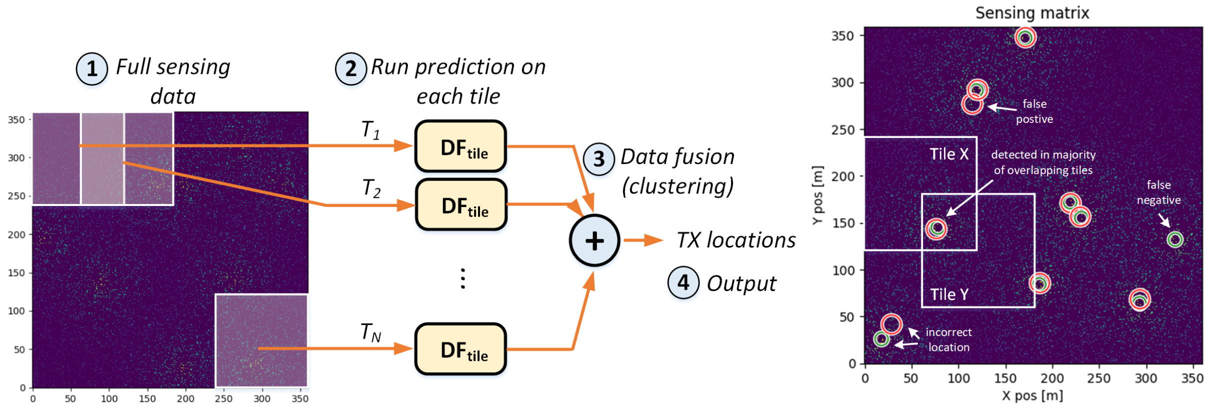
- Conv2D is the core building block of a CNN. The purpose is to let the network learn filters that activate when it detects some specific type of feature at some spatial position in the input, i.e., sensing location with similar receive power value.
- MaxPooling corresponds to a form of non-linear down-sampling. We use it to account for channel shadowing by taking maximum from a set of neighboring grids.
- Dropout layer is used typically to reduce overfitting.
- Dense layer is the regular densely-connected NN layer.
- Flatten layer is needed to flatten the input.

As activation functions, we used the softmax and sigmoid activation functions (cf. Figure 3). Finally, we compile the developed model with the mean squared error regression loss as its objective function.

Since DeepTxFinder considers a maximum number of transmitters denoted by N, it might result in low accuracy for areas of interest with many more transmitters. To prevent poor performance and also to increase the scalability of DeepTxFinder further, DeepTxFinder divides its area into smaller sub-regions referred to as *tiles*, e.g., with dimensions $120\,\mathrm{m} \times 120\,\mathrm{m}$. For each tile, DeepTxFinder executes the above-described transmitter-localization approach. With the tiling approach, we can locate the transmitters even in a large area while keeping the complexity of our deep-learning framework lower and adjustable through setting the tile dimensions. In the next section, we introduce this approach.

B. Prediction in very large environments using tiling

For the sake of lower complexity and memory footprint, we improve our proposal with a tiling-based approach. Rather than running our supervised-learning model on a very-large area, we divide the whole area in smaller regions strategically and run the detection steps introduced earlier in each small region. Recall that our model has a total of $N+1$ CNNs in Figure 2 which will result in a very complex model for very large environments. Instead, we keep our model simple by



(a) Full detector with four steps

(b) Example output of prediction in large environments.

Figure 4. (a)DeepTxFinder divides the area of interest into smaller uniform tiles and runs prediction in each tile. After finding the predicted locations, DeepTxFinder fuses the individual predictions from multiple tiles using a majority voting approach. The predictions identified in the majority of tiles are added to the final set of predicted locations. (b) Green circle is ground truth, red is predicted location as generated by our approach. The sensor density was set to 8 %, i.e., there is no sensing data available for locations colored in dark blue.

having smaller tiles, thereby lower N, and apply a four-step solution.

