SNR dependent drone classification using convolutional neural networks

DALE, Holly; Baker, Chris; Antoniou, Michail; Jahangir, Mohammed; Atkinson, George; Harman, Stephen

DOI:
10.1049/rsn2.12161

Received: 18 May 2021 | Revised: 17 June 2021 | Accepted: 12 July 2021

DOI: 10.1049/rsn2.12161

IET Radar, Sonar & Navigation

ORIGINAL RESEARCH PAPER

SNR-dependent drone classification using convolutional neural networks

Holly Dale[1] | Chris Baker[1] | Michail Antoniou[1] | Mohammed Jahangir[1] | George Atkinson[1] | Stephen Harman[2]

[1]Microwave Integrated Systems Laboratory, University of Birmingham, Birmingham, UK

[2]Aveillant, Cambridge, UK

Correspondence

Holly Dale, Microwave Integrated Systems Laboratory, University of Birmingham, Edgbaston, Birmingham, B15 2TT, UK.
Email: had449@student.bham.ac.uk

Funding information

UK National Quantum Technology Hub in Sensing and Timing, Grant/Award Number: EP/T001046/1; Engineering and Physical Sciences Research Council

Abstract

Radar sensing offers a method of achieving 24-h all-weather drone surveillance, but in order to be maximally effective, systems need to be able to discriminate between birds and drones. This work examines drone-bird classification performance as a function of signal to noise ratio (SNR). Classification at low SNR values is necessary in order to classify drones with a small radar cross-section (RCS), as well as to facilitate reliable classification at longer ranges. To investigate the relationship between classification performance and SNR, Gaussian noise is added to an experimentally obtained dataset of radar spectrograms. Classification is performed by convolutional neural networks (CNNs). It is shown that for the data available classification accuracy drops with falling SNR, as might be expected for any given CNN. The degree to which performance degrades with reduced SNR is presented. It is further shown that simpler network architectures are more robust to noise. Finally, it is demonstrated that data augmentation can be used as a means of enhancing classification accuracy at lower SNR values. Bayesian optimisation is used to find the optimal augmentation hyperparameters and overall, classification accuracies of 92% are achieved at low SNR.

1 | INTRODUCTION

Manned and unmanned airspace is undergoing a transformation. Recent years have seen a proliferation in the number of drones, and by 2030 air traffic is estimated to quadruple with a doubling of the total number of manned aircraft with matching numbers of unmanned air vehicles. This is a fundamental shift in the use of airspace and in particular, low-level airspace.

The increasing number of hobbyist and commercial drones is of major public interest. There are numerous benefits to the use of drones, including in agriculture, photography, and emergency services to support search and rescue missions, as well as for leisure purposes [1]. In the United Kingdom alone, drones are estimated to lead to a £42 billion uplift in GDP by 2030 across many industries including media, construction, and transport [2].

The use of drones in an already congested airspace poses risks to manned aircraft, with an increasing number of drone-aircraft near misses—in December 2018, Gatwick Airport (UK) was closed for a number of days following reports of a drone sighting [3]. In addition, drones can be used for nefarious purposes such as illegal surveillance, contraband delivery, and terrorism. It is therefore imperative to monitor the use of drones, in order to ensure the safety of both civilians and corporations. Radar is the only method capable of offering 24-h, all-weather non-cooperative surveillance at long distances and is thus a promising solution.

Drones are a challenge to detect due to their relatively small radar cross-section (RCS), slow velocities, and low flight altitude. Staring radar is able to provide the high sensitivity that is required to slow moving targets, and also has the advantage of providing data continuously, where the update rate is effectively set by the dwell time and limited by the processing overhead. As a consequence of the high sensitivity, other slow-moving targets with a low RCS are also detected. There is a similarity between the RCS of drones and birds [4–6], meaning birds often present as 'confuser targets' from a drone surveillance perspective. This can lead to an unacceptably high rate of false reports, reducing the effectiveness of the surveillance method.

There is an inherent need for a high performing classification algorithm, such that only detections that truly belong to drones are reported. In this study, we explore the limitations of such a drone discrimination algorithm by investigating performance on targets as a function of signal to noise ratio (SNR).

In recent years, there have been a number of research papers published on drone classification. In [7], kinematic differences in drone and bird trajectories (including velocity, acceleration, height, and jolt) were used to filter out bird returns and improve the probability of drone detection. In [8], statistical trajectory features were used to classify drone and bird tracks with a high degree of accuracy, with the dominant classification features being velocity based.

The rotation of drone propellers and the action of flapping of bird's wings modulates the radar signal, producing different and characteristic micro-Doppler signatures [9, 10]. Returns from the propellers of drones are a useful feature for classification—from these returns, one can, in principle, estimate the number of blades per rotor, the length of the blades, and the rate of rotation [11, 12]. In [13], a multi-layer perceptron was used to classify drone models by estimating the number of propellers, blades, and blade length and rotation frequency. In [14], the eigenvalues and eigenvectors of the micro-Doppler signature were used to classify 10 drone models and an artificial bird.

In [14], the integration time is significantly longer than the period of rotation of a drone propeller, therefore, Doppler signatures consist of spectral lines rather than blade flashes [15]. The spectral lines are shaped by the radar parameters (frequency, dwell time, etc.) and by the target parameters (rotor length, rotational frequency, and RCS)—this is explored in [16]. Measuring the fundamental frequency of the spectral lines gives an estimate of the rotation rate [17], whilst spectral lines have been used in [18, 19] to distinguish between loaded and unloaded drones. However, for bird and drone discrimination only the drones exhibit strong spectral lines, facilitating classification [20]. In some reported publications, both kinematic and micro-Doppler features have been used together to enhance the classification performance [21, 22]. In [23], the presence of symmetrical components about the body of the target was shown to be a key classification feature for distinguishing drones from birds.

A number of the techniques discussed here utilise fingerprint features to achieve high classification performance, relying on many components of the micro-Doppler signature —some of which have very low signal levels. As the signal power of the spectral lines reduces (whether due to a low propeller RCS or increased range, or due to self-shadowing by the platform fuselage), the classification problem will become more challenging. It is therefore expected that classification performance will deteriorate as the drone micro-Doppler returns become masked by the noise floor. This will limit the maximum range at which drones can reliably be distinguished from birds, consequently limiting the amount of time to respond to threats. Note that the maximum ranges for detection and for classification are likely to be different, therefore, will need to be a factor considered in radar design.

Convolutional neural networks (CNNs) have been demonstrated to show much promise with their ability to derive combinations of features that are not necessarily intuitive and might otherwise evade more direct methods [24]. CNNs have been used both to distinguish between drones and birds [24, 25] and drones of different models [26]. In [24–26] transfer learning was utilised in order to classify spectrograms taking advantage of the low-level features learnt from optical images. In [25, 27, 33], it is demonstrated that the use of transfer learning speeds up the learning process and leads to improved classification performance when the amount of training data is limited.

The potential performance in using CNNs for drone classification has been demonstrated in previous work. At high SNR and short range, the performance is impressive, but a natural extension is to consider performance degradation with decreasing SNR [28, 29]. In order to ensure that the radar operator has sufficient time to identify threats and respond appropriately, it is vital to maximise the distance from the radar at which inbound drones can be classified reliably. For example, consider a drone moving at 20 m/s radially towards the radar operator. Classifying the drone at 1 km gives a radar operator less than 50 s to respond as appropriate.

