DTU Library

Improving SAR Automatic Target Recognition Models with Transfer Learning from Simulated Data

Malmgren-Hansen, David; Kusk, Anders; Dall, Jørgen; Nielsen, Allan Aasbjerg; Engholm, Rasmus ; Skriver, Henning

Improving SAR Automatic Target Recognition Models with Transfer Learning from Simulated Data

David Malmgren-Hansen, Anders Kusk, Jørgen Dall, Allan Aasbjerg Nielsen, Rasmus Engholm,
and Henning Skriver

Abstract—Data driven classification algorithms have proven to do well for Automatic Target Recognition (ATR) in Synthetic Aperture Radar (SAR) data. Collecting datasets suitable for these algorithms is a challenge in itself as it is difficult and expensive. Due to the lack of labelled datasets with real SAR images of sufficient size, simulated data plays a big role in SAR ATR development, but the transferability of knowledge learned on simulated data to real data remains to be studied further.

In this paper we show the first study of Transfer Learning between a simulated dataset and a set of real SAR images. The simulated dataset is obtained by adding a simulated object radar reflectivity to a terrain model of individual point scatters, prior to focusing. Our results show that a Convolutional Neural Network (Convnet) pre-trained on simulated data has a great advantage over a Convnet trained only on real data, especially when real data is sparse. The advantages of pre-training the models on simulated data show both in terms of faster convergence during the training phase and on the end accuracy when benchmarked on the MSTAR dataset. These results encourage SAR ATR development to continue the improvement of simulated datasets of greater size and complex scenarios in order to build robust algorithms for real life SAR ATR applications.

Index Terms—SAR ATR, Convolutional Neural Networks, Transfer Learning, SAR Image Simulation.

I. Introduction

In Automatic Target Recognition (ATR) for Synthetic Aperture Radar (SAR) applications it is well known that lack of realistic and big labelled datasets is a challenge for the development of robust algorithms. For statistical studies of SAR ATR algorithm performance, it is important to have sufficient data. However, just as important is it to have a great variety of realistic scenarios. The latter should cover objects on different kind of backgrounds, (e.g. grass, road, gravel) but also scenes where objects are closely co-located, occluded by trees and similar challenging scenarios, dependent on its relevance to the ATR application at hand. Such datasets will be difficult or at least very expensive to collect for each individual SAR ATR application. Hence simulated data has great potential to improve this field.

The Moving and Stationary Target Acquisition and Recognition (MSTAR) dataset has been used for benchmarking algorithms since it was collected by AFRL[1] and DARPA[2] between 1995-1997. The vehicles in the dataset are densely sampled in azimuthal view angle but are only recorded at few depression angles and the backgrounds remain stationary throughout the dataset. Additionally, vehicles are centered in the image, and there are no confusing features in the background (trees, other vehicles, etc.). Due to these shortcomings it should not be considered as a random subset of operational interesting SAR ATR data [1]. Despite this, MSTAR is interesting since it can show an algorithm's robustness to the statistical properties of real SAR data. MSTAR can also reveal the algorithm's generalizability of features learned on one depression angle, to another, and thereby whether $2°$ difference in depression angle sampling is adequate in ATR development. Several data driven classification models have proven to do well on the MSTAR task, such as [2], [3], [4], [5], [6], [7]. In order to learn more about the algorithms' scalability to larger recognition tasks with more targets, or complex scenarios, the potential of simulated data is big.

Research on improving Convolutional Neural Networks (Convnets) by including simulated data dates back to 2004 [8] for natural images. Here training images of toy figures made the Convnet generalize to predictions on real images, when complex backgrounds were inserted in the training samples. In more recent work, [9], [10] it is suggested to use rendered images from 3D CAD models to help training Convnets in specific applications where labelled data is sparse. In [9] there is a specific focus on making the rendered images realistic. Instead of rendering complex backgrounds it is suggested to use real backgrounds and add rendered objects to them. Following such an approach for SAR data to enhance SAR ATR development is suggested several times in the literature, [11], [12], [13], but whereas realistic rendering is highly developed in the optical image domain, SAR simulation needs to deal with different challenges. These challenges include complex summation of reflections due to a coherent signal source, and edge diffraction due to the large wavelength compared to visible light. Several approaches to SAR image simulation have been suggested from very simple scatter models as the one used in [14] to sophisticated models that analyze the object and scene for geometric features that have significant scattering [15].