Figure 4a illustrates the entire approach which consists of four steps. First, the sensing matrix is split into overlapping tiles of $d \times d$ m. Each tile represents the sensing data in only a sub-region of the whole operation area and each tile overlaps with some other tiles. Let assume that the whole operation area is of dimensions $D \times D$ m. Moreover, let us define the overlapping area between two tiles as overlap ratio α which is simply the ratio of the overlapping area to the area of a tile. Dividing this region into tiles with dimensions of $d \times d$ m and having tiles with an overlap ratio of α, we can calculate the number of tiles as

$$n_{tiles} = \left(\frac{D - \alpha \times d}{(1-\alpha) \times d} \right)^2 . \tag{3}$$

Note, that $\alpha = 0.5$ in the example given in Figure 4b.

Let us represent the estimated transmitter locations in each tile by $\mathbf{Q}'_t$ for tile t where $t \in [1, \cdots, n_{tiles}]$.

Next, DeepTxFinder applies a voting scheme among the overlapping tiles to eliminate the locations reported by individual tiles and identify the locations that are reported by the majority of the tiles. Figure 4b shows an example where false alarms and true detections are highlighted. Depending on the reported transmitter location, DeepTxFinder calculates the number of tiles that should detect this transmitter, e.g., a transmitter at the edges of the operation area should be detected by at least two tiles while the transmitters at more central locations must be detected by more tiles. In the detection, DeepTxFinder uses mean shift clustering using a flat kernel from the scikit-learn library [22]. For an estimated transmitter location to be valid, the cluster size must be $T - 1$, i.e., all but one of the overlapping clusters has to agree on the same location where T is the number of tiles covering this location. With this strict voting rule representing almost a consensus, we aim at minimizing false alarms to save the resulting waste of resources of the regulatory body. Additionally, having low false alarms increases the robustness of DeepTxFinder against adversaries who might be interested in resulting in too many false alarms essentially hindering the use of DeepTxFinder.

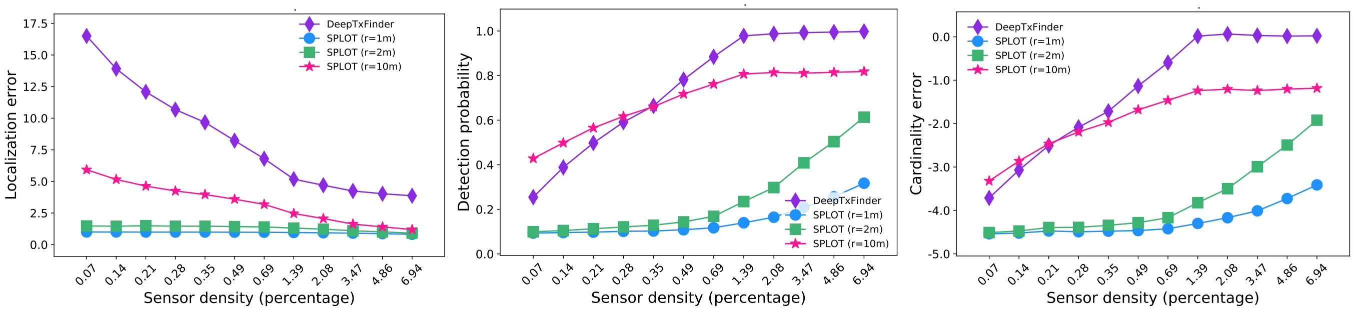
Figure 4b illustrates this with an example of two tiles. Note that the position of a transmitter is only reported if the majority of overlapping tiles decide on similar position. Here the field of size 360 m × 360 m is split up into $5 \times 5 = 25$ overlapping tiles. For each tile, the transmitter locations are estimated independently and finally fused together. From the figure, we observe that DeepTxFinder can detect most of the locations accurately. We have only one false negative, i.e., there is a transmitter but it was not detected. Moreover, we have a false positive, i.e., no transmitter but predictor reported one. Finally, one transmitter was detected but with slightly incorrect location. Note that increasing the number of tiles increases the complexity of the overall detection framework but at the same time can maintain a higher accuracy.

V. Performance Evaluation

The objective of our evaluation is twofold: (i) we assess the accuracy of DeepTxFinder with increasing uncertainty in the knowledge of the transmitter parameters, (ii) we demonstrate the feasibility of deep learning based transmitter localization and compare its performance against the existing model-based solutions. As benchmark, we use SPLOT [6] for our analysis. For the sake of fair comparison, we will consider three variants of SPLOT with three different parameters, SPLOT-1, SPLOT-2, and SPLOT-10. Here the number represents the threshold value used by SPLOT for finding the local maximas.

