IET Radar, Sonar & Navigation

Research Article

IET Journals
The Institution of Engineering and Technology

300 GHz radar object recognition based on deep neural networks and transfer learning

ISSN 1751-8784
Received on 10th December 2019
Revised 2nd April 2020
Accepted on 14th April 2020
E-First on 15th July 2020
doi: 10.1049/iet-rsn.2019.0601
www.ietdl.org

Marcel Sheeny[1], Andrew Wallace[1] ✉, Sen Wang[1]

[1]*Institute of Sensors, Signals and Systems, Heriot-Watt University, Edinburgh, EH14 4AS, UK*

✉ *E-mail: a.m.wallace@hw.ac.uk*

Abstract: For high-resolution scene mapping and object recognition, optical technologies such as cameras and LiDAR are the sensors of choice. However, for future vehicle autonomy and driver assistance in adverse weather conditions, improvements in automotive radar technology and the development of algorithms and machine learning for robust mapping and recognition are essential. In this study, the authors describe a methodology based on deep neural networks to recognise objects in 300 GHz radar images using the returned power data only, investigating robustness to changes in range, orientation and different receivers in a laboratory environment. As the training data is limited, they have also investigated the effects of transfer learning. As a necessary first step before road trials, they have also considered detection and classification in multiple object scenes.

1 Introduction

All major car manufacturers are evaluating LiDAR, passive optical and radar sensing capabilities for automotive applications [1], aiming beyond advanced driver-assistance systems (ADASs) such as automatic cruise control, parking assistance and collision avoidance, towards full automotive autonomy. Each technology has benefits and drawbacks, but a key benefit of automotive radar is an operating range up to 150 m or more, and an ability to function in adverse weather, such as fog, rain or mist. However, radar sensors offer much lower resolution than optical technologies. Current automotive radar systems operate at 24 and 79 GHz, with a typical bandwidth of 4 GHz, can perform low-resolution mapping and detection in relatively uncluttered scenes, but object recognition is really challenging.

Deep neural networks (DNNs) have proven to be a powerful technique for image recognition on natural images [2–4]. In contrast to manual selection of suitable features followed by the statistical classification, DNNs optimise the learning process to find a wider range of patterns, achieving better results than formerly on quite complicated scenarios. For example, this includes the ImageNet challenge first introduced in 2009 [5], which has at the time of writing more than 2000 object categories and 14 million images.

In this paper, we investigate the capacity of DNNs to recognise prototypical objects in azimuth-range power spectral images by a prospective 300 GHz automotive radar with an operating bandwidth of 20 GHz. The first contribution of our work is to assess the robustness of these DNNs to variations in viewing angle, range and the specific receiver operational characteristics, using a simple database of six isolated objects. For greater realism, our second contribution is to evaluate the performance of the trained neural networks in more challenging scenarios with multiple objects in the same scene, including detection and classification in the presence of both uniform and a cluttered background. Third, since we have limited data, we have also investigated how transfer learning can improve the results. This 300 GHz prototype has limited range and scanning speed; therefore, our experiments are conducted in a laboratory setting rather than from a mobile vehicle. Further, we avoid the use of range-Doppler spectra to classify images, but perform experiments using the radar power data alone. This is justifiable because future automotive technology must have the capability to classify traffic participants even when static, e.g. at traffic lights, although motion may be used as an ancillary variable to good effect.

2 Related work

Together with scene mapping, object recognition is a necessary capability for autonomous cars. When we create a map of the immediate environment, we also need to identify key actors, such as pedestrians and vehicles, and other street furniture, traffic signs, walls, junctions and so on. For actors, we also wish to predict their movements in order to create a safe system, and identity is a key component of such prediction.

The use of deep convolutional neural networks (DCNNs) [2, 6] for large scale image recognition has changed significantly the field of computer vision. Although questions remain on verifiability [7], confidence in the results [8], and on the effects of adversarial examples [9], the best results for correct identifications applied to large image datasets have been dominated by DCNN algorithms. The development of GPU's and large annotated datasets has helped the popularity of deep learning methods in computer vision.

Of course, the results on natural image data such as ImageNet can be replicated to a large extent using automotive data, such as the KITTI benchmarks [10]. However, in adverse weather, optical sensors have poor performance, so we wish to examine the potential of radar data for reliable recognition. This is especially challenging; most automotive radars sense in two dimensions only, azimuth and range, although research is underway to develop a full 3D radar [11]. Although range resolution can be of the orders of *cm*, azimuth resolution is poor, typically 1°–2° although again there is active research to improve this [12]. Natural image recognition relies to a great extent on surface detail, but the radar imaging of surfaces is much less well understood, is variable, and full electromagnetic modelling of complex scenes is extremely difficult.

There has been some recent work in applying deep learning techniques to radar images for automotive applications, but the vast majority of these rely on the Doppler capabilities of radar as a feature to recognise the objects. For example, Wöhler *et al.* [13, 14] used Long Short-Term Memory (LSTM) neural networks to classify road actors in the automotive scenario in which the motion-compensated Doppler velocity was a key feature. Other broadly similar works include Rohling *et al.* [15], who used a 24 GHz radar to classify pedestrians by analysing the Doppler spectrum and range profile, Major *et al.* [16] who classified and detected vehicles in a highway scenario using a range-azimuth-Doppler spectrum based on 3D convolutions and LSTM networks, and Bartsch *et al.* [17] who classified pedestrians using the area

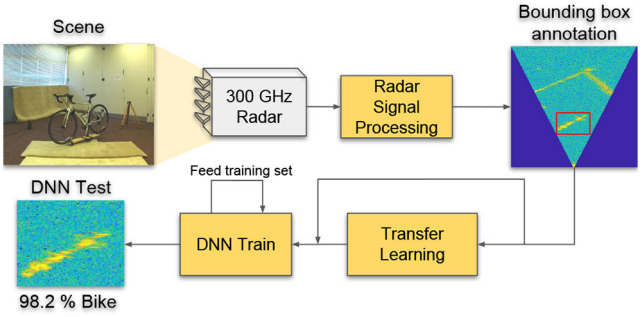

Fig. 1 *300 GHz FMCW radar object recognition: Methodology developed using DCNNs to process data acquired by a prototype high-resolution 300 GHz short-range radar [12]. Steps: 1. Radar signal processing and Cartesian radar image generation. 2. Bounding box annotation to crop object region. 3. DNN and transfer learning for radar-based recognition*

Table 1 300 GHz FMCW radar parameters for the system used in our experiments [21]

sweep bandwidth	20 GHz
H-plane (azimuth) beamwidth (−3 dB)	1.1°
E-plane (elevation) beamwidth (−3 dB)	7.0°
azimuthal scan angle increment	0.25°
range resolution	0.75 cm
azimuth resolution (at 10 m)	19.2 cm

and shape of the object and Doppler spectrum features. Bartsch *et al.* analysed the probability of each feature and used a simple decision model, achieving 95% accurate classification rates for optimal scenarios, but this dropped to 29.4% when the pedestrian was in close proximity to cars due to low resolution from the radar sensors. Of particular interest is the work of Angelov *et al.* [18] who investigated the capability of different DCNNs to recognise cars, people and bicycles with variable success rates ranging between accuracies of 44–88% depending on the problem. In Section 4, we use their network as a point of comparison. The conclusion from these studies is that prototypical motion from Doppler spectra can be a powerful aid to object identification, but with powerful caveats. First, a car is still a car if stationary at traffic lights, and second, for a moving vehicle equipped with sensors, the whole scene is moving, not just readily separable targets.

In contrast, Lombacher *et al.* [19] use the power spectrum alone to recognise a significant number of roadside objects with a 76 GHz radar system. There are several differences to the current paper. First, we use a higher frequency, 300 GHz, radar system with higher resolution in both range and azimuth. Second, Lombacher *et al.* used multiple images from different viewpoints to aggregate points from a moving car; this has an artificially created higher resolution that is not achievable in practice for a forward-looking radar. In our work, we consider a single view from such a radar system.