The goal of this study is to explore the performance of a CNN-based classifier as a function of SNR, and to explore avenues of boosting performance where necessary. Gaussian noise is added to the data to decrease the SNR, and the resulting impact upon classification performance is evaluated. Following this, methods augmenting the training data are considered, and shown to enhance classifier robustness, especially under lower SNR conditions.

The remainder of the study is organised as follows: Section 2 gives an overview of the staring radar used for data collection in this work. Section 2.2 outlines the processing steps by which spectrograms are produced. CNNs are briefly described in Section 3. Section 4 investigates the relationship between SNR and the accuracy of a CNN trained to distinguish between bird and drone radar spectrograms. Robustness is then investigated as a function of network architecture. In Section 5, the results of augmenting the training data by adding Gaussian noise is presented. Bayesian optimisation is used to find the optimal hyperparameters for data augmentation in training, in order to extend the classification range. Finally, conclusions are given in Section 6.

2 | DATA COLLECTION OVERVIEW

2.1 | Staring radar

Data were gathered using an Aveillant Gamekeeper 16U drone discrimination radar (Figure 1) [30]. The radar stares in all look directions simultaneously, therefore, all targets within the search volume are continuously monitored (Figure 2). The staring aspect allows long integration time, resulting in high Doppler resolution. The transmitter uses a broad beam covering the whole of the surveillance volume and transmits a pulsed L-band waveform. The waveform is digitised on receive by each receiver element arranged in a 4 × 16 array in azimuth and elevation, respectively. For each pulse, samples from all receiver channels are processed to form multiple receive beams.

In order to obtain the Doppler spectrogram for a given target, the three-dimensional (3D) range and angle resolution cell containing the target is selected and concatenated to form a time series. For the purpose of the spectrum data extraction, target location information is taken from the tracker output of the Gamekeeper 16U system. As detailed in [31], ground truth data is used to label the output tracks.

For this work, a Hamming window is applied to the time series data before being coherently processed in a fast Fourier transform with a window length of 2048, to obtain N Doppler spectra corresponding to time steps of ~0.25 s. Forty of these spectra are then combined to form a single spectrogram.

FIGURE 1 Aveillant Gamekeeper 16U L-Band multi-beam staring radar

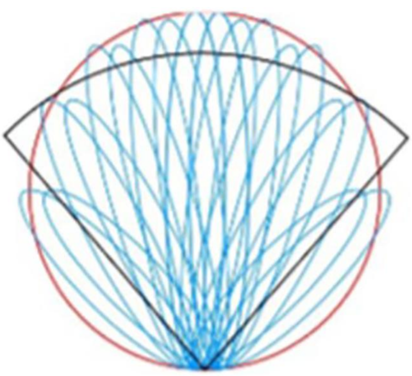

FIGURE 2 A diagram showing the transmit (red) and receive (blue) beams of a staring radar

2.2 | Spectrogram data

The long-time duration of the spectrogram captures the time evolution of the Doppler returns. The overall dataset consists of 966 spectrograms, equally balanced between birds (of various species) and DJI Inspire 1 drones, where the drones carried out different manoeuvres corresponding to pre-determined scenarios. 60% of the data was used for training and 30% was used for testing, whilst the remaining 10% was used for validation during training. Table 1 contains the number of spectrograms in the training, test and validation subsets. The main radar and signal processing parameters are given in Table 2.

Figures 3 and 4 show example spectrograms for a DJI Inspire 1 drone and a bird (species unknown), respectively. The drone body is evident as the time series of high amplitude

TABLE 1 The number of spectrograms used for training, validation and testing

	Training data	Validation data	Test data
Number of spectrograms	580	97	289
Time (s)	6335	1059	3157

Note: The spectrograms are equally balanced between the two classes.

TABLE 2 The operating parameters of the Aveillant Gamekeeper 16U radar used for data collection, and the processing parameters for spectrogram generation

Parameter	Value
Frequency	L band
Bandwidth	2 MHz
Transmit power	2 kW
Receiver channels	4 × 16
Azimuth coverage	90°
Elevation coverage	30°
Pulse repetition frequency	~7.5 kHz
Coherent processing interval	~0.25 s
Doppler resolution	~4 Hz
Spectrogram length	~11 s

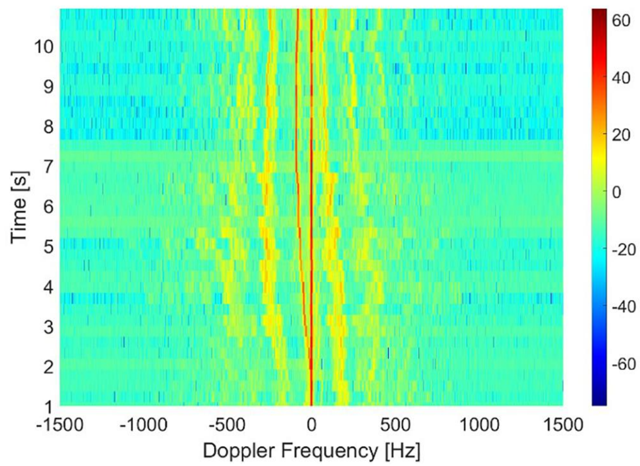

FIGURE 3 Example spectrogram showing the time series of Doppler profiles for a DJI Inspire 1 drone

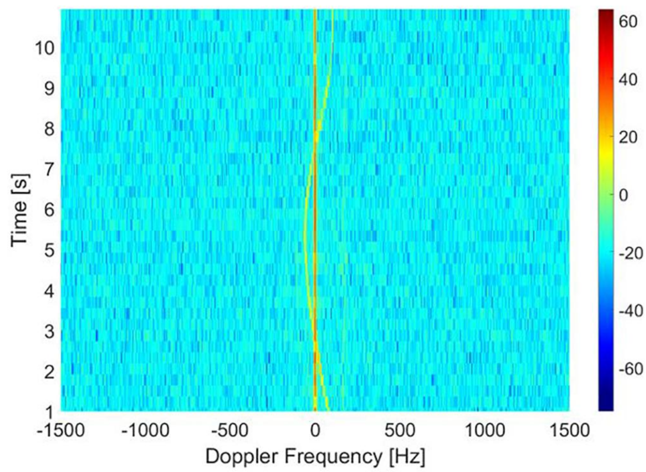

FIGURE 4 Example spectrogram showing the times series of Doppler profiles for a bird

echoes close to zero Doppler frequency in Figure 3. Micro-Doppler returns are clearly visible on either side of the drone body return. These modulations are due to the rapidly rotating propellers. The SNR of the body was approximately 45 dB. On average, the maximum propeller returns sit 15 dB below the return from the body.

In Figure 4 the bird echo is the strongest observable return. The time series shows a similar form for the body return as observed for the drone and the absence of any sidebands is obvious. The Doppler frequency oscillates from positive to negative as the bird flies in an approximately circular trajectory. The SNR of this particular bird is approximately 35 dB.

These example spectrograms are typical of those generally observed and the differences between the two suggest that high classification performance ought to be possible [24]. The most obvious difference between Figures 3 and 4 is the presence of spectral lines in the drone spectra.

