Journal of Applied Remote Sensing, Volume 12, Issue 4, Article number 046025, Oct-Dec. 2018
DOI: 10.1117/1.JRS.12.046025

Target Recognition for Synthetic Aperture Radar Imagery Based on Convolutional Neural Network Feature Fusion

O. Kechagias-Stamatis [a*]
[a]Cranfield Defence and Security, Centre for Electronic Warfare Information and Cyber, Shrivenham, SN6 8LA, UK

Abstract. Driven by the great success of deep Convolutional Neural Networks (CNNs) that are currently used by quite a few computer vision applications, we extend the usability of visual based CNNs into the Synthetic Aperture Radar (SAR) data domain without employing transfer learning. Our SAR Automatic Target Recognition (ATR) architecture efficiently extends the pre-trained Visual Geometry Group CNN from the visual domain into the X-band SAR data domain by clustering its neuron layers, bridging the visual – SAR modality gap by fusing the features extracted from the hidden layers and by employing a local feature matching scheme. Trials on the moving and stationary target acquisition dataset under various setups and nuisances demonstrate a highly appealing ATR performance gaining 100% and 99.79% in the 3-class and 10-class ATR problem respectively. We also confirm the validity, robustness and conceptual coherence of the proposed method by extending it to several state-of-the-art CNNs and commonly used local feature similarity/ match metrics.

Keywords: Automatic Target Recognition; Convolutional Neural Networks; Deep Learning; Synthetic Aperture Radar

***E-mail: o.kechagiasstamatis@cranfield.ac.uk

1 Introduction

Automatic target recognition (ATR) for military applications is an active research topic that seeks further reducing collateral damage and fratricide targeting. Investigations involve solutions based on numerous spatial, i.e. 2D/ 3D and data domains, such as 2D infrared (IR) [1–5], 2D Synthetic Aperture Radar (SAR) [6–21], 2D Inverse SAR (ISAR) [22] and 3D Light Detection and Ranging (LIDAR) [23–27], with each of these data modalities having its own strengths and weaknesses. For example, state-of-the-art local feature (data) descriptors from the visual domain have already proven their capabilities in the IR domain, but IR suffers from the time of day and the target's history [28]. LIDAR involves 3D data manipulation with numerous advantages such as invariance to illumination variation and invariance to target pose changes [25]. Despite these advantages, the processing burden implied by 3D data processing is much higher compared to the 2D data domain. Regarding SAR imagery, its main advantages are the 2D data structure that affords computational

efficiency, being invariant to the target's history and the all-weather night-and-day data acquisition capability that extends considerably the operational capabilities on the battlefield.

SAR ATR has been attempted using various techniques. For example, feature based solutions encode the SAR image by a set of attributes that are sufficiently descriptive to achieve target classification under various nuisances. Current literature includes extracting features based on Krawtchouk moments [29] derived from the discrete-defined Krawtchouk polynomials, using biologically inspired features such as episodic and semantic features [30] or sparse robust filters [31] originating from the human cognition process. Further methods include binary operations [32], utilizing the target's scattering centers [17,33] and fusing the azimuth and range target profiles [34]. Stacked Auto-encoder (SA) type of SAR ATR solutions rely on features that are extracted from the SAR imagery and are input to an SA type neural network. The latter adopts an unsupervised learning strategy used in neural networks that can convert the input data into abstract expressions utilizing a nonlinear model. For the current SA type solutions, SAR ATR oriented literature suggests either exploiting Local Binary Features [20] or modifying the reconstruction error of the typical Auto-encoder scheme by adding a Euclidean distance restriction for the neural network hidden layer features [19]. Compressive Sensing (CS) based SAR ATR approaches aim at recovering a signal that has been remapped from the originating domain to a domain where the signal is sparse, using a non-adaptive linear projection. Signal recovery is achieved via an l_1-*norm* optimization process. In the context of SAR ATR, Multitask CS [35] exploits the statistical correlation among multiple target views to recover the target's signature that is then used for target recognition under a compressive sensing scheme. Bayesian CS [16] relies on the scattering centers of the SAR image that are used as an input signal to the CS technique. Sparse Representation Classification (SRC) type of solutions aims at recovering the testing imagery out of a dictionary

