Robust estimation of the number of coherent radar signal sources using deep learning

John Rogers | John E. Ball | Ali C. Gurbuz

Electrical and Computer Engineering, Mississippi State University, Starkville, Mississippi, USA

Correspondence
John E. Ball, 406 Hardy Road, Sinnall Hall, Room 233, Electrical and Computer Engineering Department, Mississippi State University, Mississippi State, MS 39762.
Email: jeball@ece.msstate.edu

Abstract
A deep-learning-based approach to estimating the number of coherent sources in radar is presented. A proper estimate of the number of sources in a signal enables improved angle-of-arrival (AoA) estimation common in applications such as radar, sonar, and communication systems. Many AoA estimators utilised in these areas require accurate estimates of the number of sources for enhanced performance. Herein, a robust method that performs well under the existence of coherent sources is developed. The proposed method is based on deep learning and it is shown to perform better than state-of-the-art versions of the Akaike Information Criteria (AIC), Minimum Description Length (MDL), and Exponentially Embedded Families (EEF) estimators, which employ spatial smoothing of the covariance matrix to handle coherent signals.

1 | INTRODUCTION

Estimating the number of plane wave sources is an important problem in fields such as radar, sonar, and communication systems. Traditional approaches often rely on the eigenvalues of the covariance matrix, which limits their performance. These methods will fail in the case where there are coherent sources. A deep-learning-based method that utilises spatial smoothing of the covariance matrix information in addition to the eigenvalues to accurately estimate the number of sources is introduced herein. The proposed method achieves enhanced results, and works in situations where typical non-coherent methods such as the Akaike Information Criteria (AIC) estimator, Minimum Description Length (MDL) estimator, Exponentially Embedded Families (EEF) estimator, Multiple Signal Classification (MUSIC), and Minimum Variance Distortionless Response (MVDR) fail. To the best of the authors’ knowledge, this is the first deep learning (DL) network applied to the fusion of covariance data and eigenvalues used to analyse the number of incoming target signals. This is an extension of previous work [1], where new networks and better results have been examined. Specifically, the contributions of are:

1. A robust DL system that achieves state-of-the-art results and surpasses traditional eigenvalue-based methods, which fail in the scenarios examined herein.
2. Fusion of the spatially smoothed covariance matrix and the eigenvalues for joint analysis in a unified DL structure.
3. Enhanced results even under a small number of receivers or pulses in a coherent processing interval.

Section 2 briefly discusses the background information on the existing number of source estimation methods and deep learning. The proposed method is outlined in Section 3. Section 4 details generated data sets for the proposed approach, and the obtained results are presented in Section 5. Finally, Section 6 provides the conclusions and lists future work.

2 | BACKGROUND

2.1 | Nomenclature

Table 1 gives the mathematical symbols used herein.

2.2 | Conventional source estimation methods

In radar signal processing, estimating the number of signals present in noisy data is a difficult problem which has been explored extensively. However, robust solutions in this area are...
TABLE 1 Mathematical symbols and notations

| Variable | Description |
|----------|-------------|
| $x_m$   | Receive data vector for $m$-th pulse |
| $X$     | Receive data matrix |
| $R_{XX}$| True covariance matrix |
| $\hat{R}_{XX}$ | Covariance matrix estimate |
| $R_{xx}^S$ | Smoothed covariance matrix estimate |
| $R_{xx}^S(m,n)$ | Smoothed covariance matrix $m, n$ entry |
| $\lambda_i$ | $i$-th covariance matrix eigenvalue |
| $J_i^S$ | Smoothed covariance matrix $i$-th eigenvalue |
| $P$     | Number of pulses (independent realisations) |
| $M$     | Number of receiver channels |
| $S$     | Size of spatial smoothing subarray |
| $T$     | True number of targets present |
| $\hat{T}$ | Estimated number of targets present |
| $\Delta x$ | Array element spacing (metres) |
| $\phi_k$ | Azimuth angle of $k$-th target |
| $n_m$   | Noise in the $m$-th receive channel |

Still required. It is advantageous for the radar to know how many sources are present in a signal to facilitate improved target detection and tracking. Some common angle of arrival (AoA) estimation algorithms include MUltiple Signal Classification (MUSIC) [2], Estimation of Signal Parameters by Rotational Invariance Techniques (ESPRIT) [3], and the Maximum Likelihood Estimator (MLE) (especially the efficient implementation [4]). More recent approaches based on spatial sparsity of angle domain using compressive sensing [5, 6], smoothed-$\ell_0$ norm [7], sub-Nyquist sampling [8], or quadrilinear decomposition [9] have been also successfully applied to AoA estimation problems.