It is this difference that provides an obvious basis for discrimination between birds and drones and we may expect high classification results when these drone sidebands are visible. However, as SNR reduces (whether due to increasing distance from the radar or reduced drone RCS), the Doppler sidebands will also have reduced (sideband) SNR and will, ultimately, become undetectable, changing the nature of the classification problem and making it much more challenging. This poses the question as to what classification accuracy can be maintained as the SNR decreases.

In order to investigate the classifier's robustness to noise, the SNR has been reduced artificially by adding Gaussian distributed noise to all available spectrograms. The SNR of the original dataset has been decreased in increments of 3 dB, up to a maximum reduction of 24 dB, resulting in nine datasets of varying SNR. Note that adding noise to the data is not equivalent to increasing the target range, as some effects (such as the degradation in beam resolution that would increase the likelihood of the spectrogram containing multiple targets, and increase the amount of clutter present) have not been considered.

Figure 5a–c show a series of example drone spectrograms where white Gaussian noise was added to reduce the SNR by 12 and 24 dB compared to the original, corresponding to a doubling and quadrupling of range, respectively. In addition, a high pass filter was applied to remove the stationary clutter from the spectrograms, to prevent overfitting of the classifier during training. Note that, as expected, decreasing the SNR reduces the visibility of the drone propeller returns. It is, therefore, expected that classification performance will also degrade with reducing SNR or by consequence with increasing range.

3 | CONVOLUTIONAL NEURAL NETWORKS

3.1 | Feature extraction

CNNs map image inputs on to a set of output variables. In the first convolutional layer, learnt kernels are convolved with the input; the sliding window is moved across the image as shown in Figure 6, taking the dot product of the filter and the sampled image to build up a 2D activation map. Each filter within a layer is applied to the image, each producing its own activation maps. The activation map for a particular filter is an indicator of whether the feature learnt by the filter is present in the input image.

In subsequent convolutional layers, filters are reapplied to the output, mapping lower order features such as edges or a particular colour (echo strength) onto increasingly complex non-linear features in a hierarchical fashion. CNNs are often used for image classification as spatial information is preserved, the networks are invariant to small variations in shape, and they are capable of learning abstract features directly from the data.

The filters are learnt by updating the network weights and biases to minimise the number of training samples that are misclassified. Errors are backpropagated through the network, and the weights in each layer are adjusted accordingly. This process is then iterated over until the network error converges.

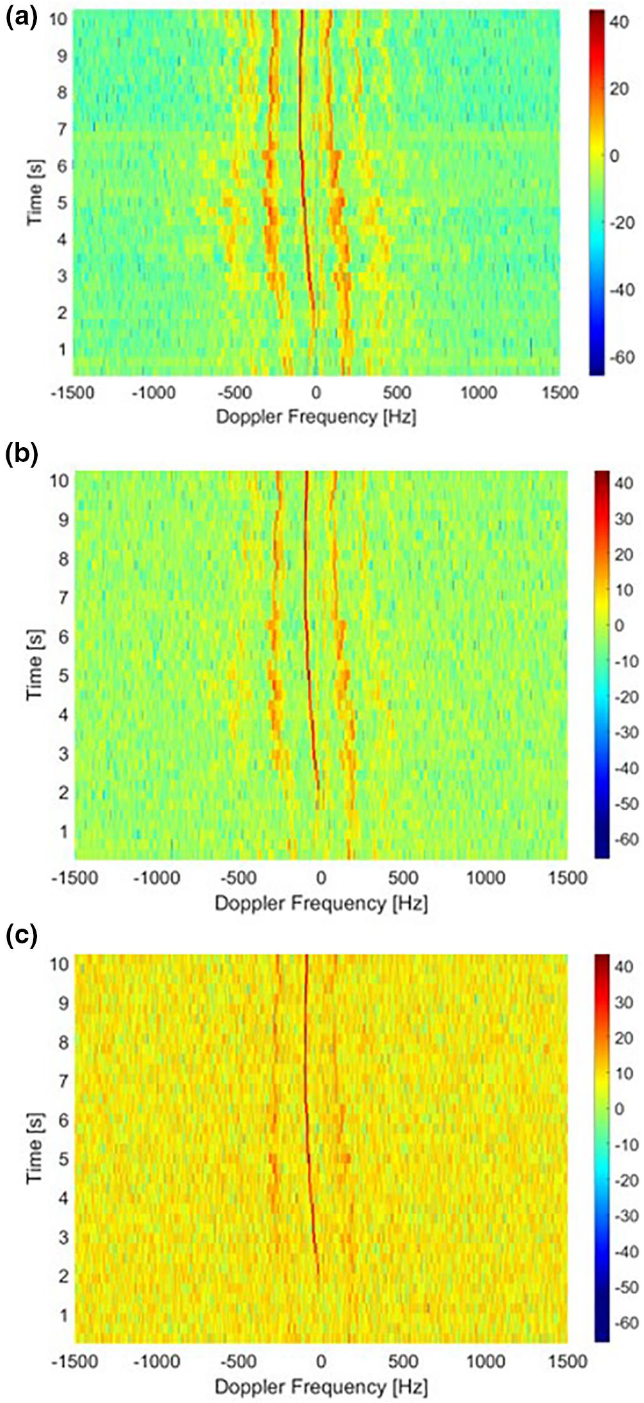

FIGURE 5 An example drone spectrogram where the signal to noise ratio (SNR) has been artificially reduced (a) original, (b) SNR reduced by 12 dB, and (c) SNR reduced by 24 dB

3.2 | Network architecture

AlexNet [32], a classifier 'pre-trained' on 1.2 million ImageNet images, across 1000 categories, was modified to output a spectrogram label. The benefit of transfer learning is that rather than starting from scratch with initial random weights, the network can already pick out low level features such as colour and edges, which are also suitable for the classification of radar spectrograms. Transfer learning using RGB images has been shown to speed up the learning process and lead to improved spectrogram classification performance [33].

The input to AlexNet is a 256 × 256 colour image. The network, as illustrated in Figure 7, consists of five convolutional layers and three fully connected layers. The weights in these layers are trained using stochastic gradient descent with momentum, to minimise the loss function on the training data. The overlapping max pooling layers down sample the feature maps and promote local translation invariance. A dropout rate of 0.5 was used, whereby nodes were randomly dropped during training. As well as this, augmentation of the training data (mirroring in X and Y dimensions) was performed to prevent overfitting of the model.

Although CNNs are a 'black box' algorithm, feeding an image to the network and viewing the corresponding activations can indicate the types of features that the network is extracting. Examining the areas of high activation in the outputs from each channel within a particular layer and comparing these with the original image may allow one to deduce which features the network is learning. Identifying these features can give a deeper understanding of the network, and potentially allow exploitation of this knowledge to enhance classifier performance.

3.3 | Hyperparameter optimisation

Model performance is not only a function of architecture but also a choice of hyperparameters: particularly learn rate, mini-batch size, and number of epochs over which the classifier is trained. Hyperparameter optimisation is used to tune the values of the hyperparameters in order to achieve the highest possible classification performance.

Due to the 'black box' nature of CNNs, the objective function that is to be maximised is unknown. Therefore, the impact of hyperparameters on classification performance must be approximated rather than calculated explicitly. 'Brute force' hyperparameter optimisation methods such as grid search and random search can be time consuming and computationally expensive, particularly when many hyperparameters are to be considered.