where the training images are the dictionary's base elements. SRC aims at identifying the sparsest representation of the testing imagery within the dictionary by employing an l_1-*norm* optimization scheme. The final classification decision matches the class that provides the smallest residual error. In the context of SAR ATR, Joint SRC [36] exploits three target views to increase the completeness of the target's SAR signature and a mixed $l_0 \backslash l_2$-*norm* for the optimization. The reasoning behind the multiple views is that these are highly correlated sharing the same response pattern within the dictionary and thus this conciseness can enhance the overall ATR performance. $L_{1/2}$-NMF [21] uses the $l_{1/2}$-*norm* optimization to identify the sparsest solution. The features used as input to the SRC technique are the result of a non-negative matrix factorization process applied on the SAR imagery. Dong *et al.* in [11] exploit the monogenic signal of a SAR image as an input to the SRC process. This signal comprises of the 2D SAR image signal and its Riesz transformed representation. Deep Convolutional Neural Networks have also been suggested for SAR ATR. Literature suggests several Convolutional Neural Networks (CNN) based solutions that rely on data specific handcrafted structures [6,8,14,15]. A common feature of these CNN architectures is the relatively small number of hidden layers, which opposes to the multilayered mainstream CNNs used in the visual domain, i.e. AlexNet [37], Visual Geometry Group (VGG) [38], GoogleNet [39] and ResNet [40]. This is because visual images have a higher information content per pixel compared to the radar reflections within a SAR image.

Current mainstream CNNs have an unarguable classification capability in the visual domain. A typical way to deviate the classification capabilities of these CNNs from the training data domain to a different dataset and data domain is by exploiting the Transfer Learning technique [41]. Nevertheless, the combination of completely different data modalities, i.e. SAR and visual imagery, along with the lack of SAR training samples impose a huge constrain to steer these CNNs

to operate with SAR data. Therefore, current CNNs operating on a multi-modal data scheme offer moderate classification performance [42]. A solution to overcome the lack of SAR training samples is to populate the SAR training images via data augmentation. However, this is a time consuming process and most importantly is a *try and evaluate* process as the size and the manner to augment the training data is not known a priori.

Driven by the object classification performance of the pre-trained in the visual domain CNNs, this work proposes a multi-modal and multi-discipline architecture that combines the advantages of CNN and local feature matching. Specifically, the suggested method aims to transfer the already proven classification capability of the VGG-16 [38] from the visual domain to the X-band SAR without involving transfer learning. This operation is not straightforward as directly activating VGG with data of a different modality, i.e. SAR imagery, is a suboptimal solution. Therefore, we bridge the data modality gap by pre-processing the SAR imagery and clustering the VGG's hidden layers into feature-specific based groups. Then a number of clusters are activated and the multi-dimensional responses of the deepest activated layer are transformed and fused into a 1D-feature vector. Finally, the scene and the template feature vectors are input to a local feature-matching scheme that relies on the *Cosine* similarity measure.

The innovations and contributions of this paper can be summarized as:

a. We extend the usability of the VGG CNN from the visual domain to the SAR by introducing a hidden layer-clustering technique. This strategy extends the usability of the mainstream state-of-the-art VGG network that is trained in the visual domain, to a completely different data modality, the SAR domain.

b. We demonstrate that it is feasible to steer a CNN towards a different data modality without employing the transfer learning technique or data augmentation, and thus avoid their disadvantages.

c. We highlight the importance of the ATR classification method by comparing the effectiveness of the *Cosine similarity* measure over several similarity/ match metrics.

d. We extend our architecture to several mainstream state-of-the-art CNNs and validate the conceptual coherence of our technique by presenting high quality ATR performance.

e. We demonstrate that our ATR architecture presents the highest to date SAR ATR capability on the moving and stationary target (MSTAR) acquisition dataset.