To open up for the use of simulated data in real SAR ATR applications, methods of exploiting it need to be explored further. This is both in terms of ensuring that the simulated

David Malmgren-Hansen and Allan Aasbjerg Nielsen are with the Department of Applied Mathematics and Computer Science at the Technical University of Denmark. e-mail: (see http://people.compute.dtu.dk/dmal/ or http://people.compute.dtu.dk/alan/)

Anders Kusk, Jørgen Dall and Henning Skriver are with National Space Institute at the Technical university of Denmark.

Rasmus Engholm is Senior Data Scientist at Terma A/S, Lystrup Denmark

[1]Air Force Research Laboratory
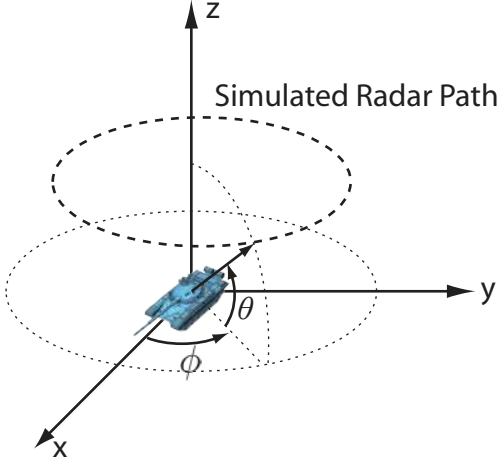[2]Defense Advanced Research Projects Agency

Fig. 1. Coordinate system used for simulated SAR images.

data is sufficiently realistic and to find practical ways of incorporating it into data driven classification algorithms. In [16] the combined use of simulated data and real data is explored in a study of pairwise binary classification tasks between SAR images with eight different ships. The exact ship models are carefully modelled in a CAD drawing and rendered with a SAR simulation tool. This study is not close to a real application in the sense that it only shows the separability between the pairs of ships, but reach an important conclusion of simulated data having a regularizing effect on Convnet training.

In this article we aim to explore the information that can be learned from a simulated SAR dataset which can be transferred to a real SAR data scenario. We do this by studying the accuracy of a Convnet pre-trained on the simulated data versus a Convnet trained with random initialized weights, i.e. Transfer Learning such as in [17]. With this approach objects in our simulated dataset, $\mathcal{D}_{sim}$, do not have to be geometric replicas of the real objects in our target dataset, $\mathcal{D}_{real}$, as long as they share image features. This is a great advantage for SAR ATR where exact target geometry rarely is known. In this way our method possesses both a practical way of incorporating simulated data into a SAR ATR application as it reveals whether the data is realistic enough to benefit a classification algorithm. We show that generic features learned on the simulated data benefit the model performance on real data. Our results show that training time can be reduced and end performance improved. In the experiments described in Section III we used simulated data from the tool introduced in [18] named SARSIM. This tool is described in Section II and the specific dataset used in this work will be publicly available for research purposes at [19]. The results from the Transfer Learning experiments are shown in Section IV and Section V concludes and summarizes our findings.

II. DATA SIMULATION

To simulate realistic SAR images of an object, a model of the object radar reflectivity is required, as well as models of the surrounding terrain and the radar system. The object reflectivity is estimated using the commercially available CST[3] Microwave Studio Asymptotic Solver with a 3D CAD model as input. CST estimates the complex scattered electric field components over a range of frequencies, elevation (θ) angles and azimuth (ϕ) angles, using the geometry illustrated in Figure 1. The frequency sweep covers the simulated radar system bandwidth, and a full 360° sweep in azimuth is carried out for each desired elevation angle. Since the CST simulation results are sampled in frequency, the sampling increment, Δf, determines the alias-free scene extent (object size) in range, W_r, that can be simulated by $\Delta f < \frac{c}{2W_r}$, c being the speed of light. Likewise, the azimuth sampling increment, $\Delta \phi$ is determined not by the antenna beamwidth (as it is in a real system), but by the alias-free scene extent in azimuth, W_x, as $\Delta \phi < \frac{\lambda}{2W_x}$, with λ being the wavelength.