In [34], Bayesian optimisation based on Gaussian processes is used to tune several parameters of a CNN trained on the CIFAR-10 dataset—a set of 60,000 images across 10 classes commonly used for training and testing machine learning algorithms. The parameters tuned include the number of training epochs, learning rate, and pooling parameters. It was demonstrated that the Bayesian optimisation approach tuned the parameters faster than other approaches, and the resulting accuracy was demonstrated to beat the then state of the art performance. In [35], a Bayesian optimisation algorithm is outlined, and used to optimise learning rate, decay rate, and mini batch size of CNNs trained on the MNIST and CIFAR-10 databases.

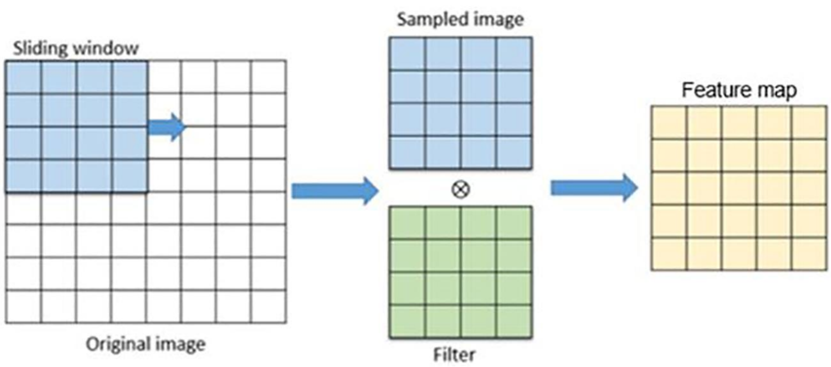

FIGURE 6 An example convolution between the input image (white) and a single filter (green). A sliding window (blue) passes over the image: the dot product of the sampled image and the filter produces a single element in the feature map (yellow)

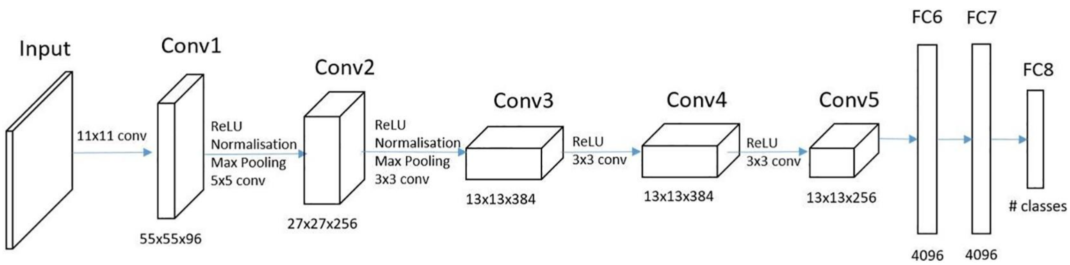

FIGURE 7 AlexNet architecture

The classifier is trained with a set of initial parameters, and the objective function is calculated. The acquisition function is used to determine the next set of hyperparameters to use for training, based on the estimated objective function. Model performance can be treated as a sample from a Gaussian process, and as more observations are made, the posterior distribution is updated. A common choice of acquisition function is the expected improvement [36], as it strikes a compromise between focussing on regions where the objective function is estimated to be at a maximum, and investigating unexplored regions of high uncertainty.

In this study, the fractional split of the training data across a range of SNR bins is expressed as a set of hyperparameters. Bayesian optimisation is used to find the optimal values of these hyperparameters, thus suggesting the optimal amount of noise for data augmentation (for a particular dataset) in order to train a robust classifier.

4 | INVESTIGATION OF CLASSIFIER ROBUSTNESS

4.1 | Testing a classifier on low SNR data

The CNN was trained on a subset of the unmodified (high SNR) data, and performance was evaluated on the unseen subsets of the spectrograms for varying levels of added noise. Figure 8 shows that, as expected, classification accuracy falls as noise is increased. For example, reducing the test data SNR by 12 dB (corresponding to a doubling of range) results in a reduction in classification accuracy from 99.0% to 95.5%, whilst reducing the SNR by 24 dB leads to a classification accuracy of 81.3%. Degrading the SNR beyond this leads to almost half of the drones being misclassified. In this way it is seen that the classification accuracy ultimately degrades as a function of decreasing SNR. Consequently, there comes a point where the SNR is too low, preventing the classifier from being used to reliably discriminate drones from birds.

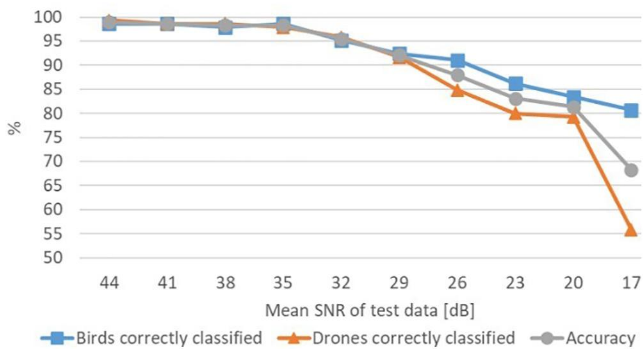

FIGURE 8 Performance of a classifier trained on data of a high signal to noise ratio (SNR), as the SNR of the test data is decreased in increments of 3 dB. Shown is the percentage of birds, drones and total number of targets correctly classified (blue, orange and grey, respectively)

Note the difference in accuracy between the two classes—degrading the SNR of the test data has more of an impact on the classification of drones than of birds. This indicates that the poor performance at low SNR is partly due to the classifier overfitting on the training data, relying on the presence of the modulations from the drone propellers. This can be seen clearly by viewing the activations from the first convolutional layer (Figure 9a–e). The filters that are most strongly activated by the input images are those selecting spectral lines, suggesting that the presence of propeller returns is one of the key features used by the classifier to achieve high classification accuracies. As noise is added to the test image, the channel becomes less activated. This aligns well with what might be expected from a visual inspection of the spectrograms such as those shown in Figure 5.