The rest of the paper is organized as follows: Section 2 presents the proposed ATR architecture. Section 3 evaluates our pipeline under various setups and nuisances, and extends this concept to various mainstream CNN's. Finally, Section 4 concludes this paper.

2. Proposed Architecture

The suggested clustered VGG-16 SAR ATR architecture is presented in Fig. 1 and is analyzed in the following sections.

2.1 Clustered Convolutional Neural Network

VGG-16 is a multi-layered CNN that encodes the scene and template features from the visual imagery that vary from low-level corners and blobs, in the initial layers, up to high-level data specific features in the last layers. For completeness, the scene and template features are characteristic local patches that ideally should describe the scene and template images respectively in a unique manner, and are robust to geometric transformations and to nuisance factors. Although VGG is powerful, it has been trained on RGB images that are fundamentally different from SAR

imagery. In fact, VGG is trained on RGB color bands while SAR imagery contains radar reflections. Therefore, directly applying on VGG SAR imagery is not an optimum solution. To bridge this modality gap, we exploit the descriptiveness of VGG's hidden layers by dividing them into six groups of layers, i.e. clusters l of varying feature description capability and introduce the clustered VGG (C-VGG) presented in Table 1. The latter table shows, for example, that C-VGG cluster one contains VGG's layers one to five. Notation l refers to the cluster layer activated with $l \in \{1,2,3,4,5,6\}$. This means, for instance, that $l = 2$ activates up to C-VGG's clustered layer 2 while the remaining layers $\{3,4,5,6\}$ are discarded. During the algorithm's tuning process, the optimum activation layer is selected which is then fixed for the experimental evaluation. The features of that layer are then linked to the corresponding template labels used to activate the C-VGG. The optimum layer selection is presented in Section 3.2. In this work C-VGG uses the same parameters (stride, padding and convolutional filter sizes) as in the original implementation [38]. It should be noted that the suggested C-VGG is an extension of the current VGG architecture and aims at exploiting the hidden layers of the pre-trained in the visual domain VGG for ATR tasks in the SAR imagery domain, where originally VGG was not trained.

Table 1 C-VGG layers

C-VGG cluster ID (*l*)	original VGG layers included	Operations involved
1	1-2-3-4-5	Image input – Convolution – ReLU – Normalization – Max pooling
2	6-7-8-9	Convolution – ReLU – Normalization – Max pooling
3	10-11	Convolution – ReLU
4	12-13	Convolution – ReLU
5	14-15-16	Convolution – ReLU – Max pooling
6	17 - end	remaining layers (not exploited)

The reason behind suggesting the specific layer grouping/ clustering strategy for C-VGG is directly related to the position of the convolutional layers of VGG-16. Hence, we cluster VGG's hidden layers so that the first layer of each cluster is a convolutional layer. This strategy affords

controlling the complexity of the features that are extracted from each cluster, because the deeper the convolutional layer, the more complex and data specific to the extracted features are. Thus, exploiting for a SAR ATR application the deep features of VGG, e.g. layer 14 of the original VGG, which corresponds to layer five for C-VGG, is not an optimum choice because the features of that layer are heavily established for visual imagery and not for SAR. In this work we demonstrate that the shallow layers of C-VGG/ VGG that extract generic features are more appropriate for SAR ATR, despite these features being originally established for the visual domain (during the VGG's original training on the ImageNet dataset). It should be clarified that C-VGG is not re-trained in the SAR domain and the capacity of the C-VGG is not restricted in the SAR imagery domain, but only in the visual domain where VGG was trained in, which is not the scope of this paper.