The geometry in Figure 1 assumes acquisition along a circular synthetic aperture. By using a time-domain back-projection focusing algorithm [20], however, the final output images can be provided on an arbitrary grid, e.g. a rectangular slant-range/azimuth grid. Compared to a linear aperture, simulation of a circular aperture yields a slightly different result due to varying incidence angles along the aperture. For an aperture corresponding to 10 cm resolution at X-band and nominal $\theta = 45°$, the difference is smaller than 0.2°. This difference is independent of sensor altitude, and is smaller yet for lower depression angles. This means that even though linear aperture SAR images are desired, the circular aperture reflectivity simulation is likely a reasonable approximation. The output resolution is determined by the frequency bandwidth and azimuth angle interval used.

Terrain clutter is modelled by adding the object signal to the simulated backscatter of a set of individual point-like scatterers prior to SAR focusing. The scatterers are specified by (X, Y, Z)-position and complex reflectivity. For every aperture angle, the visibility of each scatterer is determined by casting rays from the sensor position to the scatterer and testing for intersection with the object CAD model. Scatterers for which the ray intersects the object are suppressed. For every visible scatterer, the range is calculated, and the simulated backscatter signal is generated by multiplying a flat spectrum with (a) the complex reflectivity of the scatterer, (b) a linear phase ramp that shifts the time response to have maximum at the calculated range, and (c) the two-way propagation phase. The summed contributions of all visible scatterers are added to the object signal. Simulated thermal noise is then added to every sample based on the radar equation and the simulated SAR system parameters. Finally, the signal is focused in range by an inverse FFT (with Taylor weighting for sidelobe suppression), and azimuth focusing is done using the back-projection algorithm. The approach above ensures that the clutter and noise signals are subjected to the same SAR processing as the object signal, and also models shadowing and partial visibility along the aperture correctly. Arbitrary scatterers can be used for the clutter simulation; currently, homogeneous clutter is simulated by a point scatterer at every

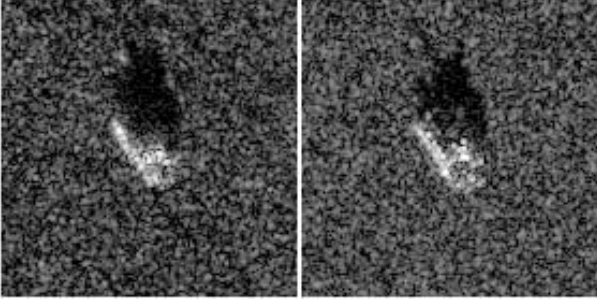[3]CST - Computer Simulation Technology, Dassault Systemes

Fig. 2. Real vs. simulated images of T-72 at 15° depression and 148° azimuth. Left MSTAR actual SAR image, Right SARSIM simulated image.

Fig. 3. MSTAR T-72 tank and T-72 CAD model used for simulated image.

output grid point, with complex reflectivity modelled by a complex zero-mean Gaussian process with identical real and imaginary part variance s^2:

$$s^2 = (\sigma^0 \delta_a \delta_r)/2 \cos \theta \quad (1)$$

where σ^0 is the terrain reflectivity, and δ_a and δ_r are the range and azimuth pixel spacings. Typical values of σ^0 for different terrain types are available in [21].

Figure 2 shows a real MSTAR SAR image of a T-72 tank and the corresponding simulated image generated with the SARSIM tool, using a T-72 CAD model and MSTAR radar parameters, assuming a grassy terrain. A photo of the tank, as well as a rendering of the CAD model are shown in Figure 3.