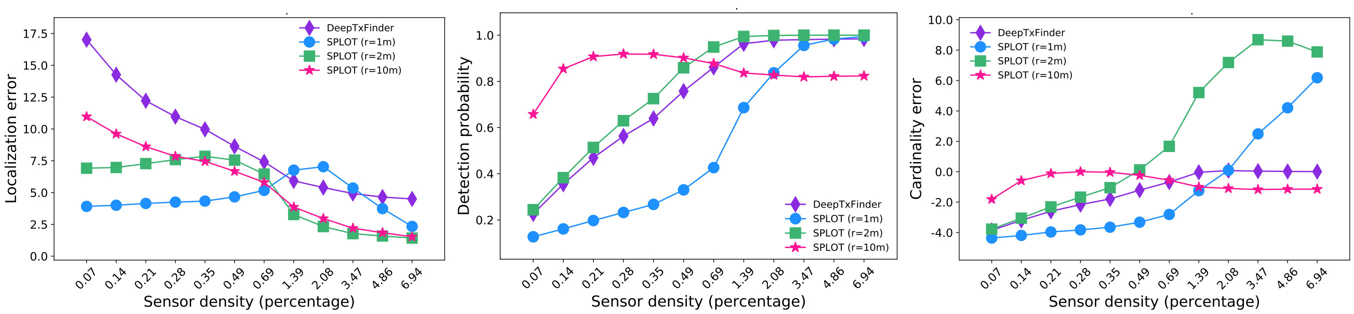
A. Model Training and Testing

We first train our network with a broad range of data collected from our custom-made system-level Python simulator. We deploy a varying number of transmitters randomly and



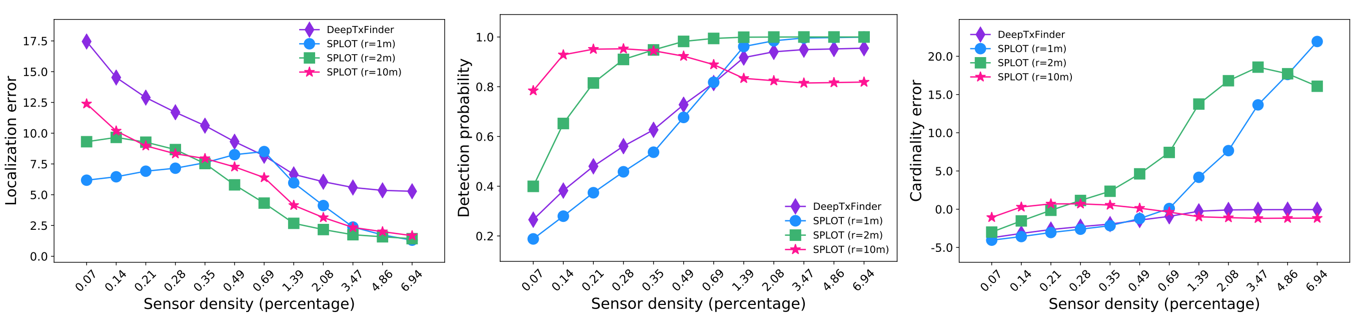
(a) Localization error. (b) Detection probability. (c) Cardinality error.

Figure 5. Scenario I: Performance comparison of SPLOT and DeepTxFinder under no shadowing and constant TX power.



(a) Localization error. (b) Detection probability. (c) Cardinality error.

Figure 6. Scenario II: Performance comparison of SPLOT and DeepTxFinder under shadowing and known transmission power.



(a) Localization error. (b) Detection probability. (c) Cardinality error.

Figure 7. Scenario IV: Performance comparison of SPLOT and DeepTxFinder under shadowing and unknown transmission power.

also consider a wide range of scenarios where the density of sensing devices (θ) varies from very low to very high, e.g., every grid cell has at least one sensing device.