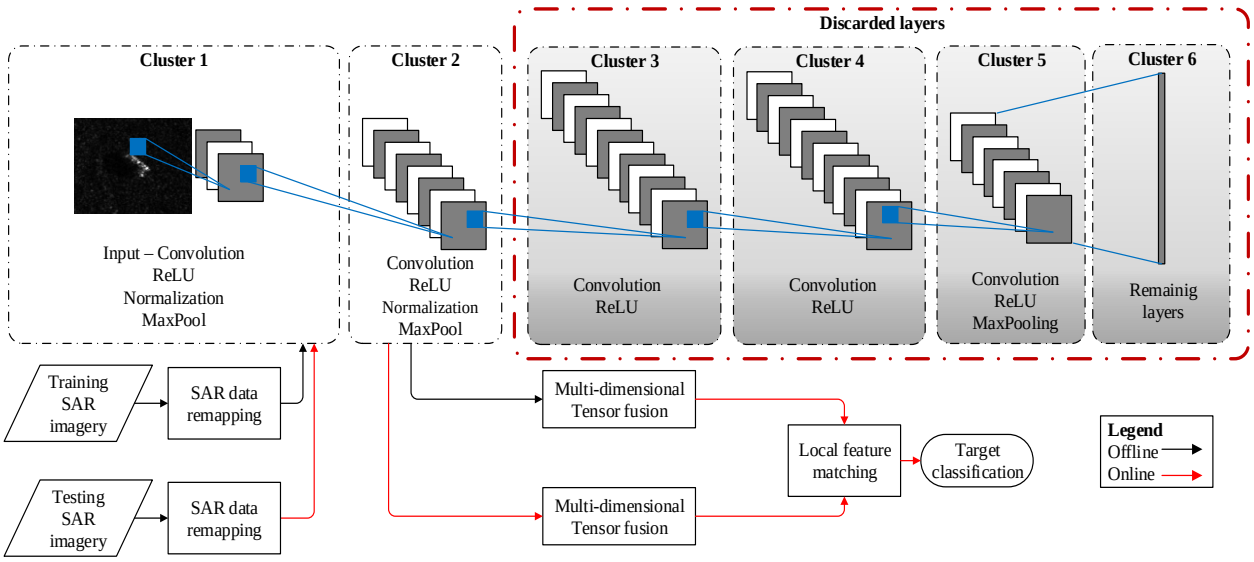


Fig.1 Proposed SAR ATR architecture showing activated cluster 2

Given a SAR image $\mathrm{I}^{m\,xn}$, $m,n \in \mathbb{Z}^{+}$ and $I(s,t) \in \{0,1,...,255\}$ with $1 \le s \le m$ and $1 \le t \le n$, we remap I into a 3D tensor by stacking in the third dimension three replicates of a processed version of

the SAR image I, notated as $B(I)$, in order to simulate the visual/ RGB image format and meet the image input requirements of VGG:

$$I_1 = B(I) \| B(I) \| B(I) \tag{1}$$

where $B(\cdot)$ is a bicubic interpolation process to resize I to the VGG image input size, and $\|\cdot$ is a 3D concatenation process.

Then I_1 is input to the C-VGG and is transformed into a 4D tensor $X^l \in \mathbb{R}^{H^l x W^l x D^l x N}$ that propagates through the hidden layers until it becomes the output Y^l of the end-layer of cluster l. *H*, *W* and *D* are the height, width and depth of the tensor at layer l and N refers to the mini batch size, i.e. the number of training instances used in one iteration to estimate the gradient of the loss function and update the parameters of the neural network. In our experiments we use $N = 1$ to increase accuracy and thus we convert the 4D tensor into a 3D one, i.e. $X^l = X^l_{i,j,d} \in \mathbb{R}^{H^l x W^l x D^l}$, where $0 \le i \le H^l$, $0 \le j \le W^l$, $0 \le d \le D^l$ (Fig. 2).