While subspace methods such as MUSIC or ESPRIT depend on separation of signal and noise subspaces and hence require a number of targets, some other techniques such as MVDR disregard the number of sources and output an estimation of the power spectrum across the angular region the sensor is observing. Methods making the assumption that the number of sources is known a priori may give misleading results if the assumed number of sources is wrong [10]. These techniques specifically are called super-resolution techniques as they can localise more accurately than the Rayleigh resolution [11]. In fact, the capability of resolving two closely spaced sources by an array of sensors is limited by its ability to estimate the number of sources correctly. If the number of sources can be determined more reliably, then we can use this information in both utilising in high-resolution AoA estimation algorithms and the angle spectrum information obtained from these approaches.

MLE requires a prior estimate of the number of sources and, like MUSIC or MVDR, it requires a parameter sweep. Its computational complexity grows exponentially with dimension. Moreover, the super-resolution techniques, in general, require extensive computations and are typically too slow for real-time implementation [12]. MUSIC is also susceptible to poor performance when the source signals are coherent [13, 14], which is the case herein. The MVDR algorithm is both a beamformer and a super-resolution AoA estimator. The MVDR can be utilised to both estimate AoA and to estimate the number of sources, as long as the sources are separated adequately. Sparsity-based AoA approaches solve constrained optimisation problems based on $\ell_1$ [6] or smoothed-$\ell_0$ norms [7], or apply greedy approaches such as orthogonal matching pursuit (OMP) [15]. Selections of constraints or stopping criteria for these techniques either require knowledge of the number of sources or good estimation of noise level in the measurements. Hence estimation of the number of sources is a very important part of most of the existing angle estimation problems.

The AIC [16] is arguably the most widely used method for number of sources estimation in the case of white Gaussian noise (WGN). It determines the number of sources by minimising a criterion over a range of detectable number of sources. However, the AIC is observed to provide inconsistent estimates and often overestimates the number of signals in radar applications [17]. As an alternative, the MDL was proposed in [17], but the MDL can lead to an underestimate of the signal subspace dimension, most commonly when the number of samples is comparably small [18]. Another model order criterion applied to source estimation is the EEF criterion [19]. This method has been demonstrated to outperform the MDL in difficult scenarios such as those with low SNR, closely spaced targets, and a limited number of signals. These methods operate on the eigenvalues of the signal covariance matrix to estimate the dimensionality of the data. More accurate methods have been proposed, such as in [18] which uses a discriminate function on the covariance eigenvalues to estimate the dimensionality of both the signal and noise subspaces. These estimates are combined in another discriminate function to estimate the number of sources.

2.3 | Coherent source estimation methods

The techniques mentioned in Section 2.2 are designed for and work well when data is incoherent. It is observed that the performance of both the number of source estimation and the AoA estimation degrade under coherent data. Additional preprocessing, such as spatial smoothing, is required for coherent signals [20]. The processing of coherent signals has been explored in [13, 20–33]. Spatial smoothing [20] utilises subarrays to separate the signal information. This method divides the receiver array into multiple overlapping subarrays of equal size. The signal covariance matrix is calculated for each subarray and averaged to produce a single covariance matrix for the signal. Variations on spatial smoothing include Du and
Kirtin's method [23] which considers the cross-correlation of the subarray outputs and Hung and Kaveh's method [26] which uses so-called “focussing matrices”. Another method employs an exchange matrix with correlated/coherent signals content and then uses a search over the parameter space and peak-finding [28].

Let a radar receiver have \( M \) channels, and define the \( [M \times P] \) receiver data matrix as \( \mathbf{X} \), where \( M \) is the number of receiver elements and \( P \) is the number of pulses. To estimate the non-spatially smoothed (e.g. standard) covariance matrix, one can use

\[
\mathbf{R}_{XX} = \frac{1}{P} \mathbf{XX}^H,
\]

where the Hermitian matrix transpose is denoted by \( (\cdot)^H \). The covariance matrix will be sized \( [M \times M] \). To estimate the smoothed covariance matrix, we define the \( S \)-element subarrays over the receiver, where the \( m \)-th subarray corresponds to receiver elements \( \{m, m+1, \ldots, m+S-1\} \), and let the \( m \)-th subarray data be \( \mathbf{X}_m \). The spatially smoothed subarray covariance matrix will have size \( [S \times S] \) and is calculated as

\[
\hat{\mathbf{R}}_{XX}^S = \frac{1}{(M-S+1)P} \sum_{m=1}^{M-S+1} \mathbf{X}_m (\mathbf{X}_m)^H
\]

where spatial smoothing is indicated by the superscript \( S \).