3 Applying DNNs to 300 GHz radar data

3.1 Objective

The main objective of the first part of our study is to design and evaluate a methodology for object classification in 300 GHz radar data using DCNNs, as illustrated schematically in Fig. 1. This is a prototype radar system; we have limited data, so we have employed data augmentation and transfer learning to examine whether this improves our recognition success. To verify the robustness of our approach, we have assessed recognition rates using different receivers at different positions, and objects at different orientations and range. We also evaluated the performance of the method in a more challenging scenario with multiple objects per scene.

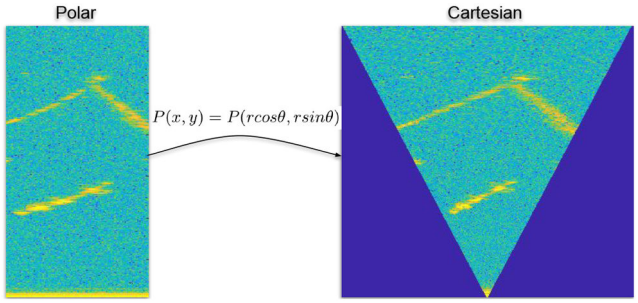

Fig. 2 *Polar to Cartesian radar image*

3.2 300 GHz FMCW radar

A current, typical commercial vehicle radar uses MIMO technology at 77–79 GHz with up to 4 GHz IF bandwidth, a range resolution of 4.3 cm, and an azimuth resolution of 15° [20]. This equates to a cross-range resolution of $\simeq$ 4 m at 15 m such that a car will just occupy one cell in the radar image. This clearly makes object recognition very challenging based on radar cross-section. Rather, in this work, we collected data using an FMCW 300 GHz scanning radar designed at the University of Birmingham [21]. The parameters for one of the 300 GHz sensors used in this work can be seen in Table 1. The main advantage of the increased frequency and bandwidth is a better-resolved radar image, which may lead to more reliable object classification. The 300 GHz radar used in this work has a bandwidth of 20 GHz, which equates to 0.75 cm range resolution. The azimuth resolution is 1.1° which corresponds to $\simeq$ 20 cm at 10 m.

The raw data captured by the 300 GHz radar is a time-domain signal at each azimuth direction. To transform the raw signal into an image, two steps were performed. The first step is to apply a fast Fourier transform (FFT) to each azimuth signal to create a range profile. The original polar image is converted to Cartesian coordinates, as shown in Fig. 2. This ensures equal dimensions in the x and y planes over all distances. Before training the neural network with this data, we applied whitening by subtracting the mean value of the image data, as this helps the stochastic gradient descent (SGD) to converge faster. The convergence happens faster because the weight initialisation of neural networks is based on a Gaussian distribution with zero mean [22]. It means that the bias term will have less influence during the learning process.

3.3 Experimental design and data collection

The main objective is to establish whether the proposed methodology has the potential to discriminate between a limited set of prototypical objects in a laboratory scenario, prior to collecting wild data in a scaled-down or alternate radar system. In the wild, by which we mean outside the laboratory and as a vehicle-mounted sensor navigating the road network, we anticipate even more problems due to overall object density and proximity of targets to other scene objects. In the laboratory, we wanted to gain knowledge of what features were important in 300 GHz radar data, and whether such features were invariant to the several possible

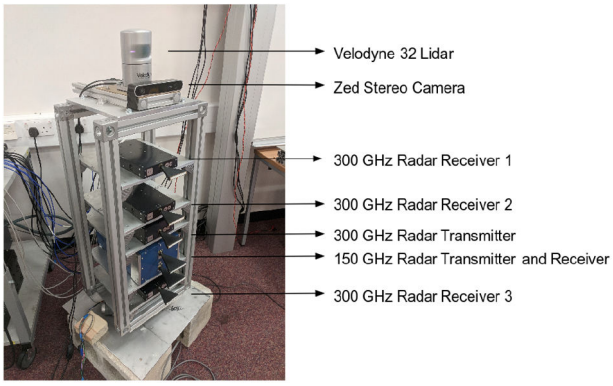

Fig. 3 *Experimental sensor setup*

Table 2 Data set collection showing a number of different raw images collected at each range

	3.8 m	6.3 m
bike	90	90
trolley	90	90
mannequin	90	90
cone	25	25
traffic sign	90	90
stuffed dog	90	90
total	475 × (3 rec.) = 1425	475 × (3 rec.) = 1425

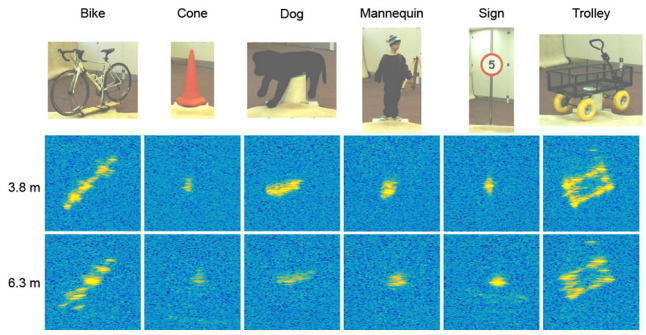

Fig. 4 *Sample images from each object from the dataset collected using the 300 GHz radar*

transformations. The objects we decided to use were a bike, trolley, mannequin, sign, stuffed dog and cone. Those objects contain varieties of shapes and materials, which to some extent, typify the expected, roadside radar images that we might acquire from a vehicle.

The equipment for automatic data collection included a turntable to acquire samples every 4°, covering all aspect angles, and at two stand-off distances, 3.8 and 6.3 m. The sensors are shown in Fig. 3. In collecting data, we used 300 and 150 GHz radars, a Stereo Zed camera and a Velodyne HDL-32e Lidar, but in this paper, only data from the 300 GHz radar is considered. The 300 GHz radar has one transmitter and three receivers. The three receivers were used to compare the object signatures at different heights, and to a lesser extent, whether the different receivers had different operational characteristics. We used a carpet below the objects to avoid multi-path and ground reflections. Table 2 summarises how many samples were captured from each object at each range. Since we have 3 receivers, we have 1425 images from each range and 2850 images in total. In Fig. 4, we can see sample images from all objects at different ranges using receiver 3.

All the collected images were labelled with the correct object identity, irrespective of viewing range, angle and receiver height. A fixed-size bounding box of 400 × 400 cells, which corresponds to 3 m × 3 m, was cropped from the image with the object in the middle of the box.

3.4 Neural network architecture

In this work, we used four networks, which are illustrated in Fig. 5.

- CNN-2: This network is a vanilla CNN with two convolutional layers and a fully connected layer in the end to classify the objects.
- CNN-3: This network is the same as CNN-2 with an additional convolutional layer.
- VGG-like: The VGG-like network was developed by Angelov *et al.* [18] for range-Doppler radar object recognition.
- A-ConvNet: This network developed by Chen *et al.* [23] achieved state-of-the-art recognition on SAR target recognition.