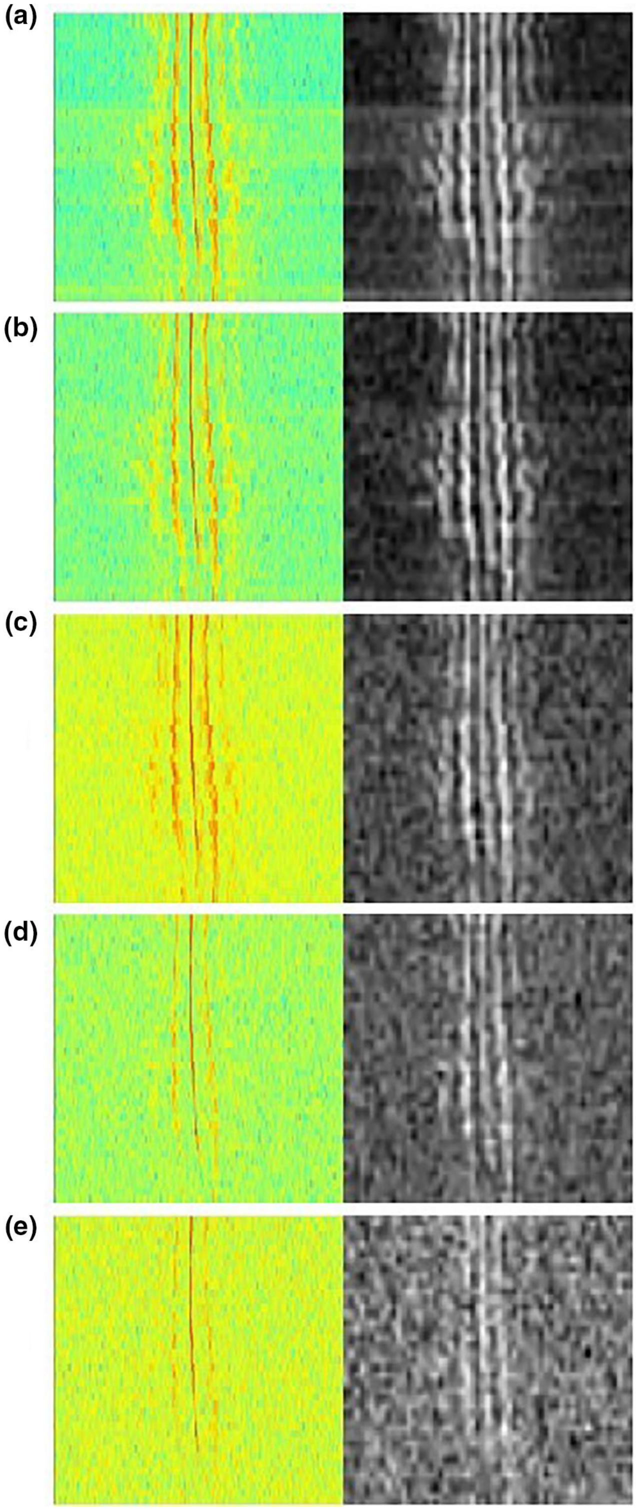

FIGURE 9 Maximum activations in the first convolutional layer when testing a CNN on a drone spectrogram with varying SNR. White indicates areas of high activation, whilst black indicates no activation (a) 46 dB SNR, (b) 40 dB SNR, (c) 34 dB SNR, (d) 26 dB SNR and (e) 20 dB SNR. CNN, convolutional neural network; SNR, signal to noise ratio

4.2 | Robustness as a function of network architecture

A robust classifier is one whose performance remains high when testing on data that differs from the data seen during training. Here, the classifier's robustness to noise is under investigation. The performance of different CNN architectures is compared in order to determine whether there is a preferential network that is more suitable for classification in the radar domain, particularly when test data SNR is low.

Choice of CNN architecture will have different trade-offs between performance, ease of training and computational cost. Recently, seven models were evaluated and compared in the ImageNet Large Scale Visual Recognition Competition (ILSVRC) [37]. Broadly, the networks that have a deeper structure such as Inception-v4 and VGG-16 and 19 had better classification performance at the expense of a higher computational load. In [37], optical data was used to examine drone classification performance. It was reported that shallower networks such as AlexNet or GoogLeNet performed better than their more complex counterparts when the number of classification classes is small.

The CNNs compared are AlexNet, GoogLeNet, SqueezeNet, Inception-v3 and ResNet (ResNet-18 and 50). These networks are in common usage, and were considered to be representative given the various options available.

Figure 10 shows a comparison of the classification accuracy of the six CNNs. All of the CNNs perform well on test data where the SNR is comparable to that of the training data, that is, 44 dB. This performance is more or less maintained up to an SNR of 32 dB and hence a doubling in range. As further noise is added, classification accuracy becomes a function of architecture. Figure 10 shows that AlexNet vastly

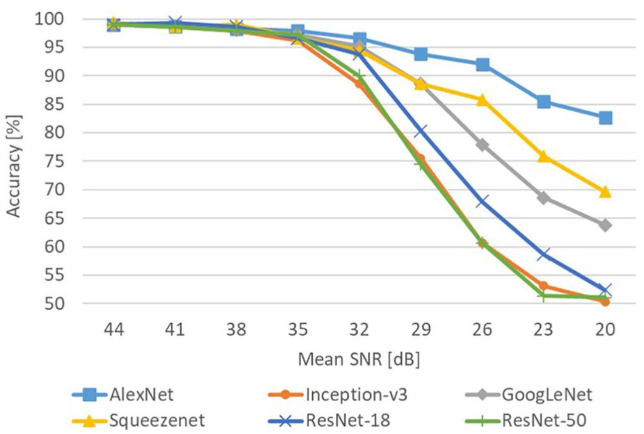

FIGURE 10 Classification accuracy as a function of network architecture

outperforms the other networks, maintaining an accuracy of 80% despite degrading the SNR of the test data by 24 dB. For this reason, the AlexNet architecture was used for the remainder of this work. The two deepest networks—Resnet50 and Inception-v3—perform very poorly, achieving an accuracy of around 50%, which is no better than guessing. The results indicate that the deeper networks are more prone to overfitting on this dataset rather than learning robust features that are characteristic of the targets. Perhaps the depth of the network leads to an unnecessary level of abstraction for the size of dataset and complexity of the problem, preventing generalisation to noisier spectrograms. Of course, the results are limited by the available dataset and it could be that the performance of the shallower networks degrade when a wider range of drone models are considered.

Table 3 shows a summary of the results for each CNN in terms of the number of layers in the network, the number of filters, and the time required for learning. Overall, choice of network will be a judgement made against the requirements of an individual application, but the results indicate that AlexNet is a good choice of classifier for low SNR data, balancing high performance with the lowest training and test times. However, this is in direct contradiction with the results of ILSVRC, but supports the findings of [37] that shallower networks can outperform deeper networks when trained on a small number of classes.

5 | DATA AUGMENTATION FOR LOW SNR CLASSIFICATION

5.1 | Training and testing on data of a similar SNR

In order to investigate whether reliable classification accuracy is possible as the SNR is reduced, a classifier was trained on each of the 10 training datasets. Each subset of test data was then tested on the model that was trained exclusively on data of the same SNR. As shown in Figure 11, as the SNR of the training data was decreased, the fall-off in test accuracy was less severe. The accuracy only drops by 3.4%, after reducing the SNR by 24 dB, where spectrograms are similar to those shown in Figure 9e and do not clearly show sideband returns, whereas previously it had dropped by 19%. This suggests that higher classification accuracies can be achieved at longer ranges, provided that data of a similar SNR is represented in the training data. In other words, the training data is now more representative of both the higher and lower SNR conditions and hence the classification performance is boosted and the range of potential applications is extended.

Classifying spectrograms using CNNs trained on data at similar SNR levels therefore appears to result in lower false alarm rates at lower SNR (that can equate to data at longer ranges) compared to using an algorithm trained solely on high SNR data. In our example, comparing Figures 8 and 11, a target at nearly four times the distance (24 dB less SNR) can be classified with a ∼15% better accuracy. This is an important result in terms of strategies for collecting training data, for best practices to follow when training a classifier, and for constraints on the performance achieved by the CNN.