### 2.4 | Neural networks and deep learning in radar signal processing

Neural networks (NNs) are loosely meant to mimic neurons in the brain. In a typical shallow NN, each neuron has a number of inputs which are multiplied by weights and a bias term is added to this sum. The linear combination of inputs then passes through a non-linear activation function. Multiple layers allow the NN to learn any complex function of the inputs, provided enough neurons are present and there are at least two layers [34].

NNs are typically trained by backpropagation, which adjusts each weight in the network according to the error criterion in the training function. Several works have developed NN approaches to AoA estimation in radar [10, 12, 14, 35–52]. Herein, we briefly review some of the NN-based AoA estimation papers.

El Zoohgy et al. [39] utilised the covariance matrix and a radial basis function (RBF) NN to estimate the AoA of multiple radar signals. The covariance matrix contains detailed information about the incoming data signals. Du et al. [53] examine several NN architectures for antenna array signal processing: multilayer perceptrons, Hopfield networks, radial-basis function NN (RBFNN), PCA-based NN, and Fuzzy NN. Amari and Cichocki [54] examined adaptive blind signal processing using NN and provided a list of 10 open questions in the field. In [12, 36, 55] RBFNN for signal analysis is utilised. Solazzi et al. [56] developed a spline NN to address blind source separation. These networks are not directly applicable for radar as the radar data is complex. Complex NNs have been studied for about 15 years now [57, 58], but there is little published on radar processing using complex NNs.

DL has gained much attention in various research communities due to the significant performance gains of many DL systems over more standard (hand-crafted) feature-based learning. Deep networks can learn very complicated features and decision boundaries directly from the raw data. In this sense, Grais et al. [59] utilised a deep (five-layer) NN where the initial estimates were generated using non-negative matrix factorisation. Their system identified the data source (source one or source two) in speech processing. Vesperini et al. [60] put forth a DL system that could handle both static and moving sound sources. Although these are not applications with radar signals, they do show that a DL system can perform AoA analysis.

### 3 | PROPOSED METHOD

Based on the ability of spatial smoothing of the covariance matrix to handle coherent signals, and the ability of DL networks to learn very complicated functions, the authors propose a DL network to estimate the number of coherent sources in a signal that is received by an array of sensors. The proposed NN features a deep architecture with inputs constructed from the separated real and imaginary values of the covariance matrix and its eigenvalues. The proposed DL attempts to fuse these inputs in an optimal manner to give a robust estimation of the number of sources in a signal. To the best of the authors’ knowledge, this is the first use of a DL to fuse the covariance matrix and its eigenvalues to estimate the number of sources.

#### 3.1 | Signal model

Herein, the following complex-valued radar signal model is utilised:

\[
x_m = \left[ \sum_{k=1}^{T} a_k \exp \left( -j2\pi \frac{f}{c} \Delta x m \sin(\phi_k) \right) \right] + n_m
\]

where \( j = \sqrt{-1} \), \( x_m \) is the complex received signal at the \( m \)-th receiver, \( T \) is the true number of sources, \( a_k \) is the amplitude of the \( k \)-th source, \( \phi_k \) is the azimuth angle of the \( k \)-th source (in radians), \( \Delta x \) is the element spacing in metres, \( f \) is the radar frequency in Hz, \( c \) is the speed of a wave in the medium in metres/second, and \( n_m \) is the independent and identically distributed (i.i.d.) complex white Gaussian noise associated with the \( m \)-th receive channel (and is independent across channels). The radar collects \( P \) pulses.

The data from pulse \( p \) (snapshot) is stored in the complex vector \( \mathbf{x}_p = [x_1, x_2, \ldots, x_M] \) of size \( [M \times 1] \). The data of all \( P \) pulses is stored in the data matrix \( \mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_P] \) of size \( [M \times P] \). The targets are Swerling 1 targets, with signal-to-noise ratios (SNRs) ranging from 0 to 20 dB.
3.2 | Network input formatting

In order to estimate the number of sources, a combination of the estimates of the covariance matrix itself and the covariance matrix eigenvalues is used herein. For each case analysed, the \([M \times M]\) covariance data matrix is estimated using the same spatial smoothing averaging methods used in [20, 32].