All networks contain standard layers such as a convolutional layer, rectified linear unit (ReLU), max pooling, dropout, fully connected and softmax layers. A description of the properties of all these layers can be found in [24]. The CNN-2 and -3 networks provide a baseline solution of minimal complexity. The VGG-like network was chosen as it provided a very recent point of comparison on a similar problem, of course, with the significant difference that it was designed for range-Doppler data. Finally, we chose the A-ConvNet architecture because it was also employed to recognise static objects in radar images, albeit synthetic aperture radar (SAR) images. This also allowed us to investigate transfer learning using this same network trained on the SAR data and sharing the initial weights. For all networks, we decided to use the original input

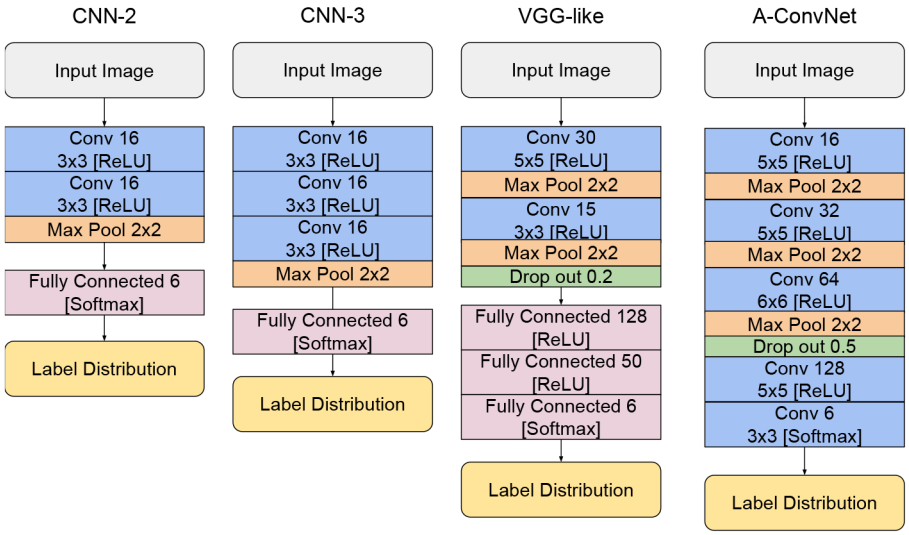

Fig. 5 *Networks architectures used*

Table 3 Neural network parameters

learning rate (α)	0.001
momentum (η)	0.9
epochs	100
batch size	100

Table 4 Set of experiments performed

	Train	Test
experiment 1	random (70%)	random (30%)
experiment 2	two receivers	one receiver
experiment 3	one range	other range
experiment 4	quadrants 1, 3	quadrants 2, 4

Table 5 Accuracy for Experiment 1: random selection from all data

	CNN-2	CNN-3	VGG-Like	A-ConvNet
random selection	92.3%	94.9%	96.4%	**99.7%**

Bold value indicates the best result.

layer of A-ConvNet (88×88), so our input data was resized using bilinear interpolation.

To train our neural network, we used stochastic gradient descent (SGD). SGD updates the weights of the network depending on the gradient of the function that represents the current layer, as in the following equation:

$$W_{t+1} = W_t - \alpha \nabla f(x; W) + \eta \Delta W \tag{1}$$

In (1), η is the momentum, α is the learning rate, t is the current time step, W defines the weights of the network and $\nabla f(x; W)$ is the derivative of the function that represents the network. To compute the derivative for all layers, we need to apply the chain rule, so we can compute the gradient through the whole network. The loss function used to minimise was the categorical cross-entropy (2). The parameters used in all experiments in all training procedures are given in Table 3. For all experiments, we used 20% of the training data as validation, and we used the best results from the validation set to evaluate the performance. In (2), $\hat{\boldsymbol{y}}$ is the predicted vector from softmax output and y is the ground truth.

$$L(\hat{\boldsymbol{y}}, y) = -\sum_i y_i \log(\hat{y}_i) \tag{2}$$

3.5 Data augmentation

As shown in Table 2, we have limited training data. Using a restricted dataset, the DCNNs will easily overfit and be biased towards specific artefacts in the dataset. To help overcome this problem, we generated new samples to create a better generalisation. The simple technique of random cropping takes as input the image data of size 128×128 and creates a random crop of 88×88. This random crop ensures that the target is not always fixed at the same location, so that the location of object should not be a feature. We cropped each sample eight times and also flipped all the images left to right to increase the size of the dataset and remove positional bias.

4 Experiments: classification of isolated objects

As described in Section 3.3, we used 6 objects imaged from 90 viewpoints with 3 receivers at two different ranges (3.8 and 6.3 m). Four different experiments were performed, as shown in Table 4. The metric used to evaluate the results is accuracy, i.e. the number of correct divided by the total number of classifications in the test data.

Experiment 1: Random selection from the entire data set: This is the often used, best-case scenario, with a random selection from all available data to form training and test sets. Intuitively, the assumption is that the dataset contains representative samples of all possible cases. To perform this experiment, we randomly selected 70% of the data as training and 30% as test data. The results are summarised in Table 5.

From Table 5, we conclude that the results are very high across the board, so it is possible to recognise objects in the 300 GHz radar images, with the considerable caveats that the object set is limited, they are at short range in an uncluttered environment, and as all samples are used to train, then any test image will have many near neighbours included in the training data with a high statistical probability.

Experiment 2: Receiver/height influence: The second experiment was designed to investigate the influence of the receiver antenna characteristics and height (see Fig. 3). The potential problem is that the DCNNs may effectively overfit the training data to learn partly the antenna pattern from a specific receiver or a specific reflection from a certain height. All available possibilities were tried, i.e.

- Experiment 2.1: Receivers 2 and 3 to train and receiver 1 to test.
- Experiment 2.2: Receivers 1 and 3 to train and receiver 2 to test.
- Experiment 2.3: Receivers 1 and 2 to train and receiver 3 to test.

Table 6 shows the results for Experiment 2. In comparison with Experiment 1, the results are poorer, but not to the extent that we can determine as significant on a limited trial. This was expected

Table 6 Accuracy for Experiment 2: receiver influence

	CNN-2	CNN-3	VGG-Like	A-ConvNet
receiver 1 test experiment	82.6%	85.9%	81.1%	**98.9%**
receiver 2 test experiment	90.0%	93.2%	93.5%	**98.4%**
receiver 3 test experiment	60.4%	65.9%	65.6%	**87.7%**

Bold values indicates the best results.

Table 7 Accuracy for Experiment 3: range influence

	CNN-2	CNN-3	VGG-Like	A-ConvNet
object at 6.3 m test experiment	58.3%	60.2%	59.6%	**82.5%**
object at 3.8 m test experiment	58.5%	69.5%	37.7%	**91.1%**

Bold values indicates the best results.

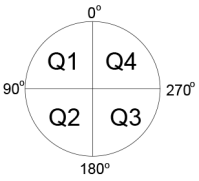

Fig. 6 *Quadrants*

Table 8 Accuracy for Experiment 4: orientation influence

	CNN-2	CNN-3	VGG-Like	A-ConvNet
Q2,Q4 test experiment	76.5%	78.3%	92.2%	**92.5%**

Bold values indicates the best results.

Fig. 7 *MSTAR dataset*

from the examination of the raw radar data, since there is not much difference in the signal signatures from the receivers at different heights. If anything, receiver 3, which was closest to the floor and so, received more intense reflections, gave poorer results when used as the test case, which implied that the DCNNs did include some measure of receiver or view-dependent characteristics from the learnt data. In this instance, the drop in performance is markedly less severe in the preferred A-ConvNet architecture.

Experiment 3: Range influence: Clearly, the range of the object influences the return signature to the radar as the received power will be less due to attenuation, and less cells are occupied by the target in the polar, radar image due to degrading resolution over azimuth. Therefore, if the training data set is selected only at range 3.8 m, for example, to what extent are the features learnt representative of the expected data at 6.8 m (and vice versa)? Table 7 summarises the results achieved when we used one range to train the network, and the other range to test performance.

- Experiment 3.1: Train with an object on 3.8 m. Test with an object on 6.3 m.
- Experiment 3.2: Train with an object on 6.3 m. Test with an object on 3.8 m.

The key observation from Table 7 is that if we train the DCNNs at one specific range which has a given cell structure and received power distribution, and then test at a different range, the performance is not as accurate as in the base case as this drops from over 99 to 82.5% and 91.1%, respectively, in the case of A-ConvNet. Again, the other networks do not perform as well.

Experiment 4: Orientation influence: The final experiment was designed to examine whether the neural network was robust to change of viewing orientation. Here, we used as training sets the objects in quadrants 1 and 3, and as test sets the objects in quadrants 2 and 4. Quadrant 1 means orientation from 0° to 89°, quadrant 2 means orientation from 90° to 179°, quadrant 3 means orientation from 180° to 269° and quadrant 4 means orientation from 270° to 359°, as shown in Fig. 6.