III. EXPERIMENTS

The experiments explained in this section are intended to investigate the transferability of information learned in a parametric model. Let us consider a classification task on a dataset, $\mathcal{D}$, solved with a Convnet model, $y(\mathbf{x}, \mathbf{w})$, with input $\mathbf{x} \in \mathcal{D}$. The model parameters or weights, $\mathbf{w}$, are initialized following the approach in [22] with a uniform distribution in the interval,

$$\mathbf{w} \sim U[-\frac{\sqrt{6}}{\sqrt{n_l + n_{l+1}}}, \frac{\sqrt{6}}{\sqrt{n_l + n_{l+1}}}],$$
$$for \; l = 0, ..., L - 1 \quad (2)$$

with L being the number of layers in our model, and n_l being the number of incoming nodes in the l'th layer. For a multi class classification task, the maximum likelihood optimization of our model will be given with the categorical cross entropy

error function if the output of the network is a softmax function, [23]. The error function is given by,

$$E(\mathbf{w}) = - \sum_{n=1}^{N} \mathbf{t}_n \ln(y(\mathbf{x}_n, \mathbf{w})), \quad (3)$$

with $\mathbf{t}_n$ being a "one-hot" vector encoding a one for the correct class and zeros otherwise given the n'th image. $y()$ has an output representing the probability of each k-classes in the given classification task. The error function can then be minimized with a stochastic gradient descent (SGD) scheme. In our analysis we use a version of SGD where the gradient for each iteration is divided with a root-mean-square weighted sum of the weights and previous gradients. This method called RMSprop [24], helps ensure a stable learning curve by effectively adjusting the step size of the weight updates.

In order to explain the amount of information that can be transferred from a simulated dataset, $\mathcal{D}_{sim}$, to a real dataset, $\mathcal{D}_{real}$, we investigate the error, $E(\mathbf{w})$, during the minimization process of Equation (3) with $\mathbf{x}_n \in \mathcal{D}_{real}$, for the following two scenarios.

1) $\mathbf{w}$ being initialized according to Equation (2).
2) $\mathbf{w}$ being initialized from $\underset{w}{\operatorname{argmin}} E(\mathbf{w}, \mathbf{x}), \; for \; \mathbf{x} \in \mathcal{D}_{sim}$.

Additionally, we can consider random subsets of $\mathcal{D}_{real}$ in order to study the relationship with dataset sparsity. The amount of data needed for training a model to SAR ATR is critical in operational scenarios, since it can be difficult to obtain dense datasets of the relevant objects. It should be emphasized that in our setup the classification task on $\mathcal{D}_{sim}$ and $\mathcal{D}_{real}$ can differ in terms of objects and complexity. Since the number of classes in $\mathcal{D}_{sim}$ and $\mathcal{D}_{real}$ in our experiment differ, we re-initialize the last layer with random weights according to Equation (2) in both scenarios. The two datasets used in the experiments are described in Section III-A.

The Convnet used in our experiments has 412,028 parameters. Layers and input sizes of the Convnet can be seen in Table I. We run our optimization by mini batch gradient descent with a batch size of 128 images. We further add L_2-norm weight decay to our update scheme.

Several approaches to Transfer Learning are reported in literature where only the last layer is fine tuned on a new dataset or where early layers are fine tuned with reduced learning rate. We will fine tune all parameters in the Convnet with equal learning rate as this was experimentally shown as the best procedure in [17].

On the standard MSTAR 10-way classification task with randomly initialized weights the model presented in Table I reaches 93.2% accuracy on the test set.

A. Datasets

The two datasets used in the experiments are described below.

1) MSTAR: MSTAR is a set of real SAR images of military vehicles. The subset often used for testing classification algorithms contains 10 vehicle types recorded at 15° and

TABLE I
CONVNET MODEL USED IN EXPERIMENTS. THE OUTPUT SIZE DENOTES
(number of rows x number of columns x number of nodes) IN EACH LAYER.
RECTIFIED LINEAR UNIT IS AN ACTIVATION FUNCTION DEFINED BY
$f(x) = max(0, x).$

Layer Type	Layer Output size	Kernel size	comment
Input	128x128x1	-	-
Convolutional	128x128x12	5x5	ReLU activation
Maxpooling	42x42x12	3x3	
Convolutional	42x42x36	5x5	ReLU activation
Maxpooling	21x21x36	2x2	
Convolutional	17x17x72	5x5	ReLU activation
Maxpooling	8x8x72	2x2	
Fully connected	56	1x4608	ReLU activation
Fully connected	10	1x56	Softmax activation

17° depression angles. The vehicles are densely sampled in azimuthal rotation with an image for each 1° - 2° depending on the object. The training set contains the ten vehicles at 17° and the test set contains the 15° samples.