As channel model, we have used spatial correlated shadowing in which two sensing devices within the correlation distance of each other experience similar shadowing conditions [23]. As operation frequency, we consider 900 MHz band and channel bandwidth of 20 MHz. We vary the shadowing parameter σ from 0 dBm–10 dBm. Briefly, our training takes the following parameters as input to generate the sensing matrix: (i) number of transmitters, (ii) transmitter power, (iii) number of sensing devices, (iv) shadowing parameter, (v) shadowing correlation parameter. We labeled each data with the actual number of transmitters (for the first step in Figure 2) and actual locations of the transmitters (for the second step in Figure 2). As our data, we collected 10^5 samples of the sensing matrix. We used 70 % for training and the rest for testing (validation).

B. Scenarios

To test the performance of DeepTxFinder in a wide range of settings, we consider the following scenarios:

- **Scenario-I [no shadowing]:** This scenario represents the simplest case where the channel pathloss is fully deterministic, i.e., depends exclusively on the distance ($\sigma = 0$ dB). Moreover, the transmitter power is constant for all transmitters and DeepTxFinder is aware of the transmitter power level.

- **Scenario II [shadowing]:** This scenario considers a more realistic case where the signal propagation experiences shadowing (where $\sigma = 5\,\mathrm{dB}$). However, the transmitter power is constant for all transmitters and DeepTxFinder is aware of the transmitter power level.
- **Scenario III [shadowing, unknown P_{tx}]:** This scenario represents a more challenging case where the channel has shadowing (with $\sigma = 5\,\mathrm{dB}$) and the transmitter power is not constant, i.e., uniformly distributed between 0 dBm–10 dBm.

C. Results

We first evaluate the impact of increasing sensing device density (θ) under the aforementioned scenarios. Figure 5 shows the localization error in meters, detection probability, and cardinality error for DeepTxFinder and the three versions of SPLOT for Scenario I wherein the transmitter power is constant for all transmitters and the pathloss is deterministic. In Figure 5a, we observe that both DeepTxFinder and SPLOT benefit from increasing sensor density in terms of lower localization error. Second, both schemes can converge to acceptable localization errors in the order of a few meters with only 1 %–2 % sensor density which proves that sparse sensing is feasible. Third, we observe that SPLOT has very high accuracy for all settings which is lower than DeepTxFinder. However, a closer look to Figures 5b and 5c shows us that SPLOT has low detection probability and misses most of the transmitters. On the other hand, DeepTxFinder can miss only one or two transmitters when sensor density is between 0.35 to 1.39. When $\theta \geqslant 1.39$, DeepTxFinder detects all transmitters without false alarms as the cardinality error is zero for DeepTxFinder. Finally, the best configuration for SPLOT is $r = 10\,\mathrm{m}$ if we target a trade-off between localization error and detection probability. Despite its higher localization error in this operation regime in comparison to SPLOT with $r = 10\,\mathrm{m}$, under sufficient sensor density, we believe that DeepTxFinder offers a more reliable approach than the state-of-the-art at the expense of a slight increase, e.g., 3 m, in localization error.

When we consider shadowing as in Scenario II, SPLOT starts to have higher localization error but also higher detection probability as shown in Figures 6a and 6b. Meanwhile, performance of DeepTxFinder remains almost the same showing its robustness against different environment conditions and thereby its feasibility in a wide range of settings. SPLOT with $r = 10\,\mathrm{m}$ still provides the best trade-off between localization error and detection probability in this scenario as others result in a very high positive cardinality error, i.e., high false alarms. The high cardinality error is due to the existence of many local maximas under shadowing and SPLOT considering local maximas as possible transmitter locations. Initially, under lower sensor density, SPLOT variants cannot detect some of the transmitters due to sparse sensor observations. But, with increasing sensor density, SPLOT-variants with $r = 1$ and $r = 2$ start to give frequent false alarms, which is undesirable as it will lead to waste of human expert labor.

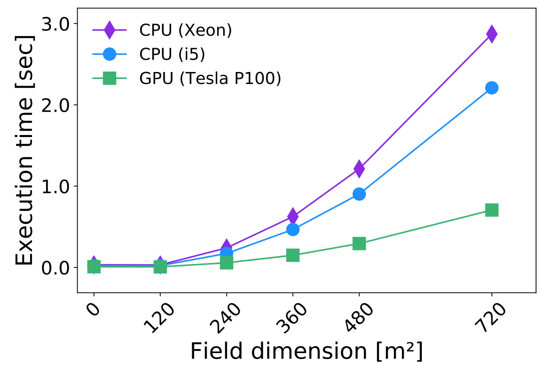


Figure 8. Execution performance DeepTxFinder.