Herein, the spatially smoothed covariance matrix is unrolled to produce a \([2M^2 \times 1]\) feature vector as follows, where \(M = S\):

\[
f_R = \begin{bmatrix}
    \text{real}\{\hat{R}_{XX}^S(1, 1)\} \\
    \text{real}\{\hat{R}_{XX}^S(2, 1)\} \\
    \vdots \\
    \text{real}\{\hat{R}_{XX}^S(M, 1)\} \\
    \text{imag}\{\hat{R}_{XX}^S(1, 1)\} \\
    \text{imag}\{\hat{R}_{XX}^S(2, 1)\} \\
    \vdots \\
    \text{imag}\{\hat{R}_{XX}^S(M, 1)\} \\
    \text{imag}\{\hat{R}_{XX}^S(M, M)\}
\end{bmatrix}.
\]  

(4)

Although the diagonal terms of the covariance matrix should be real, for simplicity, all entries of the covariance matrix are unrolled for the feature vector. Thus, the omission of zero imaginary parts of diagonal elements is an opportunity for optimisation not implemented here. The eigenvalues are computed using the singular value decomposition (SVD) [61] as follows from the covariance matrix:

\[
\hat{R}_{XX}^S = USU^H,
\]

(5)

where \(S\) is the diagonal singular-value matrix whose diagonals are the covariance matrix eigenvalues: \(S = \text{diag}(\lambda_1^S, \lambda_2^S, \ldots, \lambda_M^S)\), where \(\lambda_1^S \geq \lambda_2^S \geq \cdots \geq \lambda_M^S\). The eigenvalue features are placed in the \([M \times 1]\) feature vector as follows

\[
f_{\lambda^S} = [\lambda_1^S, \lambda_2^S, \ldots, \lambda_M^S]^T.
\]

(6)

The final data feature is the \([2M^2 + M] \times 1\) vector given by

\[
f = \begin{bmatrix} f_R \\ f_{\lambda^S} \end{bmatrix}.
\]

(7)

Whereas, the unsmoothed covariance matrix estimate \(R_{XX}\) and its eigenvalues \(\lambda\) were also tested as network inputs. Figure 1 shows the data pre-processing stages to produce the four input data sources. In this figure, the “Input Select and Formatting” block selects combinations of the input data sources and converts these to a single vector as discussed earlier in this section. Herein, six combinations were tested as network inputs. All four inputs were tested independently as well as each covariance matrix paired with its corresponding eigenvalues.

3.3 | Radar parameters

The data set utilised herein simulates a radar consisting of a notional uniform linear array operating at \(f = 5.0\) GHz with element spacing \(\Delta x = \lambda/2\). There are \(M = 11\) receivers in the array, and this value was chosen to present a small array (in order to challenge the algorithms). The number of pulses per coherent processing interval is \(p = 10\), which was also chosen to be a smaller number of pulses in order to evaluate the proposed method with a smaller number of pulses. The sub-array size used in spatial smoothing is \(S = 6\).

3.4 | Proposed DL network

A DL method is proposed utilising a small number of neurons that provides robust results. In the case of a receiver with 11 channels utilising subarrays of six elements each for spatial smoothing, the network will have 78 input features. The proposed network has nine fully connected (FC) layers, a parametric rectified linear unit (PReLU) and a batch normalisation layer.

The PReLU [62] is a non-linear operator that allows both positive and negative values to pass through the layer. Performance was found to increase over a similar network with rectified linear unit (ReLU) activations, which do not

![FIGURE 1] Block diagram of data pre-processing stages
allow negative inputs to pass through. The PRelu is defined as

\[
f(x) = \begin{cases} 
  x & x \geq 0 \\
  -\alpha x & x < 0
\end{cases}
\]  

where \(\alpha\) is a network-learnt parameter.

To mitigate overfitting (i.e. learning the training data at high accuracy but not being able to generalise well to novel data), a dropout layer is used. After the last FC layer, a softmax and classifier layer provide the final outputs. During training, the dropout layer randomly zeros 50% of the outputs connections. During testing, all outputs are available. The FC layers have a connection to each output of the previous layer. They also have an activation function and compute their output as the dot product of the input vector with the internal weight vector plus the bias term. Then the linear or non-linear activation function is applied to provide the final FC output. The batch normalisation layers learn input distribution mean and variance and normalise the output by making it tend to zero mean and unity variance. Batch normalisation layers are utilised to allow network sizes to be large [63]. Finally, the softmax layer in conjunction with the classification layer learns a distribution to estimate the number of sources [64].

Herein, the network weights are initialised randomly from Gaussian distribution with zero mean and variance 0.01, and the biases are initialised as zero, and the network weights are updated using stochastic gradient descent with momentum [64]. The DL architecture is described in Table 2. The sizes of the FC layers are integer multiples of the input vector size.

### 3.5 Proposed algorithm description

The proposed algorithm, shown in Algorithm 1, uses the complex \([M \times P]\) receiver data matrix and estimates the number of sources present.