2) SARSIM: The dataset consists of simulated SAR images of vehicles and is simulated according to the description in Section II. Fourteen vehicle CAD models have been downloaded from the web and each one is simulated for every two degree azimuthal object rotation at seven different depression angles (15°, 17°, 25°, 30°, 35°, 40° and 45°). The objects belong to seven different categories with two objects in each category (truck, car, motorbike, bus, tank, bulldozer and pickup). Each object has been simulated with three different statistically generated background clutter types, corresponding to grass, roads and a mean of the two.

In our experiments the dataset is split randomly with 90% for training and 10% for testing. The effect on end performance by splitting the data in different ways (e.g. by depression angle, object model instance etc.) is not in the scope of this article to explore, but could be relevant for future studies. With the random split of 90%/10% we achieve a high accuracy on the test set (>99%). This is probably due to the very dense sampling of objects in different view angles.

IV. RESULTS

For each of the Convnet training scenarios on MSTAR, we consider five random subsets of the full dataset, (20%, 40%, 60%, 80% and 100%). In Figure 4 the accuracy on the test set is shown for each epoch, with epoch being the number of batch iterations corresponding to all images in the dataset being processed.

It is clearly seen that pre-training on simulated data makes the network converge faster on the real data set. This can be a great advantage in operational SAR ATR scenarios where problems can involve more vehicles than present in the MSTAR and where new vehicles are added regularly. Additionally, MSTAR lacks the variance of objects on different backgrounds, which makes real problems require even bigger datasets to fulfil operational requirements.

The difference in the end performance between the randomly initialized model and the pre-trained model seems to diminish with increasing dataset sizes. To illustrate this, Figure 5 shows a bar-plot of the end performance for each experiment.
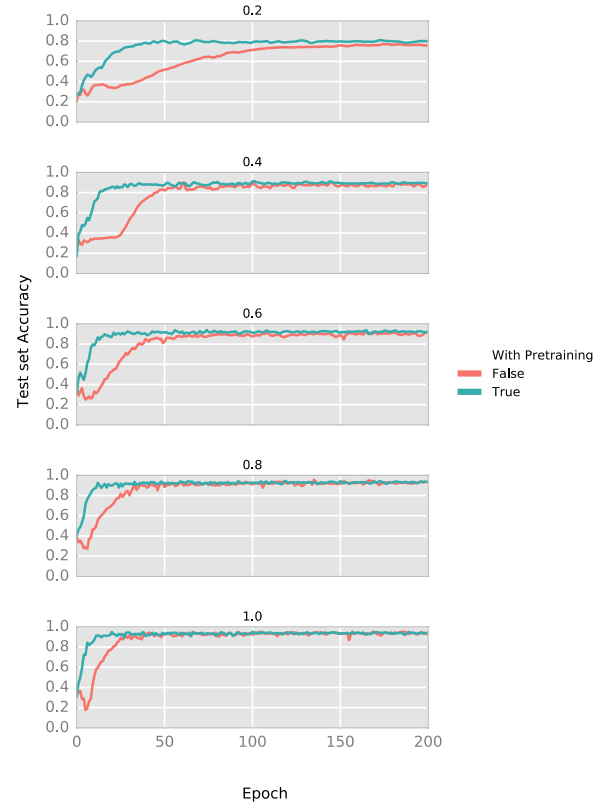