Finally, Figure 7 confirms our earlier conclusions. In this case, we observe that DeepTxFinder maintains a lower detection probability if the transmitter power is randomly distributed between 0 dB–10 dB. In earlier scenarios, DeepTxFinder could achieve 100 % detection probability whereas in Scenario III it converges to 90 %. In this considered setting with $\theta \geqslant 1.39$, SPLOT-10's detection probability is around 80 %, while its localization error is 4 m lower than that of DeepTxFinder.

Finally, we assess the running time of DeepTxFinder. Figure 8 shows the execution time of DeepTxFinder for different field sizes on state-of-the-art machines: i) Intel i5 (3.5GHz from 2013), ii) Intel Xeon CPU E5-2640 v4 with 2.40GHz, 20 cores, 250 GByte RAM and iii) same as ii) but with graphic card (GPU) support: Tesla P100, 12 GB RAM. We clearly see the speedup with GPU.

VI. Conclusions

With increasing flexibility of radio spectrum use, there is an increasing need for identifying the source of transmissions and localizing them to prevent illegitimate spectrum use. While crowdsensing the spectrum offers scalability, it is not always possible to have a dense sensor setup. Hence, in this work, our focus was on transmitter localization under sparse spectrum sensing observations. To cope with the uncertainty in the number of transmitters and their transmission power levels, we have proposed a deep learning based solution to localize an unknown number of transmitters. Comparing our work against the state-of-the-art [6], we observe that DeepTxFinder works well even in environments with strong shadowing and unknown transmission power of the source to be localized. This is in contrast to the benchmark which is sensitive to changes in the environment. Moreover, our proposal does not result in high false alarms, which is essential to avoid waste of expert labour (e.g., officers at the regulatory body) to inspect the source of the transmission. As future work, we plan to extend our approach so that it can estimate the transmission power of the localized transmitters. Moreover, we plan to evaluate the impact of tile sizes on the localization performance and running time.

Acknowledgment

We would like to thank L. Sander for his contribution to implementation and evaluation of SPLOT, and Felix Knopp, Stefan Wahl, and Philipp Wisznawitzki for their contribution in the earlier phases of the project.

References

[1] ETSI, "Reconfigurable Radio Systems (RRS); Feasibility study on temporary spectrum access for local high-quality wireless networks," ETSI, TR 103 588 V1.1.1, Feb. 2018.

[2] M. Matinmikko, M. Latva-aho, P. Ahokangas, and V. SeppÃ¤nen, "On regulations for 5G: Micro licensing for locally operated networks," *Telecommunications Policy*, vol. 42, no. 8, pp. 622–635, Sep. 2018.

[3] S. Bayhan, G. Gür, and A. Zubow, "The Future is Unlicensed: Coexistence in the Unlicensed Spectrum for 5G," arXiv, cs.NI 1801.04964, Jan. 2018.

[4] Z. Li, Z. Xiao, B. Wang, B. Y. Zhao, and H. Zheng, "Scaling Deep Learning Models for Spectrum Anomaly Detection," in *20th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc 2019)*. Catania, Italy: ACM, Jul. 2019.

[5] J. K. Nelson, M. R. Gupta, J. E. Almodovar, and W. H. Mortensen, "A Quasi EM Method for Estimating Multiple Transmitter Locations," *IEEE Signal Processing Letters*, vol. 16, no. 5, pp. 354–357, May 2009.

[6] M. Khaledi, M. Khaledi, S. Sarkar, S. Kasera, N. Patwari, K. Derr, and S. Ramirez, "Simultaneous Power-Based Localization of Transmitters for Crowdsourced Spectrum Monitoring," in *23rd ACM International Conference on Mobile Computing and Networking (MobiCom 2017)*. Snowbird, UT: ACM, Oct. 2017.

[7] S. Ureten, "Single and Multiple Emitter Localization in Cognitive Radio Networks," PhD Thesis, University of Ottawa, Jan. 2017.