#### Algorithm 1 Estimate number of targets

**Input**: \(X\), \([M \times P]\) receiver complex data matrix.  
**Input**: \(S\), spatial smoothing subarray size.  
**Output**: \(\hat{T}\), estimate of number of sources.  

**Extract feature vector:**
1: Compute the spatially smoothed covariance matrix using Equation 2.  
2: Compute eigenvalues using Equation 5.  
3: Compute the covariance feature vector using Equation 4.  
4: Compute the eigenvalue feature using Equation 6.  
5: Create the final feature vector using Equation 7.  
6: Evaluate feature vector with proposed DL network.  
7: return \(\hat{T}\)

### 4 DATA SETS

The network presented herein is tested on simulated data and compared to contemporary methods. Table 3 shows the various train and test cases utilised. In Table 3, the function \(\text{rand}[A, B]\) represents the selection of a random value from a uniform distribution in the range \(A \leq x \leq B\). The SNR values in Table 3, are independent for each signal. Each signal features zero to three sources. If no source is simulated, the signal contains only noise. The angles of

| Case | Training | Testing |
|------|----------|---------|
| \(A\) | 0: 20,000 | 0: 20,000 |
| | 1: 20,000 @ 10dB | 1: 20,000 @ 10dB |
| | 2: 20,000 @ 10dB | 2: 20,000 @ 10dB |
| | 3: 20,000 @ 10dB | 3: 20,000 @ 10dB |
| Total: 80,000 | Total: 80,000 |
| \(B\) | 0: 20,000 | 0: 20,000 |
| | 1: 20,000 @ rand[0,20] dB | 1: 20,000 @ rand[0,20] dB |
| | 2: 20,000 @ rand[0,20] dB | 2: 20,000 @ rand[0,20] dB |
| | 3: 20,000 @ rand[0,20] dB | 3: 20,000 @ rand[0,20] dB |
| Total: 80,000 | Total: 80,000 |
| \(C\) | 0: 20,000 | 0: 20,000 |
| | 1: 20,000 @ rand [13,20] dB | 1: 20,000 @ rand [13,20] dB |
| | 2: 20,000 @ rand [13,20] dB | 2: 20,000 @ rand [13,20] dB |
| | 3: 20,000 @ rand [13,20] dB | 3: 20,000 @ rand [13,20] dB |
| Total: 80,000 | Total: 80,000 |

**TABLE 2 Proposed deep network architecture**

| Layers | Type     | Size |
|--------|----------|------|
| Input  |          | 78   |
| 1–3    | PB       | 234  |
| 4–6    | PB       | 468  |
| 7–9    | PB       | 780  |
| 10     | DR       | 50%  |
| 11–13  | PB       | 4    |
| 14     | SM       | 4    |
| 15     | CL       | 4    |

PB = Processing Block, Fully Connected, Parametric ReLU; Batch Normalisation,  
DR = Dropout, SM = Softmax, CL = Classifier.
sources are selected randomly from a uniform distribution within a limited FOV of $-60^\circ$ to $60^\circ$. A minimum source spacing is enforced in the case of multiple sources as sufficiently close sources are impossible to distinguish. This minimal spacing is set to $\Delta \phi = 0.5^\circ$ for all cases. With these parameters, coherent signals are simulated and added together with i.i.d. complex WGN. To simulate coherent sources first independent signals are generated and Cholesky decomposition of the covariance matrix with desired coherency level $\sigma$ is used to generate coherent signals as detailed in [65, 66].

Test case A has all targets at 10 dB SNR. This test case might be for a case of tracking up to three large vehicles. Test case B assumes all targets have random SNRs from 0 to 20 dB, which is a much harder case. Test case C is similar to test case B, but the targets are restricted to the range 13–20 dB SNR. This third test case assumes that we need 13 dB SNR to reliably detect a target. In all cases, the SNR in dB is calculated as $10\log_{10}$ of the ratio of signal power to the noise power at the input channel of the receiver array.

## 5 | RESULTS AND DISCUSSION

Herein, as discussed above, synthesised radar returns from the same range bin are utilised in this study. Returns from different range bins can be processed separately (and similarly for range/Doppler processing). To compare results, the proposed method is compared to AIC, MDL, and EEF using spatial smoothing of the covariance matrix. The proposed method results are shown as confusion matrices and also summarised in terms of the overall accuracy, the percentage of correct entries, underestimates, and overestimates. Also, the overall accuracies for the proposed method, AIC, MDL, and EEF are compared.

### 5.1 | Training parameters

Table 4 shows the training parameters used for the networks. In all cases, stochastic gradient descends with momentum [64]. The network is updated relative to the classification cross entropy loss assuming multiple mutually exclusive classes.