Fig. 4. Error on test dataset measured for each epoch during the training process. Different sub-plots show the fraction of the training data that was used.
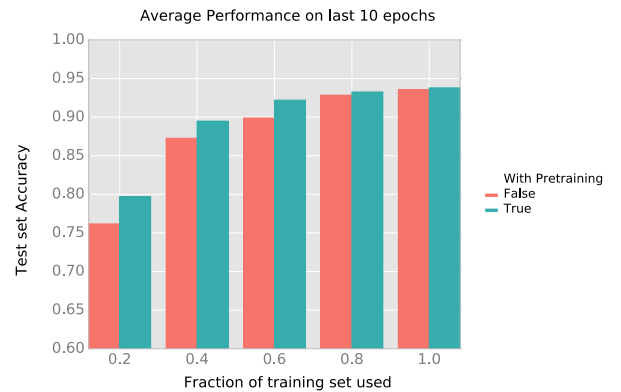

Fig. 5. End test performance averaged over the last 10 epochs to even out test error variance due to training set/test set bias.

The end performance is reported as the average over the last ten epochs in order to reduce the variance on the test accuracy. This variance is likely due to biases between test and training sets and since the MSTAR problem splits between test and training on the depression angle, there is a significant bias between them. This variance was also reported in [2].

The difference in performance with increasing percentage of data included in the training is likely due to the very dense sampling in azimuth angles and lack of background clutter

variance in MSTAR. In operational scenarios it is very unlikely that such a dense sampling can be obtained for all objects of concern. Figure 5 shows that there is a great potential for increased performance with pre-training on simulated data when only sparsely sampled datasets are available.

V. Conclusion

The advances in simulation of SAR images lead towards practical data driven classification algorithms being useful in SAR ATR. We have shown that the simulated SAR images do in fact contain valid information that is useful for real SAR data ATR problems. The simulated data used in this work is improving both training convergence and the end performance of the classifier.

Transfer Learning provides a favourable scheme in utilizing simulated data. It enables the use of the many detailed CAD models freely available online to learn generic features that can be transferred to real SAR images in ATR applications. As detailed CAD models might be hard to obtain of all objects in a given operational classification task, Transfer Learning might be necessary to alleviate the lack of sufficient amounts of real SAR data.

The proposed method can as well be useful for other remote sensing problems, such as land cover classification and detection of objects in cities. In the proposed simulation algorithm it is assumed that object and ground interaction effects are negligible by separately simulating each of them. When comparing with the MSTAR images this assumption seemed reasonable, though it might be different for other radar frequencies and different environments. By proposing Transfer Learning between the simulated and real data it is likely that generic features can be learned although some simplifications are made in the simulation process. A study of which parts of the simulation are important in order to make Convnets generalize between the two data domains is an interesting subject for future work.

For the future improvement of SAR ATR, bigger and more realistic datasets need to be gathered. Including more complex scenes such as co-located vehicles, occlusions by trees and confusing objects such as buildings, will be the next step in the development. The performance of algorithms on a greater variety of object types will also shed light on data driven classification methods' robustness in operational SAR ATR applications. It must be expected that in operationally interesting applications, algorithms should deal with far more vehicle types compared to MSTAR and SARSIM. As shown in our experiments simulated data may play an important role in these studies.

Acknowledgement

References

[1] T. D. Ross, S. W. Worrell, V. J. Velten, J. C. Mossing, and M. L. Bryant, "Standard SAR ATR evaluation experiments using the MSTAR public release data set," in *Aerospace/Defense Sensing and Controls.* International Society for Optics and Photonics, 1998, pp. 566–573.

[2] D. A. E. Morgan, "Deep convolutional neural networks for ATR from SAR imagery," *Proc. SPIE*, vol. 9475, pp. 94 750F–94 750F–13, 2015. [Online]. Available: http://dx.doi.org/10.1117/12.2176558

[3] J. A. O'Sullivan, M. D. DeVore, V. Kedia, and M. I. Miller, "SAR ATR performance using a conditionally gaussian model," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 37, no. 1, pp. 91–108, 2001.

[4] U. Srinivas, V. Monga, and R. G. Raj, "SAR automatic target recognition using discriminative graphical models," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 50, no. 1, pp. 591–606, 2014.