The result also indicates that performance accuracy could be optimised by having a series of classifiers, all trained on a narrow range of SNRs. The SNR of a target would then be measured, and the spectrogram would be fed to the corresponding classifier. However, this may be an unnecessarily complicated approach. Simply training on lower SNR data could result in the extraction of more robust classification features that allow classification at both high and low SNR.

To investigate this, classifiers were trained on each of the training subsets, in 6 dB increments, and each model was tested on every subset of test data. As shown in Figure 12, adding some noise to the training data facilitates improved classification of noisier test spectrograms, and in some cases, high accuracy on the high SNR data is still maintained. This indicates that a single classifier may be robust across a span of ranges, provided that it has been exposed to a sufficient amount of noise in the training data. Whilst performance may not be as good as in the ideal case shown in Figure 11, it may be deemed acceptable for some applications, and is an improvement over the classifier trained exclusively on high SNR spectrograms (Figure 8). In the case considered here, the highest performing classifier is the one trained on data with an SNR 12 dB lower

TABLE 3 Comparison of the training and test times of six different CNN architectures

Network	Number of layers	Number of learnable parameters [M]	Training time [s]
AlexNet	8	61	2029
Squeezenet	18	1.24	1668
Resnet-18	18	25.6	3821
GoogLeNet	22	7	4136
Inception-v3	48	23.9	16,808
Resnet-50	50	44.6	10,842

Abbreviation: CNN, convolutional neural network.

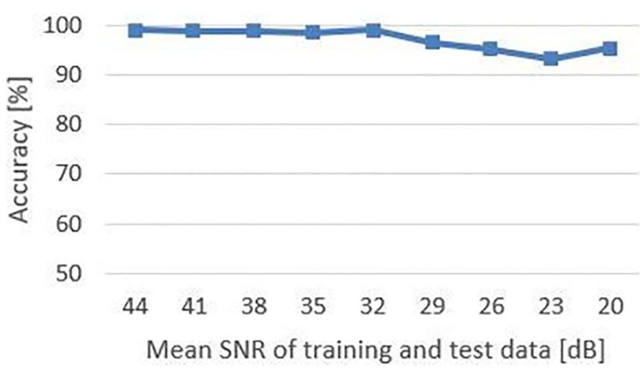

FIGURE 11 Classification performance when training and testing on data of a similar SNR. SNR, signal to noise ratio

than the original SNR. This experiment indicates that it is possible to achieve higher performance without training on data of a similar SNR. There is clearly a limit to the amount of noise to add to the training data, as evidenced by the result of training a classifier on the lowest SNR training dataset, whereby performance on the high SNR test dataset is poor, since the training and test datasets are very different from each other. However, performance on the low SNR dataset is significantly higher than that of the other classifiers considered in Figure 12, suggesting that adding small amounts of noise to the training data may lead to an improvement in classifier robustness. The next section will explore this in more detail, by training a classifier on multiple SNR bins.

5.2 | Data augmentation for a robust classifier

A method of improving classifier robustness to noise is to train a single classifier on data with a range of SNRs. This is similar to a common method of data augmentation often used in machine learning, whereby noise is added to some of the training samples to reduce overfitting. This method introduces additional hyperparameters for training, namely, which values of SNR to train on, and how to split this percentage-wise.

The 10 training subsets were equally sampled and used to train a CNN. Note that each image was only represented in the training data once, at a random noise level. The network was then tested on each of the training subsets. As shown in Figure 13, this method still resulted in a sharp falloff in accuracy as the SNR of the test images was reduced, although the percentage of drones correctly classified remained high throughout. Augmenting the data using this method produces a classifier that is slightly more robust than the original, although performance on high SNR data has been impacted somewhat.

Better performance could be achieved by weighting the training data towards a particular noise value, rather than sampling uniformly. To investigate this, Bayesian optimisation was used to find the optimal percentage split across the training subsets, where the objective function is the performance loss on a set of test data. There is a choice of which test data to use to calculate the objective function—whether to optimise performance across the board, or to optimise solely on low SNR data. Both methods are considered.

First, six hyperparameters were introduced: $\%^0$, $\%^6$, $\%^{12}$, $\%^{18}$, $\%^{21}$, and $\%^{24}$, with the constraint that the sum of these values is equal to 100. These hyperparameters control the amount of training data taken from a particular SNR bin. The Expected Improvement acquisition function was used to choose the next set of hyperparameters after each evaluation. A time constraint of 12 h was imposed on the objective function evaluation, after which the optimal hyperparameters were selected.

The hyperparameters that resulted in optimal performance on the test data are displayed in Table 4. If the parameters are chosen based on performance across the board, then the optimal solution is to reduce the SNR of over half of the training data by 12 dB. Only 2% of the training data should have no noise added to it. Conversely, if the hyperparameters are chosen based on classifier performance of low SNR data (i.e., test data that had the SNR reduced by 12 dB or more) then the optimal solution is slightly different that is, one third of the training data should have the SNR reduced by 18 dB, and 14% of the data should be unchanged. The results can be understood as follows: adding a moderate amount of noise to the majority of training samples improves generalisation to both medium and low SNR data.

The performances of the resulting classifiers are shown in Figures 14 and 15. Both figures show that high performance is retained at high SNRs, and the fall of in performance at lower SNR is now decreased. In both cases, an accuracy above 90%

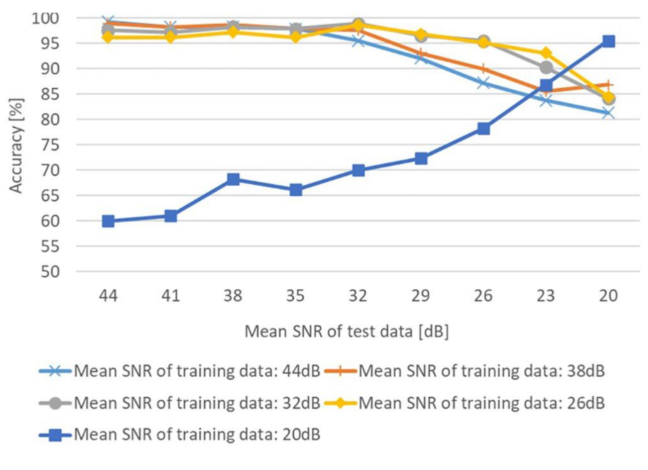

FIGURE 12 Classification accuracy as a function of test SNR, for classifiers trained on a range of SNR bins. SNR, signal to noise ratio

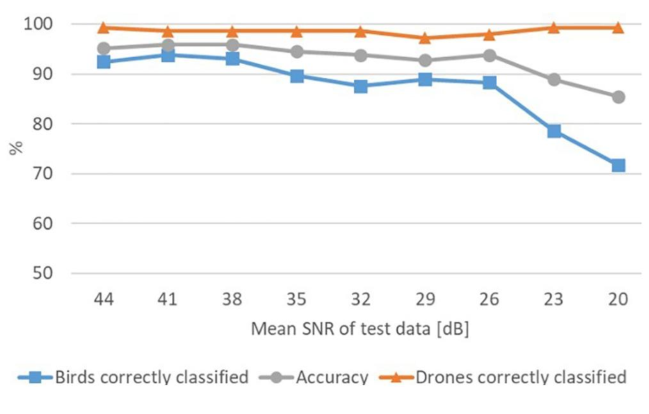

FIGURE 13 Classification accuracy as a function of test SNR, when each SNR bin is equally represented in the training data. SNR, signal to noise ratio