### 5.2 | Test case analysis

The authors note that the training data is independent of the testing data and the training data is only used to train the DL network. For all testing, the test data is utilised (the AIC, MDL, and EEF comparison methods do not require training).

A breakdown of results using confusion matrices is provided in Tables 5, 7, and 9. These tables can be interpreted as follows. The top row indicates the correct number of sources in a signal, whereas the first column indicates the network output. Therefore, the diagonal terms correspond to correct estimates. Additionally, the upper triangle corresponds to underestimates and the lower triangle corresponds to overestimates.

Table 5 shows the confusion matrix and Table 6 shows the overall results for case A, respectively. These tables show the network demonstrated perfect performance for case A. Tables 7 and 8 show results for test case B. This case is much more difficult than case A, as the lower bound for SNR is zero and there is a 20 dB SNR range of the signals. Correspondingly, the network demonstrates a performance drop of approximately 5%. Tables 9 and 10 show the results for test case C. These results achieve near-perfect classification accuracy due to both the higher SNR sources and the reduced range of SNR values.

For all cases, the training and testing data results are very similar, indicating the network is not overtraining to the training data sets.

Table 11 shows the proposed DL method compared to the AIC, MDL, and EEF methods. In all cases the proposed method’s performance is over 6% higher than the compared methods. Additionally, in all of the test cases, AIC, MDL, and EEF have very similar performance.

Two questions about the network performance are “How will the network perform with only the covariance matrix or eigenvalues as inputs?” and “How does the full covariance matrix compare to the smoothed covariance matrix?” In order to address these questions, the network was modified for the
spatially smoothed: covariance matrix plus its eigenvalues. The obtained from the input being the concatenation of the when comparing the results using the covariance matrix smoothing. All results featuring the covariance matrix feature when only provided with eigenvalues regardless of spatial smoothing. Theses observations indicate that what the radar is expected to observe. When applying this method to actual radar systems, the network should be trained on a range of SNR values larger that what the radar is expected to observe.

The effect of the number of pulses included in each signal was tested by generating data sets with the same parameters as case B and varying the number of pulses. These data sets correspond to pulse counts of 1–10, 20, and 30. The network was retrained for each of these data sets and the training and testing accuracy is reported in Figure 3. As this plot is constructed from single training instances and the small scale of accuracy variations, the plot is not monotonically increasing but a general trend of increased pulse counts corresponding to increased accuracy is observed. For pulse counts over 10 only a 1% improvement is observed despite the increased complexity of computing the signal covariance matrix. When decreasing the number of pulses, the network is able to maintain adequate performance even with a small number of pulses.

Timing experiments were performed to indicate the run time of the network compared to the AIC, MDL, and EEF methods. All experiments were performed using MATLAB R2018B. The PC used for experiments runs Windows Server 2012 with an Intel Xeon E5-2670 CPU, 128 GB of RAM, and a Nvidia GTX 970 GPU.

All methods require computation of the smoothed covariance matrix eigenvalues, therefore all methods feature the same pre-processing. This pre-processing time was estimated by computing the smoothed covariance matrix and its eigenvalues for the testing data set in case B This process took 13.7 s, which corresponds to an average of 171.4 μs per signal. The network performance was evaluated in the same manner. The trained network took 3.4 s to estimate the number of sources for all signals, which corresponds to an average of 42.8 μs per signal. The average times for the AIC, MDL, and EEF using the same methodology were 54.5 μs, 55.7 μs, and 100.8 μs respectively. A breakdown of the number of mathematical operations for a similar network is presented in [1].

| TABLE 8 | Proposed method case B overall results |
|----------|----------------------------------|
| Case B   | Train   | Test    |
| Overall accuracy (%) | 95.218 | 94.801 |
| Underestimated (%)   | 3.886  | 4.118  |
| Overestimated (%)    | 0.896  | 1.081  |

| TABLE 9 | Proposed method case C test confusion matrix |
|---------|----------------------------------|
|         | 0   | 1   | 2   | 3   |
| 0       | 20,000 | 0   | 0   | 0   |
| 1       | 0    | 20,000 | 58  | 0   |
| 2       | 0    | 0    | 19,834 | 339 |
| 3       | 0    | 0    | 108  | 19,661 |

| TABLE 10 | Proposed method case C overall results |
|----------|----------------------------------|
| Case 3   | Train   | Test    |
| Overall accuracy (%) | 99.453 | 99.369 |
| Underestimated (%)   | 0.436  | 0.496  |
| Overestimated (%)    | 0.111  | 0.135  |

| TABLE 11 | Overall test results shown as accuracy in percent |
|----------|----------------------------------|
| Proposed | AIC      | MDL      | EEF      |
| A        | 100.00  | 89.67    | 89.48    | 89.47    |
| B        | 94.80   | 88.22    | 87.85    | 87.88    |
| C        | 99.37   | 91.92    | 91.73    | 91.73    |