The DCNNs do not perform as well compared to Experiments 1 and 2, for example dropping to 92.5% for A-ConvNet. However, since we flipped the images left to right as a data augmentation strategy, the network was capable of learning the orientation features, as the objects exhibited near mirror symmetry, and in one case, the cone, is identical from all angles. Therefore, we have to be hesitant in drawing conclusions about any viewpoint invariance within the network as the experiments are limited and all objects have an axis or axes of symmetry (as do many objects in practice).

Together with Experiments 2 and 3, this experiment shows that it is necessary to take into account the differences in the acquisition process using different receivers at different ranges and orientation in training the network. While, this is to some extent obvious and equally true for natural images, we would observe that the artefacts introduced by different radar receivers are much less standardised than those introduced by standard video cameras, so the results obtained in future may be far less easy to generalise. Although Experiment 2 only showed limited variation in such a careful context, we would speculate that the effects of multi-path and clutter would be far more damaging than in the natural image case, as highlighted in [17].

4.1 Comparison between the networks

As shown in Tables 5–8, in all scenarios, A-ConvNet was superior. The CNN-2 and CNN-3 networks are the baseline, and it shows that without much engineering, we manage to have networks with suitable results, however, they are not as good as a network designed for SAR target recognition. Angelov *et al.* [18] designed a network for a range-Doppler image, which in our scenario (just using the power spectra) did not manage to work as well as A-ConvNet. The A-ConvNet architecture was designed for the most similar problem, and for these experiments achieved very good results. Since A-ConvNet was shown to be the superior network of the ones presented here, succeeding experiments in this paper use this architecture.

4.2 Transfer learning

As summarised in Table 2, we have a small dataset and there is the potential to learn image-specific characteristics rather than features of the objects themselves. Therefore, we have investigated the use of transfer learning to help capture more robust features using a pre-existing dataset, i.e. to use prior knowledge from one domain and transfer it to another [25]. To apply transfer learning, we first trained the DCNNs on the MSTAR (source) data, then the weights from the network were used as initial weights for a new DCNNs trained on our own 300 GHz (target) data. The MSTAR data is different in viewing angle and range compared to our own data as shown in Fig. 7. It was developed to recognise military targets using SAR images. The data contains 10 different military targets and around 300 images per target with similar elevation viewing angles of 15 and 17. In total, MSTAR has around 6000 images and

Table 9 Accuracy after applying transfer learning

		Without TL	With TL
Exp 1:	random split Exp.	**99.7%**	99.1%
Exp 2.1:	Rec. 1 test Exp.	**98.9%**	95.8%
Exp 2.2:	Rec. 2 test Exp.	98.4%	**98.8%**
Exp 2.3:	Rec. 3 test Exp.	87.7%	**94.1%**
Exp 3.1:	6.3 m test Exp.	82.5%	**85.2%**
Exp 3.2:	3.8 m test Exp.	91.1%	**93.5%**
Exp 4:	Q2, Q4 test Exp.	92.5%	**98.5%**

Bold values indicates the best results.

Table 10 Orientation experiment trained on A-ConvNet without transfer learning

Acc: 0.925		Predicted label					
		Bike	Trolley	Cone	Mannequin	Sign	Dog
true label	bike	1.00	0.00	0.00	0.00	0.00	0.00
	trolley	0.03	0.97	0.00	0.00	0.00	0.00
	cone	0.00	0.00	1.00	0.00	0.00	0.00
	mannequin	0.00	0.00	0.00	0.86	0.00	0.14
	sign	0.00	0.00	0.00	0.00	1.00	0.00
	dog	0.03	0.00	0.02	0.10	0.00	0.86

Table 11 Orientation experiment trained on A-ConvNet with transfer learning from MSTAR

Acc: 0.985		Predicted label					
		Bike	Trolley	Cone	Mannequin	Sign	Dog
true label	bike	1.00	0.00	0.00	0.00	0.00	0.00
	trolley	0.00	1.00	0.00	0.00	0.00	0.00
	cone	0.00	0.00	1.00	0.00	0.00	0.00
	mannequin	0.00	0.00	0.00	0.96	0.00	0.04
	sign	0.00	0.00	0.00	0.00	1.00	0.00
	dog	0.00	0.00	0.00	0.03	0.00	0.97

is used widely by the radar community in order to verify classification algorithms.

The DCNNs function in the source domain is defined by the following equation:

$$y_s = f(W_s, x_s) \tag{3}$$

where W_s are the weights of a network, x_s and y_s are the input and output from the source domain. To learn the representation, an optimiser must be used, again stochastic gradient descent (SGD), expressed by the following equation:

$$W_{s_{i+1}} = \mathrm{SGD}(W_{s_i}, x_s, y_s) \tag{4}$$

where *SGD* is a function that updates the weights of the neural network, as expressed in (1). Hence, using the trained weights from our source domain as the initial weights, this is expressed as (5). It is intended that the initial weights give a better initial robust representation, which can be adapted to the smaller dataset. W_{t_1} represents the first step of the SGD before we start to train and W_s is the trained weights from the source dataset

$$W_{t_1} = \mathrm{SGD}(W_s, x_t, y_t) \tag{5}$$

We repeated Experiments 1–4 using transfer learning. The results are summarised in Table 9. To gain further insight, we also show the confusion matrix from the orientation experiments without and with transfer learning in Tables 10 and 11. The main confusion is between the dog and mannequin, since both have similar clothed material; and cone and sign, since they have a similar shape.

4.3 Effect of transfer learning

As can be seen, transfer learning gives higher values for accuracy in the majority but not all cases. The MSTAR dataset is a much bigger dataset, and although it exhibits some characteristics in common with our own data, it uses a synthetic aperture technique, and there is no significant variation in elevation angle during data collection. However, there are two distinguishable strong features, the shape and reflected power, and like our data, the objects are viewed at all possible rotations in the ground plane. As these characteristics have much in common with our own data, it is possible that the network is able to better generalise to cope with new situations as shown, for example in the Receiver 3 and different range experiments. To draw any firmer conclusion requires much more extensive evaluation.

Nevertheless, in these experiments, we can conclude that the neural network approach is robust in maintaining accuracy with respect to sensor hardware, height, range and orientation.

4.4 Visualisation of feature clusters

To better understand what is being learned by our network, the t-stochastic neighbour embedding technique (t-SNE) [26] was used to visualise the feature clusters. t-SNE employs nonlinear dimensionality reduction to build a probability distribution by comparing the similarity of all pairs of data, then transformed into a lower dimension. Then it uses Kullback–Leibler (KL)-divergence to minimise with respect to the locations in the cluster space.

Fig. 8 shows the result from the t-SNE clustering of samples using raw image features; in this case, the orientation experiment. Figs. 8*b* and *c* show the t-SNE clusters from the features extracted from the penultimate layer of the trained neural network with and without transfer learning, using different colour maps for each object for better visualisation. First, we can see that the trained neural network was able to cluster similar classes and similar features in each case. Second, transfer learning shows slight