[8] A. Nika, Z. Li, Y. Zhu, Y. Zhu, B. Y. Zhao, X. Zhou, and H. Zheng, "Empirical Validation of Commodity Spectrum Monitoring," in *14th ACM Conference on Embedded Network Sensor Systems (SenSys 2016)*. Stanford, CA: ACM, Nov. 2016.

[9] M. Z. Zheleva, R. Chandra, A. Chowdhery, P. Garnett, A. Gupta, A. Kapoor, and M. Valerio, "Enabling a Nationwide Radio Frequency Inventory Using the Spectrum Observatory," *IEEE Transactions on Mobile Computing (TMC)*, vol. 17, no. 2, pp. 362–375, Feb. 2018.

[10] Y. Zeng, V. Chandrasekaran, S. Banerjee, and D. Giustiniano, "A Framework for Analyzing Spectrum Characteristics in Large Spatio-temporal Scales," in *25th ACM International Conference on Mobile Computing and Networking (MobiCom 2019)*. Los Cabos, Mexico: ACM, Oct. 2019.

[11] M. Joneidi, H. Yazdani, A. Vosoughi, and N. Rahnavard, "Source Localization and Tracking for Dynamic Radio Cartography using Directional Antennas," in *19th IEEE International Conference on Sensing, Communication, and Networking (SECON 2019)*. Boston, MA: IEEE, Jun. 2019.

[12] A. Bhattacharya, C. Zhan, H. Gupta, S. R. Das, and P. M. Djuric, "Selection of Sensors for Efficient Transmitter Localization," in *39th IEEE Conference on Computer Communications (INFOCOM 2020)*. Toronto, Canada: IEEE, Jul. 2020.

[13] R. Vaze and C. R. Murthy, "Multiple Transmitter Localization and Whitespace Identification Using Randomly Deployed Binary Sensors," *IEEE Transactions on Cognitive Communications and Networking (TCCN)*, vol. 2, no. 4, pp. 358–369, Dec. 2016.

[14] C. Zhan, A. Bhattacharya, H. Gupta, and M. Ghaderibaneh, "Efficient Localization of Multiple Intruders for Shared Spectrum System," in *19th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN 2020)*. Sydney, Australia: ACM, Apr. 2020.

[15] M. Ghaderibaneh, M. Dasari, and H. Gupta, "Multiple Transmitter Localization under Time-Skewed Observations," in *IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN 2019)*. Newark, NJ: IEEE, Nov. 2019.

[16] X. Jin, J. Sun, R. Zhang, Y. Zhang, and C. Zhang, "SpecGuard: Spectrum Misuse Detection in Dynamic Spectrum Access Systems," *IEEE Transactions on Mobile Computing (TMC)*, vol. 17, no. 12, pp. 2925–2938, Dec. 2018.

[17] X. Zhang, P. Huang, Q. Jia, and L. Guo, "CREAM: Unauthorized Secondary User Detection in Fading Environments," in *15th IEEE International Conference on Mobile Ad Hoc and Sensor Systems (MASS 2018)*. Chengdu, China: IEEE, Oct. 2018.

[18] A. Nika, Z. Zhang, B. Y. Zhao, and H. Zheng, "Toward Practical Spectrum Permits," *IEEE Transactions on Cognitive Communications and Networking (TCCN)*, vol. 3, no. 1, pp. 112–122, Mar. 2017.

[19] V. Kumar, H. Li, J.-M. J. Park, and K. Bian, "Crowd-Sourced Authentication for Enforcement in Dynamic Spectrum Sharing," *IEEE Transactions on Cognitive Communications and Networking (TCCN)*, vol. 5, no. 3, pp. 625–636, Sep. 2019.

[20] ——, "Enforcement in Spectrum Sharing: Crowd-sourced Blind Authentication of Co-channel Transmitters," in *IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN 2018)*. Seoul, South Korea: IEEE, Oct. 2018.

[21] J. R. Hoffman and R. P. S. Mahler, "Multitarget Miss Distance via Optimal Assignment," *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 34, no. 3, pp. 327–336, May 2004.

[22] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and É. Duchesnay, "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research*, vol. 12, no. 85, pp. 2825–2830, 2011.

[23] "IEEE 802.16m Evaluation Methodology Document (EMD)," IEEE, Standard 802.16m-08/004r2, Jul. 2008.