Note: AIC, MDL, and EEF utilise spatial smoothing. The best results are in bold.

following cases (1) the full (e.g. non-spatially smoothed): covariance matrix only (input vector size of 242), (2) full covariance matrix eigenvalues only (input size 11), (3) concatenation of the full covariance matrix and its eigenvalues (input size 253), (4) smoothed covariance matrix only (input size 72), and (5) the smoothed covariance eigenvalues only (input size 6). Table 12 shows the results for the DL network with these different inputs. From the table, the best results are obtained from the input being the concatenation of the spatially smoothed covariance matrix plus its eigenvalues. The combined smoothed eigenvalue and covariance matrix demonstrate over 5% improvement over all cases not using spatial smoothing. The network shows significant degradation when only provided with eigenvalues regardless of spatial smoothing. All results featuring the covariance matrix feature acceptable results. Interestingly, the addition of the eigenvalues to the covariance matrix degrades the network’s performance when comparing the results using the covariance matrix without spatial smoothing.

To test the performance of the network relative to SNR, the network trained on case B was used to evaluate multiple single SNR data sets. Each single SNR data set contained 1000 signals for each possible number of sources 0 through 3 for a total of 4000, with all sources set at a single SNR value. Thirty-one single SNR data sets were created for each integer SNR value from −5 to 25 dB. The accuracy as well as the number of over- and under-estimates for each case are plotted against SNR in Figure 2. This plot shows steep declines in performance outside of the training data set SNR range of 0–20 dB for both low and high SNR regimes. Additionally, a slight decline in accuracy can be observed at both ends of the trained SNR range, specifically 0–5 dB and 15–20 dB. Only a gradually decline of 5% accuracy is observed as the SNR is reduced from 15 to 5 dB. For low SNR values all errors are underestimates as expected. For high SNR targets a trend of overestimation is observed up until 20 dB when underestimates begin to occur. This behaviour is likely caused by the limited maximum output value of the network which prevents 3 source cases from ever being overestimated. These observations indicate that when applying this method to actual radar systems, the network should be trained on a range of SNR values larger that what the radar is expected to observe.
| Input (Dim) | E (11) | C (242) | C&E (253) | SE (6) | SC (72) | SC&SE (78) |
|-------------|--------|---------|-----------|--------|---------|------------|
| OA %        | 63.104 | 88.710  | 84.790    | 63.004 | 89.526  | **94.801** |
| UE %        | 14.119 | 8.249   | 10.397    | 11.865 | 7.746   | **4.118**  |
| OE %        | 22.777 | 3.041   | 4.813     | 25.131 | 2.728   | **1.081**  |

Note: Best results in bold.

Abbreviations: C,SC = cov., smoothed cov.; E,SE = eigenvalues cov., and smoothed cov.; C&E,SC&SE = cov. and eigs., smoothed cov. and eigs.; OA = Overall Accuracy; UE = Underestimated; OE = Overestimated.

![Performance vs SNR](image1.png)

**FIGURE 2** Accuracy of the proposed method as a function of signal-to-noise ratio

![Performance vs Number of Pulses](image2.png)

**FIGURE 3** Accuracy of the proposed method as a function of number of pulses

### 6 | CONCLUSIONS

Standard solutions for estimating the number of sources, such as AIC and MDL, all fail in cases where the signals present are coherent. However, spatial smoothing is an effective method to address coherent signals.

The proposed DL system, which fuses the spatially smoothed covariance matrix and eigenvalues, was found to accurately estimate the number of sources, even when the number of receiver channels is small and the number of pulses is also modest. The proposed method outperformed two state-of-the-art methods, namely AIC and MDL, when they also used spatial smoothing for inputs. This is an important contribution, because signal-subspace methods such as MUSIC and MLE require a priori estimates of the number of sources. Also, the proposed method does not require matrix inversion or adding diagonal loading to make the covariance matrix better conditioned, and thus artificially inflating the noise floor.

Using the spatially smoothed covariance matrix and eigenvalues provides the best results, and the DL network is smaller than the DL network required when the non-smoothed covariance matrix and eigenvalues are used. The downside of any method using spatial smoothing is that there are fewer overall elements, and thus the number of sources that are detectable is smaller than if the full covariance matrix is used. For realistic subarray sizes, this may not be a large issue in practice. Also, spatial smoothing allows a fewer number of receiver elements to be processed simultaneously, which limits the array resolution, as pointed out by [20].