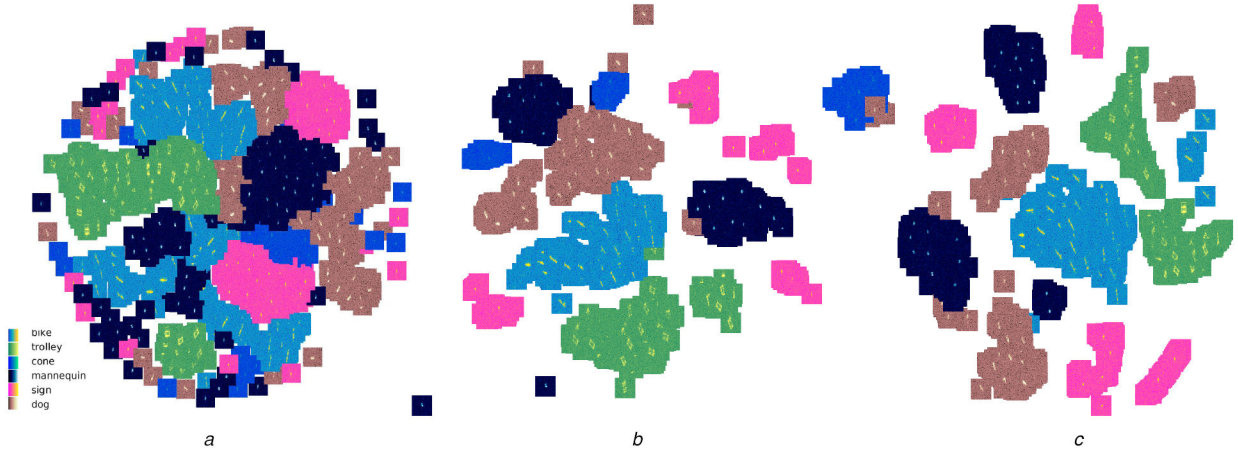

Fig. 8 *t-SNE plots from the orientation experiment*
(a) t-SNE using raw features, ***(b)*** t-SNE without transfer learning, ***(c)*** t-SNE with transfer learning

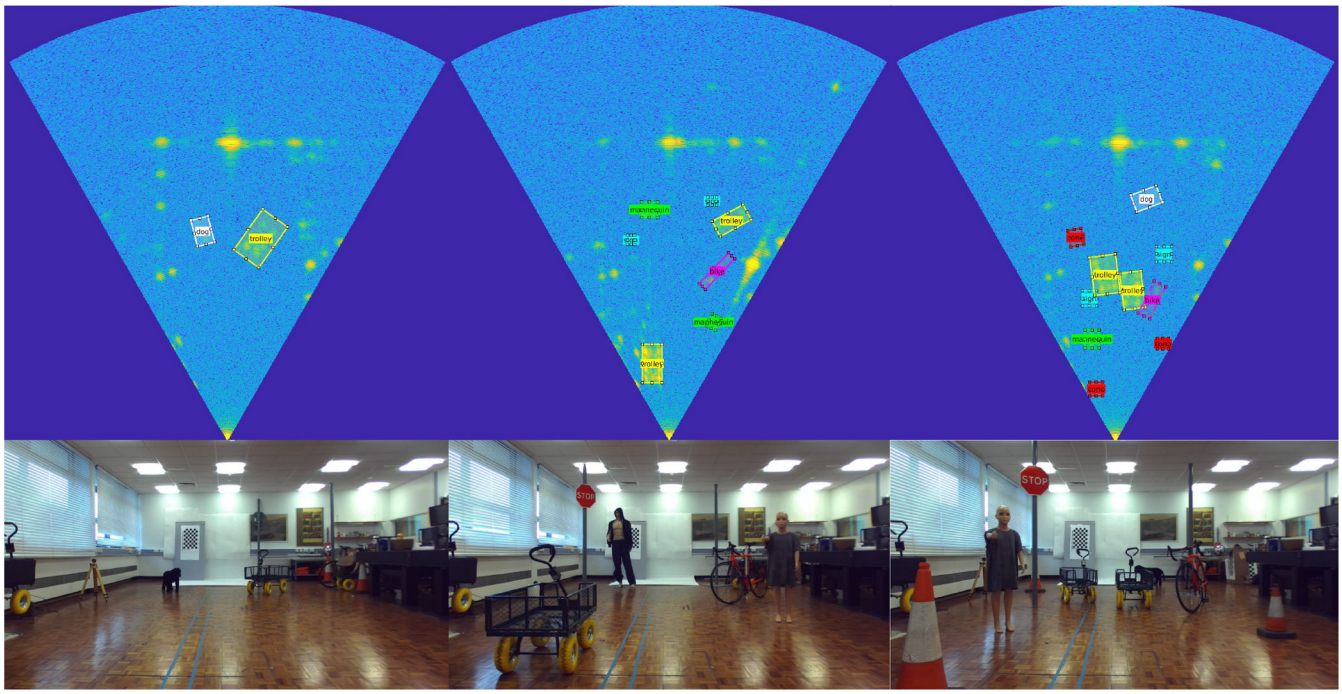

Fig. 9 *Multiple object dataset. Above: 300 GHz radar image. Below: reference RGB image*

improvement by creating larger, better-separated clusters of objects of the same class. Although it is hard to give actual interpretability of neural networks, the t-SNE framework can give some insights into the type of features that have been learned.

5 Experiments: detection and classification within a multiple object scenario

The previous dataset contains one windowed object in each image. In an automotive or more general radar scenario, we must both detect and classify road actors in a scene with many pre-learnt and unknown objects. This is much more challenging. Hence, in the next set of experiments, we include multiple objects, and this has several additional phenomena, including occlusion, multi-path and interference between objects, as well as objects which are not included as a learnt object of interest. We use the same object dataset (bike, trolley, cone, mannequin, sign, dog) in different parts of the room with arbitrary rotations and ranges, and the network is trained by viewing the objects in isolation, as before. We also include some within-object variation, using, for example different mannequins, trolleys and bikes. The unknown, laboratory walls are also very evident in the radar images. This new dataset contains 198 scenes, 648 objects, an average of 3.27 movable objects per scene. Fig. 9 shows the examples of 3 scenes in the multiple object dataset. Fig. 10 shows the statistical data explaining the number of instances of each learnt object, the number of objects in each scene, and the distribution of ranges of the objects. Fig. 11 illustrates the possible problems that can occur in multiple objects dataset.

5.1 Methodology

In classical radar terminology, detection is described as 'determining whether the receiver output at a given time represents the echo from a reflecting object or only noise' [27]. Conversely, in computer vision, using visible camera imagery to which the vast majority of CNN methods have been applied, detection is the precise location of an object in an image (assuming it is present) containing many other objects, as for example in the pedestrian detection survey of Dollar *et al.* [28]. Although the image may be noisy, this is generally not the major cause of false alarms.

The extensive literature on object detection and classification using cameras, e.g. [29–32], can be grouped into *one-stage* and *two-stage* approaches. In the *one-stage* approach, localisation and classification are done within a single step, as with the YOLO [32], RetinaNet [31] and SSD [30] methods. Using a *two-stage* approach, first localises objects, proposing bounding boxes and then performs classification in those boxes. R-CNN [33], fast R-CNN [34] and faster R-CNN [29] are examples of the *two-stage* approach.