TABLE 4 The optimal percentage split of the training data across various SNR bins

	$\%^0$	$\%^6$	$\%^{12}$	$\%^{18}$	$\%^{21}$	$\%^{24}$
Validated on all data	2	7	57	4	28	1
Validated on low SNR data	14	10	23	33	6	13

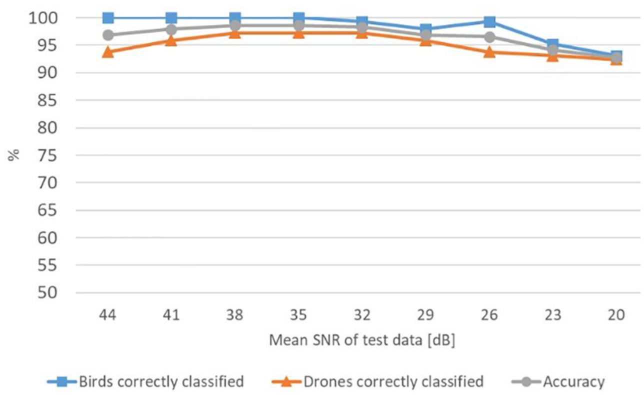

FIGURE 14 Classification accuracy as a function of test signal to noise ratio (SNR), when the hyperparameters for augmentation of the training data have been found using Bayesian optimisation on data across a range of SNRs

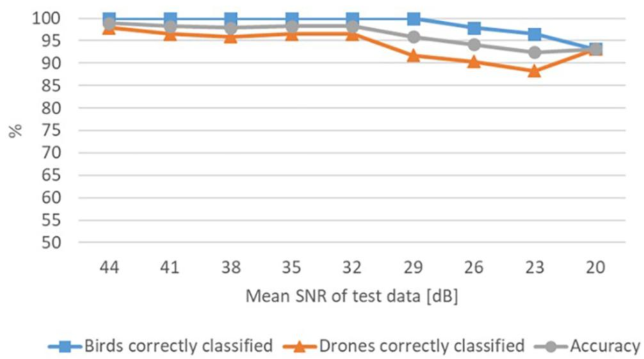

FIGURE 15 Classification accuracy as a function of test signal to noise ratio (SNR), when the hyperparameters for augmentation of the training data have been found using Bayesian optimisation on data across low SNRs (32–20 dB)

is maintained despite reducing the test data SNR by 24 dB. Both the optimisation methods resulted in a more robust classifier than the case where the training data was equally split across the SNR bins, resulting in an improvement in accuracy of over 10% at the most extreme range.

Optimising only on the low SNR data only leads to a slight improvement in performance at lower SNR but slightly poorer performance on intermediate SNR. In particular, note the number of drones misclassified—these are the critical errors, as the consequences of failing to identify a hazardous drone may be much more severe than the occasional nuisance of bird false alarms. Of course, this is very much application dependent. The second method results in drone misclassification rate of 11.7% at its largest, compared with 7.6% for the first method. Therefore, for most applications, the best method is the Bayesian optimisation approach, where the choice of hyperparameters is made by validating on the full spectrum of SNR bins.

Using augmentation and choosing the training hyperparameters using Bayesian optimisation produces a classifier that maintains an accuracy of 95% when degrading the test data SNR by 18 dB. Assuming a drone is moving radially towards the radar at a uniform velocity, classifying at this range would give an operator almost triple the time to respond to the threat.

6 | CONCLUSIONS

In this study the problem of drone-bird discrimination has been considered as a function of SNR in both the data used to train a CNN and the data used for evaluation of classification accuracies. It has been shown that a classifier trained on high SNR data will have degraded performance on lower SNR data, which may be the case for longer range. Methods of restoring classifier performance to improve overall robustness have been examined. This has included investigation of different network architectures, augmentation of the training data, and the use of Bayesian optimisation to find the optimal augmentation parameters.

It was demonstrated that adding noise to the training data results in a more robust classifer, facilitating classification at longer ranges than would otherwise be the case. The optimal solution for the dataset considered here is to train a series of classifiers, each on a particular SNR bin, then to measure the SNR of test data and use the corresponding classifier. However, in practice, this would be difficult to implement. It was shown that training on noisier data could lead to the extraction of features that are robust across the board, leading to an improvement in accuracy at low SNR whilst maintaining high accuracy on high SNR data.

It has also been shown that training on a range of SNR bins leads to a marked improvement in classification accuracy and robustness. The optimal solution was found to be reducing the SNR of 57% of the data by 12 dB. This solution leads to a more robust classifier that maintains an accuracy of 92.4% upon test data with a 24 dB degradation in SNR, reflecting a quadrupling of range.

Based on the results of this work it is recommended to train on a range of SNR values in order to train a robust classifier—either through direct measurement or by deliberately degrading training data acquired at high SNR. Very high test performance can be achieved on test data with an SNR between 45 and 20 dB by training primarily on data with an SNR in the region of 30 dB. Based on this work an initial recommendation for training a robust classifier is to have 55%–60% of the training data at the halfway point between the minimum and maximum range at which classification is desired. Of course, simply adding noise to the training and test data is not equivalent to operating at longer ranges; future work will use real long range data to validate the results of this work.

With an SNR of 20 dB, drones and birds can be classified with 92% accuracy. Since this is comparable to the SNR values required for robust detection, perhaps such a classification method paves the way for an improved method of detection, following a classify before detect approach.

In this work, each spectrogram consists of 40 frames of data, meaning that there is a latency of approximately 11 s before the target is classified. Obviously this will have some implications for operational use. To mitigate this, one might consider using a fused classification approach whereby a short-dwell time classifier (whether a CNN or a traditional machine learning approach) is used for the first 11 s before switching to the network discussed here. Note that after 11 s a label could be output every quarter of a second, although this may be highly correlated and the update rate may be reduced in practice. The latency could be further reduced by reducing the integration period, although this will negatively impact upon the Doppler sensitivity and SNR.

The results of this work are likely to be dataset dependent; data for this work was collected from a single site, and only a single model of drone was considered. Bird data could have come from number of species but the actual bird types were not known. The effects of SNR dependency are likely to vary on a drone-by-drone basis, depending on the RCS of the propellers relative to the body of the drone. SNR has been measured as the ratio of the body signal to the noise floor, where it is assumed that the drone signal sits approximately 15 dB below the body return. A drone model whose propellers have a smaller RCS relative to the body will likely require a higher body SNR for reliable classification. One could expect performance to degrade rapidly in the case where spectral returns from the propellers cannot be detected.

A benefit of staring radar is that the integration time can be arbitrarily long. It is postulated that increasing the integration time will enhance the SNR—provided coherent integration is possible—and that this in return will improve the visibility of target body and propeller returns, facilitating classification. Future work will consider varying the processing interval to raise the SNR, thus improving classification.

ACKNOWLEDGEMENTS

This work was supported by the Engineering and Physical Sciences Research Council, and UK National Quantum Technology Hub in Sensing and Timing (EP/T001046/1) project.

CONFLICT OF INTEREST

The authors declare that there are no conflict of interests.