[5] Y. Sun, Z. Liu, S. Todorovic, and J. Li, "Adaptive boosting for SAR automatic target recognition," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 43, no. 1, 2007.

[6] S. Wagner, "Combination of convolutional feature extraction and support vector machines for radar ATR," in *Information Fusion (FUSION), 2014 17th International Conference on.* IEEE, 2014, pp. 1–6.

[7] Q. Zhao and J. C. Principe, "Support vector machines for SAR automatic target recognition," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 37, no. 2, pp. 643–654, 2001.

[8] Y. LeCun, F. J. Huang, and L. Bottou, "Learning methods for generic object recognition with invariance to pose and lighting," in *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, vol. 2. IEEE, 2004, pp. II–104.

[9] X. Peng, B. Sun, K. Ali, and K. Saenko, "Learning deep object detectors from 3d models," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1278–1286.

[10] A. Shrivastava, T. Pfister, O. Tuzel, J. Susskind, W. Wang, and R. Webb, "Learning from simulated and unsupervised images through adversarial training," *arXiv preprint arXiv:1612.07828*, 2016.

[11] K. Tang, X. Sun, H. Sun, and H. Wang, "A geometrical-based simulator for target recognition in high-resolution SAR images," *IEEE Geoscience and Remote Sensing Letters*, vol. 9, no. 5, pp. 958–962, 2012.

[12] T. Balz, H. Hammer, and S. Auer, "Potentials and limitations of SAR image simulators–a comparative study of three simulation approaches," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 101, pp. 102–109, 2015.

[13] R. Sharma and N. S. Subotic, "Construction of hybrid templates from collected and simulated data for SAR ATR algorithms," in *Aerospace/Defense Sensing and Controls.* International Society for Optics and Photonics, 1998, pp. 480–486.

[14] D. Malmgren-Hansen, R. Engholm, and M. O. Pedersen, "Training convolutional neural networks for translational invariance on SAR ATR," in *EUSAR 2016: 11th European Conference on Synthetic Aperture Radar, Proceedings of.* VDE, 2016, pp. 1–4.

[15] N. Ødegaard, A. Knapskog, C. Cochin, and B. Delahaye, "Comparison of real and simulated SAR imagery of ships for use in ATR," in *SPIE Defense, Security, and Sensing.* International Society for Optics and Photonics, 2010, pp. 76 990O–76 990O.

[16] N. Ødegaard, A. O. Knapskog, C. Cochin, and J.-C. Louvigne, "Classification of ships using real and simulated data in a convolutional neural network," in *Radar Conference (RadarConf), 2016 IEEE.* IEEE, 2016, pp. 1–6.

[17] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" in *Advances in neural information processing systems*, 2014, pp. 3320–3328.

[18] A. Kusk, A. Abulaitijiang, and J. Dall, "Synthetic SAR image generation using sensor, terrain and target models," in *EUSAR 2016: 11th European Conference on Synthetic Aperture Radar, Proceedings of.* VDE, 2016, pp. 1–5.

[19] A. Kusk, J. Dall, and H. Skriver, "Simulated SAR data of vehicles on a background [data set]." http://doi.org/10.5281/zenodo.573750, 2017.

[20] Ulander, L.M.H. and Hellsten, H. and Stenstrom, G., "Synthetic-aperture radar processing using fast factorized back-projection," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 39, no. 3, pp. 760–776, 2003.

[21] F. Ulaby and M. Dobson, *Handbook of radar scattering statistics for terrain*, ser. The Artech House remote sensing library. Artech House, 1989.

[22] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks." in *Aistats*, vol. 9, 2010, pp. 249–256.

[23] C. Bishop, *Pattern Recognition and Machine Learning*, 8th ed., ser. Princeton Legacy Library. Springer, 2006.

[24] G. Hinton, N. Srivastava, and K. Swersky, "Lecture 6a overview of mini–batch gradient descent," *Coursera Lecture slides https://class. coursera. org/neuralnets-2012-001/lecture,[Online,* 2012.