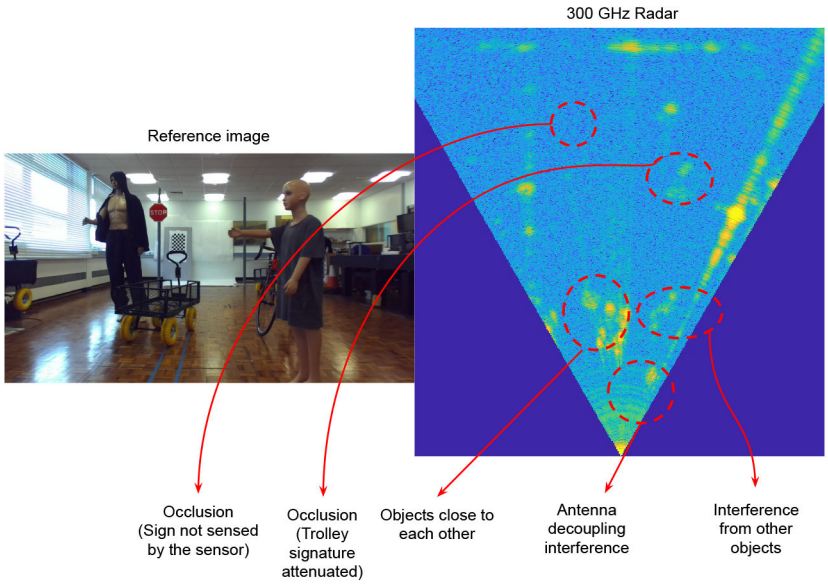

Fig. 10 *Possible unwanted effects in the multiple object dataset*

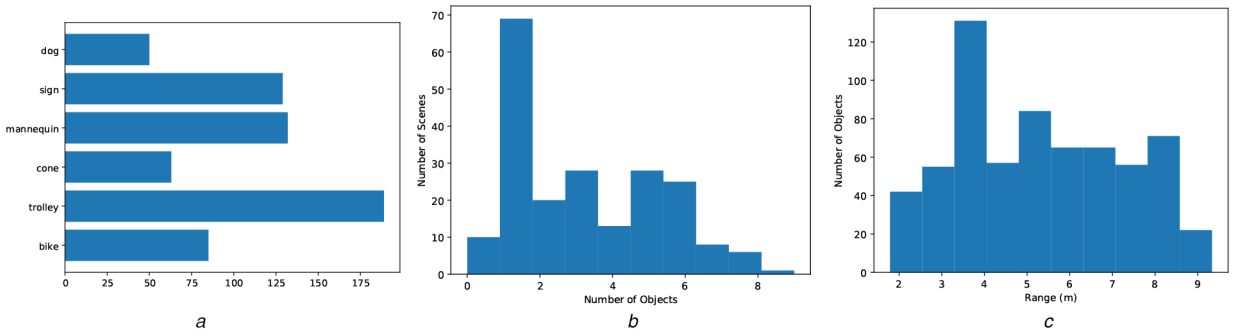

Fig. 11 *Multi-object dataset statistics*

(a) Number of object instances per class, ***(b)*** Number of objects per scene, ***(c)*** Distribution of object range

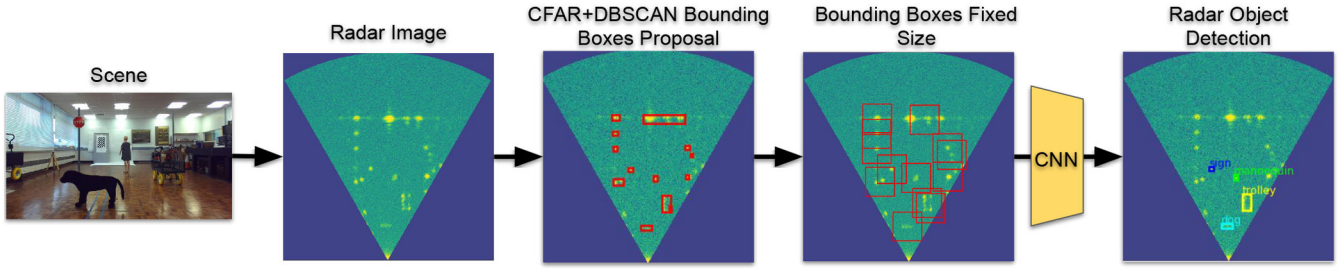

Fig. 12 *Methodology developed for the detection task*

For this work, we developed a two-stage technique. We first generate bounding boxes based on the physical properties of the radar signal, then the image within each bounding box is classified, similar to the R-CNN [33]. Fig. 12 shows the pipeline of the detection methodology developed. For radar echo detection, we use simply *Constant False Alarm Rate* (CFAR) [27] detection. There are many variations, including Cell Averaging Constant False Alarm Rate (CA-CFAR) and Order Statistics Constant False Alarm Rate (OS-CFAR). In this work, we used the CA-CFAR algorithm to detect potential radar targets. In order to compute the false alarm rate, we measured the background noise level, and the power level from the objects, setting a CFAR level of 0.22. After detecting potential cells, we form clusters using the common *Density-based spatial clustering of applications with noise* (DBSCAN) algorithm [35], which forms clusters from proximal points and removes outliers. For each cluster created we use the maximum and minimum points to create a bounding box of the detected area. The parameters for DBSCAN were selected empirically; $\epsilon = 0.3\,\text{m}$ which is the maximum distance of separation between 2 detected points, and $S = 40$, were S is the minimum number of points to form a cluster.

To compute the proposed bounding boxes with DBSCAN, we use the centre of the clusters to generate fixed-size bounding boxes of known dimensions, since, in contrast to the application of CNNs to camera data, the objects in radar images are fixed scale over a range. Hence, the boxes are of size 275×275, the same size as the data used to train the neural network for the classification task. The image is resized to 88×88 and each box is classified. As with the isolated objects experiments, we used the A-ConvNet architecture.

To consider the background, we randomly cropped 4 boxes which do not intersect with the ground truth bounding boxes containing objects in each scene image from the multiple object dataset and incorporated these in our training set. However, as there are effectively two types of background, that which contains other unknown objects such as the wall, and the floor areas which have low reflected power, we ensured that the random cropping contained a significant number of unknown object boxes. This is not ideal, but we are limited to collect data in a relatively small

laboratory area due to the restricted range of the radar sensor and cannot fully model all possible cluttering scenarios.

5.2 Results for multiple objects

In order to evaluate performance, we have considered three different scenarios. In particular, we wish to ascertain how performance is affected by failures in classification, assuming a perfect CFAR + DBSCAN pipeline, and to what extent failures in the box detection process lead to misclassification. Further, we make a distinction between confusing objects (mainly the lab wall) and due to system noise from the floor area.

- *Perfect detector:* In this scenario, we do not use the CFAR + DBSCAN pipeline; we use the labelled ground truth to form the detected bounding boxes. Each bounding box is fed to the trained neural network.
- *Easy:* In this scenario, we manually crop the walls and focus on the potential area containing objects of interest. This includes the CFAR + DBSCAN in an easy scenario, in which the removal of static objects is analogous to background subtraction.
- *Hard:* In this scenario, we assume the whole scene has potential targets. Hence, the wall should result in positive detections and is a challenge to the CNN classification.

We also decided to label our scene data depending on the density of objects, since a highly cluttered scene should increase the likelihood of unwanted radar sensing effects, such as multi-path, occlusion, and multiple objects in the same bounding box.

- *#Objects <4:* At a low density of objects, it is likely that the scene will suffer less from these effects.
- *4≤#Objects <7:* At mid-density, we will encounter some of the unwanted effects.
- *#Objects ≥7:* At high density, many of these effects occur.

We also have decided to evaluate performance at different ranges:

- Short-range (objects <3.5 m): This scenario is not necessarily the easiest since coupling between the transmitter and receiver happens at this range [36].
- Mid-range (3.5 m <objects 7 m): This is the ideal scenario, as the objects were learnt within these ranges, and the antenna coupling interference is reduced.
- Long-range (objects >7 m): This is the most challenging scenarios. At >7 m, most of the objects have low power of return, close to background noise.

The metric we use for evaluation is average-precision (AP), which is a commonly used standard in the computer vision literature for object detection, classification and localisation [37] in which the Intersection over Union (IoU) measures the overlap between 2 bounding boxes. If the overlap is greater than 0.5 and the classification is correct, then this is a true positive. To compute AP we need to compute precision (6) and recall (7), where TP is true positive, FP is a false positive and FN is a false negative. To compute AP we compute the area under the curve from the precision–recall plot varying the confidence level of the prediction of each bounding box. The AP is computed as shown in the (8) where p is precision and r is recall

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{6}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{7}$$

$$\text{AP} = \int_0^1 p(r)\mathrm{d}r \tag{8}$$

For these experiments, we retrained the neural network from the single object dataset using the orientation experiments. For the *Easy* and *Perfect Detector* cases, we do not include the background data. For the *Hard* case, we also added four background images per scene inside our training set. Extensive results for all these scenarios are shown in Tables 12–14.

As expected, the results from a scene containing many known objects and confusing artefacts are much poorer than when the objects are classified from images of isolated objects. Nevertheless, the results show promise. For example, considering the mid-range *Perfect Detector* case, there is an overall mean average precision (mAP) of 61.36%, and for specific easily distinguishable objects such as the trolley, it is as high as 97.06% in one instance. Other objects are more confusing, for example cones usually have low return power and can be easily confused with other small objects. As also expected, the results degrade at long range and in scenes with a higher density of objects.