There could be potential electronic countermeasure (ECM) applications, such as detecting how many low power radars are operating in an area. Classification of jamming environments also requires estimation of the number of sources to make inference on the number of jamming signals affecting each range bin within the radar range swath [67]. There is also potential for non-radar applications, such as modifying the method to not only estimate the number of sources, but also to estimate the relative SNRs of the different sources. This would require adding additional NN regression modules. This approach could have many applications in wireless communications, such as estimating the number of radios talking simultaneously on a channel.

Future work includes (1) analysing the network in the presence of receiver alignment errors, phase and amplitude mismatches, (2) extending the results to wide-bandwidth signals, (3) investigating denoising techniques (denoise the inputs before using the network), (4) extending the work to also estimate the AoA of the sources, (5) extending the solution from a uniform linear array to a 2D array, (6) utilising a complex-valued NN to process the data (7) applying non-Swerling (e.g., log-normal) and wideband data, and (8) continue optimisation of network parameters and configuration.

---

**ORCID**

John E. Ball 🏗️ https://orcid.org/0000-0002-6774-4851

**REFERENCES**

1. Rogers, J., Ball, J.E., Gurbuz, A.C.: Estimating the number of sources via deep learning. In: 2019 IEEE Radar Conference (RadarConf), 1–5. IEEE, Boston (2019)

2. Schmidt, R.: Multiple emitter location and signal parameter estimation. IEEE Trans. Antennas Propag. 34(3), 276–280 (1986)
Advances in Multimedia, Software Engineering and Computing, Vol. 1, pp. 389–394. Springer (2011)

51. Le, Z.: Research on direction of arrival estimation algorithm in smart antenna. South China University of Technology, Guangzhou (2010)

52. Agatonovic, M., et al.: Efficient neural network approach for 2d DOA estimation based on antenna array measurements. Prog. Electromag. Res. 137, 741–758 (2013)

53. Du, K.L., et al.: Neural methods for antenna array signal processing: a review. Signal Process. 82(4), 547–561 (2002)

54. Amari, S.i., Cichocki, A.: Adaptive blind signal processing-neural network approaches. Proc. IEEE. 86(10), 2026–2048 (1998)

55. Tan, Y., Wang, J., Zurada, J.M.: Nonlinear blind source separation using a radial basis function network. IEEE Trans. Neural Network. 12(1), 124–134 (2001)

56. Solazzi, M., Piazza, F., Uncini, A.: Nonlinear blind source separation by spline neural networks. In: 2001 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2001, 5, 2781–2784.IEEE (2001)

57. Hirose, A.: Complex-valued neural networks: theories and applications, vol. 5. World Scientific, River Edge (2003)

58. Hirose, A.: Complex-valued neural networks: Advances and applications, vol. 18. John Wiley & Sons, River Edge (2013)

59. Grais, E.M., Sen, M.U., Erdogan, H.: Deep neural networks for single channel source separation. In: 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 3734–3738. IEEE, Florence (2014)

60. Vesperini, F., et al.: Localising speakers in multiple rooms by using deep neural networks. Comput. Speech Lang (2017)

61. Axler, S.J.: Linear algebra done right, vol. 2. Springer (1997)

62. He, K., et al.: Delving deep into rectifiers: surpassing human-level performance on imagenet classification. In: Proceedings of the IEEE International Conference on Computer Vision, 20126–1034. Santiago (2015)

63. Ioffe, S., Szegedy, C.: Batch normalisation: accelerating deep network training by reducing internal covariate shift Batch normalisation: accelerating deep network training by reducing internal covariate shift. In: International conference on machine learning (pp. 448-456). PMLR, Lille, France (2015)

64. Bishop, C.M.: Pattern recognition and machine learning (information science and statistics). Springer–Verlag (2006)

65. Gan, L., Luo, X.: Direction-of-arrival estimation for uncorrelated and coherent signals in the presence of multipath propagation. IET Microw. Antenn. Propag. 7(9), 746–753 (2013)

66. Smallwood, D.O.: Matrix methods for estimating the coherence functions from estimates of the cross-spectral density matrix. Shock Vib. 3(4), 237–246 (1996)

67. Carotenuto, V., De Maio, A., Iommelli, S.: Clustering for jamming environment classification. In: 2020 IEEE Radar Conference (Radar-Conf20), 1–6. IEEE (2020)

How to cite this article: Rogers J, Ball JE, Gurbuz AC. Robust estimation of the number of coherent radar signal sources using deep learning. IET Radar Sonar Navig. 2021;15:431–440. https://doi.org/10.1049/rsn2.12047