ORCID

Holly Dale https://orcid.org/0000-0002-4018-9392

REFERENCES

1. Hassanalian, M., Abdelkefi, A.: Classifications, applications, and design challenges of drones: a review. Prog. Aerosp. Sci. 91, 99–131 (2017)
2. The impact of drones on the UK Economy – PwC UK. https://www.pwc.co.uk/dronesreport. Accessed December 2020
3. Military called in to help Gatwick drone crisis – The Guardian. https://www.theguardian.com/uknews/2018/dec/19/gatwick-flights-halted-after-drone-sighting. Accessed 15 March 2020
4. Ritchie, M., Fioranelli, F., Griffths, H.: Micro-drone RCS analysis. In: 2015 IEEE Radar Conference, pp. 452–456. Johannesburg, South Africa (2015)
5. Vaughn, C.R.: Birds and insects as radar targets: a review. Proc IEEE. 73(2), 205–227 (1985)
6. Rahman, S., Robertson, D.: Radar micro-Doppler signatures of drones and birds at K-band and W-band. Sci. Rep. 8, 17396 (2018)
7. Jahangir, M., Baker, C.J.: Persistence surveillance of difficult to detect micro-drones with L-band holographic radar. In: 2016 CIE International Radar Conference. Guangzhou, China (2016)
8. Mohajerin, N., et al.: Feature extraction and radar track classification for detecting UAVs in civillian airspace. In: 2014 IEEE Radar Conference, pp. 3759–3761. Cincinnati, OH (2014)
9. Chen, V.C., Miceli, W.J., Himed, B.: Micro-Doppler analysis in ISAR-review and perspectives. In: 2009 IEEE International Radar Conference - Surveillance for a Safer World (RADAR 2009), pp. 1–6. Bordeaux, France (2009)
10. Zaugg, S., et al.: Automatic identification of bird targets with radar via patterns produced by wing flapping. J. R. Soc. Interface. 5(26), 1041–1053 (2008)
11. Gannon, Z., Tahmoush, D.: Measuring UAV propeller length using micro-Doppler signatures. In: 2020 IEEE International Radar Conference, pp. 1019–1022. Washington, DC (2020)
12. Tait, P.: Radar target classification. In: IET Radar, Sonar and Navigation Series 18. Institution of Engineering and Technology (2005)
13. Regev, N., Yoffe, I., Wulich, D.: Classification of single and multi propelled miniature drones using multilayer perceptron artificial neural network. In: International Conference on Radar Systems (RADAR 2017), pp. 1–5. Belfast, Ireland (2017)
14. Molchanov, P., et al.: Classification of small UAVs and birds by micro-Doppler signatures. Int. J. Microw. Wirel. Technol. 6(3-4), 435–444 (2004)
15. Herr, D.B., Tahmoush, D.: Data-driven STFT for UAV micro-Doppler signature analysis. In: 2020 IEEE International Radar Conference (RADAR), pp. 1023–1028. Washington, DC (2020)
16. Markow, J., Balleri, A.: Examination of drone micro-Doppler and JEM/HERM signatures. In: 2020 IEEE Radar Conference. Florence, Italy (2020)
17. Huang, A., et al.: Fundamental frequency estimation of HERM lines of drones. In: 2020 IEEE International Radar Conference (RADAR), pp. 1013–1018. Washington, DC (2020)
18. Ritchie, M., et al.: Multistatic micro-Doppler radar feature extraction for classification of unloaded/loaded micro-drones. IET Radar Sonar Navig. 11(1), 116–124 (2017)
19. Pallotta, L., et al.: A feature-based approach for loaded/unloaded drones classification exploiting micro-Doppler signatures. In: 2020 IEEE Radar Conference. Florence, Italy (2020)
20. Ritchie, M., et al.: Monostatic and bistatic radar measurements of birds and micro-drone. In: 2016 IEEE Radar Conference (RADAR), pp. 1–5. Philadelphia, PA (2016)
21. Jahangir, M., Baker, C.J.: CLASS U-space drone test flight results for non-cooperative surveillance using an L-band 3-D staring radar. In: 2019 20th International Radar Symposium (IRS), pp. 1–11. Ulm, Germany (2019)
22. Jahangir, M., Ahmad, B.I., Baker, C.J.: Robust drone classification using two-stage decision trees and results from SESAR SAFIR trials. In: 2020 IEEE International Radar Conference (2020)
23. Bennett, C., et al.: Use of symmetrical peak extraction in drone micro-Doppler classification for staring radar. In: 2020 IEEE Radar Conference. Florence, Italy (2020)
24. Dale, H., et al.: An initial investigation into using convolutional neural networks for classification of drones. In: 2020 IEEE International Radar Conference. Washington, DC (2020)
25. Rahman, S., Robertson, D.A.: Classification of drones and birds using convolutional neural networks applied to radar micro-Doppler spectrogram images. IET Radar Sonar Navig. 14(5), 653–661 (2020)
26. Kim, B.K., Kang, H.S., Park, S.O.: Drone classification using convolutional neural networks with merged Doppler images. IEEE Geosci. Remote Sens. Lett. 14(1), 38–42 (2017)
27. Seyfioğlu, M.S., Gürbüz, S.Z.: Deep neural network initialization methods for micro-Doppler classification with low training sample support. IEEE Geosci. Remote Sens. Lett. 14(12), 2462–2466 (2017)
28. Taha, B., Shoufan, A.: Machine learning-based drone detection and classification: state-of-the-art in research. IEEE Access. 7, 138669–138682 (2019)

29. Patel, J.S., Fioranelli, F., Anderson, D.: Review of radar classification & RCS characterisation techniques for small UAVs or drones. IET Radar Sonar Navig. 12(9), 911–919 (2018)
30. Jahangir, M., Baker, C.J.: Robust detection of micro-UAS drones with L-band 3-D holographic radar. In: 2016 Sensor Signal Processing for Defence (SSPD). Edinburgh, UK (2016)
31. Sim, J., et al.: Effective ground-truthing of supervised machine learning for drone classification. In: 2019 International Radar Conference (RADAR). Toulon, France (2019)
32. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. Commun. ACM. 60(6), 84–90 (2017)
33. Park, J., et al.: Micro-Doppler based classification of human aquatic activities via transfer learning of convolutional neural networks. Sensors. 16, 1990 (2016)
34. Snoek, J., Larochelle, H., Adams, R.: Practical Bayesian optimization of machine learning algorithms. Adv. Neural Inf. Process. Syst. 4, 2951–2959 (2012)
35. Wu, J., et al.: Hyperparameter optimization for machine learning models based on Bayesian optimization. J. Electron. Sci. Technol. 17(1), 26–40 (2019)
36. Jones, D.R., Schonlau, M., Welch, W.J.: Efficient global optimization of expensive black-box functions. J. Glob. Optim. 13(4), 455–492 (1998)
37. Oh, H.M., Lee, H., Kim, M.Y.: Comparing convolutional neural network (CNN) models for machine learning-based drone and bird classification of anti-drone system. In: 19th International Conference on Control, Automation and Systems (ICCAS 2019), pp. 87–90. Jeju, South Korea (2019)

How to cite this article: Dale, H., et al.: SNR-dependent drone classification using convolutional neural networks. IET Radar Sonar Navig. 1–12 (2021). https://doi.org/10.1049/rsn2.12161