The *Easy* case shows performance comparable but not as good as the *Perfect Detector*, for example dropping to 50.35% mAP. The CFAR + DBSCAN method is a standard option to detect objects in

Table 12 Perfect detector

AP	Overall	$\#Objects < 4$				$4 \leq \#Objects < 7$				$\#Objects \geq 7$				Short	Mid	Long
		Overall	Short	Mid	Long	Overall	Short	Mid	Long	Overall	Short	Mid	Long			
bike	64.88	79.17	50.00	83.33	75.00	48.42	25.00	56.77	33.33	76.26	N/A	67.05	N/A	35.00	66.19	57.14
cone	46.87	50.00	50.00	50.00	50.00	58.29	55.56	83.33	N/A	42.49	68.75	26.67	N/A	62.07	43.30	3.57
dog	51.34	77.62	77.78	87.72	47.62	49.13	77.78	55.45	33.33	26.40	60.0	N/A	12.50	70.95	65.02	20.19
mannequin	37.73	70.53	53.33	85.71	33.33	25.57	36.36	30.00	8.00	37.78	14.29	50.00	22.35	33.08	48.72	13.61
sign	85.64	81.86	0.00	89.47	66.67	86.60	N/A	90.10	81.08	86.44	N/A	88.89	85.46	0.00	89.65	81.94
trolley	81.68	87.75	79.17	97.06	82.35	85.35	100.00	87.61	70.13	75.45	92.67	83.65	10.00	93.53	88.76	60.41
mAP	61.36	74.49	51.71	82.22	59.16	58.89	58.94	67.21	37.65	57.47	58.93	63.25	26.06	49.1	66.94	39.48

Table 13 CFAR + DBSCAN detector easy

AP	Overall	$\#Objects < 4$				$4 \leq \#Objects < 7$				$\#Objects \geq 7$				Short	Mid	Long
		Overall	Short	Mid	Long	Overall	Short	Mid	Long	Overall	Short	Mid	Long			
bike	53.97	79.17	50.00	66.67	100.0	43.79	50.00	36.28	6.67	43.80	N/A	31.82	N/A	50.00	42.39	65.08
cone	19.49	36.36	50.0	16.67	0.00	47.60	55.56	60.00	N/A	2.12	0.00	3.81	N/A	23.71	18.16	0.00
dog	34.32	53.36	77.78	77.35	12.12	31.09	50.00	37.01	0.00	18.18	60.0	N/A	0.00	64.00	47.33	3.33
mannequin	36.91	70.57	64.00	85.71	16.67	21.51	32.73	26.67	5.83	39.66	0.00	52.94	29.41	29.57	48.72	14.22
sign	81.84	81.86	0.00	89.47	66.67	81.65	N/A	84.88	77.17	83.89	N/A	83.33	84.5	0.00	85.02	79.31
trolley	75.55	77.30	67.42	87.72	71.56	81.56	97.44	87.32	56.73	71.33	79.56	74.49	13.33	82.46	80.78	51.62
mAP	50.35	66.44	51.53	70.60	44.50	51.20	57.14	55.36	24.40	43.16	34.89	49.28	25.45	41.62	53.73	35.60

Table 14 CFAR + DBSCAN detector hard

AP	Overall	$\#Objects < 4$				$4 \leq \#Objects < 7$				$\#Objects \geq 7$				Short	Mid	Long
		Overall	Short	Mid	Long	Overall	Short	Mid	Long	Overall	Short	Mid	Long			
bike	37.71	52.77	16.67	62.96	50.00	51.29	20.00	44.48	30.00	16.78	N/A	7.11	N/A	14.41	36.11	27.76
cone	6.35	8.33	25.00	0.00	0.00	17.65	22.22	16.67	N/A	0.00	0.00	0.00	N/A	10.34	3.7	0.00
dog	31.00	43.49	58.73	42.86	0.00	30.77	50.00	16.67	0.00	9.09	10.00	N/A	0.0	37.09	35.0	0.0
mannequin	7.95	37.39	66.67	35.71	0.00	3.92	0.00	6.67	0.00	0.00	0.00	0.00	0.00	16.67	8.97	0.0
sign	62.27	75.73	100.00	79.31	66.67	63.91	N/A	68.79	61.05	53.46	N/A	41.85	75.07	100.00	62.68	64.77
trolley	65.8	64.52	90.48	80.7	39.53	63.46	97.44	90.54	20.33	70.55	81.15	76.55	9.76	86.18	78.98	20.14
mAP	35.18	47.04	59.59	50.26	26.03	38.50	37.93	40.63	18.56	24.98	22.79	25.10	16.97	44.12	37.58	18.78

radar, but it does introduce some mistakes where, for example the bounding box is misplaced with respect to the learnt radar patterns.

Regarding the *Hard* case, the mAP drops significantly to 35.18%. This shows how hard it is to recognise objects in radar images when the scene contains other, unseen and un-learnt, objects. Indeed, when the density of objects is greater than 7, some mAP values for bike, cone and mannequin are actually 0.00, which means that those objects were not recognised under those specific conditions.

Finally, we observe that the trolley is the easiest object to recognise in all cases. The trolley has a very characteristic shape, and strongly reflecting metal corner sections that create a distinguishable signature from all other objects. In interpreting true and false results in non-standardised datasets, which is the case in radar as opposed to visible camera imagery, one should be careful when comparing diverse published material.

6 Conclusions

In this work, we evaluated the use of DCNNs applied to images from a 300 GHz radar system to recognise objects in a laboratory setting. Four types of experiments were performed to assess the robustness of the network. These included the optimal scenario when all data are available for training and testing at different ranges, different viewing angles, and using different receivers. As expected, this performs best when all the training and test data are drawn from the same set. This is a valuable experiment as it sets an optimal benchmark, but this is not a likely scenario for any radar system applied in the wild, first because radar data is far less ubiquitous or consistent than camera data, and second because the influence of clutter and multi-path effects are potentially more serious than for optical technology.

Regarding the single object scene data, we should be encouraged by two principal results, first that the performance was so high for the optimal case, and second that transfer learning may lead to improvements in other cases, Transfer learning from MSTAR using A-ConvNet can prevent overfitting to the 300 GHz source data by generalising using more samples from a different radar data set, e.g. increasing from 92.5 to 98.5% in the experiment using Q1 and Q3 to train and Q2 and Q4 to test. This leads to a more robust classification.

The multiple object dataset is a very challenging scenario, but we achieved mean average precision rates in the easy case >60% (<4 objects per scene), but much less, 35.18%, in a high cluttered scenario. However, the pipeline we have adopted is probably subject to improvement, in particular, using the classification results to feedback to the detection and clustering. To avoid problems with occlusion, object adjacency, and multi-path, further research on high-resolution radar images is necessary. We also note that we have not made use of Doppler processing, as this implies motion of the scene, the sensor or both. For automotive radar, there are many stationary objects (e.g. a car at a traffic light), and many different motion trajectories in the same scene, so this too requires further research.

7 Acknowledgments

This work was supported by Jaguar Land Rover and the UK Engineering and Physical Research Council, grant reference EP/N012402/1 (TASCC: Pervasive low-TeraHz and Video Sensing for Car Autonomy and Driver Assistance (PATH CAD)). The authors acknowledge particularly the work done by the Birmingham group led by Marina Gashinova in designing and building the 300 GHz radar used for these experiments, and the practical help of Liam Daniel and Dominic Phippen in operating the radar and advising us on the authors' experimental design. They should also like to thank David Wright at the University of Edinburgh for the phase correction code. They thank NVIDIA for the donation of the TITAN X GPU.

8 References

[1] Guerrero-Ibanez, J., Zeadally, S., Contreras-Castillo, J.: 'Sensor technologies for intelligent transportation systems', *Sensors*, 2018, **18**, pp. 1–24

[2] Krizhevsky, A., Sutskever, I., Hinton, G.: 'Imagenet classification with deep convolutional neural networks'. Advances in Neural Information Processing Systems, Lake Tahoe, NV, USA., 2012, pp. 1097–1105

[3] Szegedy, C., Liu, W., Jia, Y.*, et al.*: 'Going deeper with convolutions'. 2015 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA., June 2015, pp. 1–9

[4] Simonyan, K., Zisserman, A.: 'Very deep convolutional networks for large-scale image recognition'. Int. Conf. on Learning Representations, San Diego, CA, USA., 2015

[5] Deng, J., Dong, W., Socher, R.*, et al.*: 'Imagenet: a large-scale hierarchical image database'. 2009 IEEE Conf. on Computer Vision and Pattern Recognition, Miama Beach, FL, USA., June 2009, pp. 248–255

[6] LeCun, Y., Bengio, Y.: 'The handbook of brain theory and neural networks', in '*Convolutional networks for images, speech, and time series*' (MIT Press, Cambridge, MA, USA, 1998), pp. 255–258

[7] Huang, X., Kwiatkowska, M., Wang, S.*, et al.*: 'Safety verification of deep neural networks'. Int. Conf. on Computer Aided Verification, Heidelberg, Germany, 2017, pp. 3–29

[8] Gal, Y., Ghahramani, Z.: 'Dropout as a Bayesian approximation: representing model uncertainty in deep learning'. Int. Conf. on Machine Learning, San Juan, Puerto Rico, 2016, pp. 1050–1059

[9] Nguyen, A., Yosinski, J., Clune, J.: 'Deep neural networks are easily fooled: high confidence predictions for unrecognizable images'. 2015 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA., June 2015, pp. 427–436

[10] Geiger, A., Lenz, P., Urtasun, R.: 'Are we ready for autonomous driving? The kitti vision benchmark suite'. 2012 IEEE Conf. on Computer Vision and Pattern Recognition, Providence, RI, USA., June 2012, pp. 3354–3361

[11] Phippen, D., Daniel, L., Gashinova, M.*, et al.*: 'Trilateralisation of targets using a 300 ghz radar system'. Int. Conf. on Radar Systems, Belfast, UK., 2017

[12] Daniel, L., Stove, A., Hoare, E.*, et al.*: 'Application of Doppler beam sharpening for azimuth refinement in prospective low-tHz automotive radars', *IET Radar Sonar Navig.*, 2018, **12**, (10), pp. 1121–1130

[13] Wöhler, C., Schumann, O., Hahn, M.*, et al.*: 'Comparison of random forest and long short-term memory network performances in classification tasks using radar'. Sensor Data Fusion: Trends, Solutions, Applications (SDF), Bonn, Germany, 2017, pp. 1–6

[14] Schumann, O., Hahn, M., Dickmann, J.*, et al.*: 'Semantic segmentation on radar point clouds'. 2018 21st Int. Conf. on Information Fusion (FUSION), Cambridge, UK., 2018, pp. 2179–2186

[15] Rohling, H., Heuel, S., Ritter, H.: 'Pedestrian detection procedure integrated into a 24 GHz automotive radar'. 2010 IEEE Radar Conf., Washington, DC, USA., May 2010, pp. 1229–1232

[16] Major, B., Fontijne, D., Ansari, A.*, et al.*: 'Vehicle detection with automotive radar using deep learning on range-azimuth-Doppler tensors'. The IEEE Int. Conf. on Computer Vision (ICCV) Workshops, Seoul, Korea, October 2019

[17] Bartsch, A, Fitzek, F, Rasshofer, R.H.: 'Pedestrian recognition using automotive radar sensors', *Adv. Radio Sci.: ARS*, 2012, **10**, pp. 45–55

[18] Angelov, A., Robertson, A., Murray-Smith, R.*, et al.*: 'Practical classification of different moving targets using automotive radar and deep neural networks', *IET Radar Sonar Navig.*, 2018, **12**, (10), pp. 1082–1089

[19] Lombacher, J., Hahn, M., Dickmann, J.*, et al.*: 'Potential of radar for static object classification using deep learning methods'. 2016 IEEE MTT-S Int. Conf. on Microwaves for Intelligent Mobility (ICMIM), San Diego, CA, USA., 2016, pp. 1–4

[20] Texas Instruments: 'Short range radar reference design using awr1642'. Technical report, April 2017

[21] Phippen, D., Daniel, L., Hoare, E., *et al.*: '3d images of pedestrians at 300 GHz'. 2019 20th Int. Radar Symp. (IRS), Bonn, Germany, 2019, pp. 1–10
[22] Glorot, X., Bengio, Y.: 'Understanding the difficulty of training deep feedforward neural networks'. Proc. of the Thirteenth Int. Conf. on Artificial Intelligence and Statistics, Sardinia, Italy, 2010, pp. 249–256
[23] Chen, S., Wang, H., Xu, F., *et al.*: 'Target classification using the deep convolutional networks for sar images', *IEEE Trans. Geosci. Remote Sens.*, 2016, **54**, (8), pp. 4806–4817
[24] Goodfellow, I., Bengio, Y., Courville, A.: '*Deep learning*' (The MIT Press, USA., 2016)
[25] Yosinski, J., Clune, J., Bengio, Y., *et al.*: 'How transferable are features in deep neural networks?'. Advances in Neural Information Processing Systems, Montreal, Canada, 2014, pp. 3320–3328
[26] Van der Maaten, L., Hinton, G.: 'Visualizing data using t-sne', *J. Mach. Learn. Res.*, 2008, **9**, pp. 2579–2605
[27] Richards, M.A.: '*Fundamentals of radar signal processing*' (McGraw-Hill Education (India) Pvt Limited, USA., 2005)
[28] Dollar, P., Wojek, C., Schiele, B., *et al.*: 'Pedestrian detection: an evaluation of the state of the art', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2012, **34**, (4), pp. 743–761
[29] Ren, S., He, K., Girshick, R., *et al.*: 'Faster r-cnn: towards real-time object detection with region proposal networks', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2017, **39**, (6), pp. 1137–1149
[30] Liu, W., Anguelov, D., Erhan, D., *et al.*: 'Ssd: single shot multibox detector'. Computer Vision – ECCV 2016, Cham, 2016, pp. 21–37
[31] Lin, T., Goyal, P., Girshick, R., *et al.*: 'Focal loss for dense object detection'. 2017 IEEE Int. Conf. on Computer Vision (ICCV), Venice, Italy, October 2017, pp. 2999–3007
[32] Redmon, J., Divvala, S., Girshick, R., *et al.*: 'You only look once: unified, real-time object detection'. 2016 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA., June 2016, pp. 779–788
[33] Girshick, R., Donahue, J., Darrell, T., *et al.*: 'Rich feature hierarchies for accurate object detection and semantic segmentation'. 2014 IEEE Conf. on Computer Vision and Pattern Recognition, Columbus, OH, USA., June 2014, pp. 580–587
[34] Girshick, R.: 'Fast r-cnn'. 2015 IEEE Int. Conf. on Computer Vision (ICCV), Santiago, Chile, December 2015, pp. 1440–1448
[35] Ester, M., Kriegel, H., Sander, J., *et al.*: 'A density-based algorithm for discovering clusters a density-based algorithm for discovering clusters in large spatial databases with noise'. Proc. of the Second Int. Conf. on Knowledge Discovery and Data Mining, KDD'96, Portland, OR, USA., 1996, pp. 226–231
[36] Marchetti, E., Du, R., Willetts, B., *et al.*: 'Radar cross-section of pedestrians in the low-tHz band', *IET Radar Sonar Navig.*, 2018, **12**, (10), pp. 1104–1113
[37] Everingham, M., Van Gool, L., Williams, C., *et al.*: 'The pascal visual object classes (voc) challenge', *Int. J. Comput. Vis.*, 2010, **88**, (2), pp. 303–338