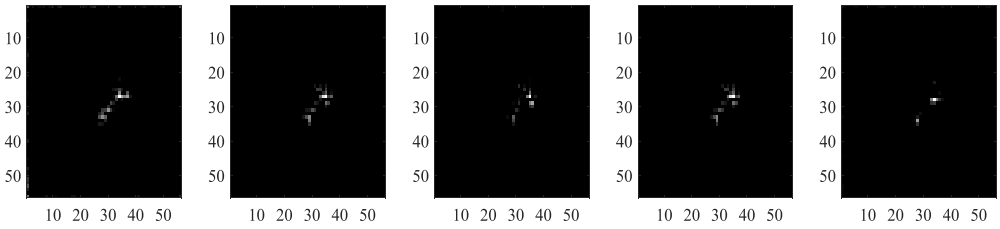


Fig. 2 Examples of 2D response maps of clustered layer *l=2*. The 3D tensor descriptor is the ensemble of the 2D matrices. (as an example only five out of the 128 response maps are shown)

3D tensors X^l and Y^l are stacks of 2D matrices that highlight features of various complexity in a response map type of representation (Fig 2.). As the X^l tensor propagates through the CNN's activated clusters and ultimately transforms to the output tensor Y^l, the tensor's size changes based

on the size of the convolutional kernel of each layer.

Therefore, tensors X^l and Y^l can be regarded as a generalized implementation of the scale-space theory [43] concept where the scale changes are envisaged via the subsequent shrinking of the convolutional kernel size and the degree of blurring via the kernel weights that are automatically adjusted by the CNN during the training stage.

As noted in Table 1, the output feature Y^l is provided by the end-layer of the activated clustered layer l that may be a Rectified Linear Unit (ReLU) layer or a Max Pooling layer. Therefore, it is important to present the operating details of these layers.

2.1.1 ReLU

This layer aims at increasing the non-linearity of a CNN by applying an individual truncation process on every $X^l_{i,j,d}$:

$$Y^l_{i,j,d} = \max\left\{0, X^l_{i,j,d}\right\} \tag{2}$$

where $Y^l_{i,j,d}$ is the output of cluster layer l.

The advantages of ReLU against the classic *tanh* activation function are the reduction in training time [37] and incorporating a purely supervised training scheme avoiding the need of unsupervised pre-training [44]. Current trends in neural networks either use a ReLU layer or its extension named the Parametrized ReLU that has an adaptive slope for the negative part of the activation function. In this paper, the CNNs evaluated use a ReLU activation function.

2.1.2 Max Pooling

This operation substitutes a sub-region X^l_s of size s x s named *pooling size* of the tensor $X^l_{i,j,d}$ with

its maximum value:

$$Y_{i,j,d}^{l} = \max\left(X_{s}^{l}\right) \tag{3}$$

The output size of $Y_{i,j,d}^{l}$ after the max pooling operations will be $H^{l+1} = H^{l} / s$ and $W^{l+1} = W^{l} / s$. Max pooling operates independently on each dimension d on a non-overlapping regional basis and therefore $D^{l+1} = D^{l}$.

2.2 Feature Fusion and Matching

Driven by the appealing classification performance and robustness to nuisances of the local feature based techniques [46], we partially adopt the sparse coding classification (SRC) [45] method and combine the advantages of both these theories with the 3D output feature of the proposed C-VGG $Y_{i,j,d}^{l}$. Specifically, we extend the method of [45] and perform a multi-dimensional tensor fusion process to convert the sparse 3D output tensor $Y_{i,j,d}^{l}$ (Fig. 2) into a single 1D-feature vector in order to input the latter to a local feature matching scheme. It is important to note that in contrast to the technique of [45] that exploits the raw pixel values of the entire model/ scene imagery, we take full advantage of the entire 3D tensor that encompasses the full response map of the activated layer providing an enhanced descriptiveness for the 1D-feature vector.

The multi-dimensional tensor fusion process comprises of a multi-dimensional vectorization process defined as:

$$\Theta_{i,d}\left(a_{i,j,d}\right) = \sum_{j=1}^{W} e_{j} \otimes a_{i,j,d} e_{j} \tag{4}$$

over dimension *j*, where e_j is the j^{th} canonical basis vector in the w-dimensional space and $\otimes$ the Kronecker product. The output of Eq. (4) is then followed by a vectorization procedure to create the 1D-feature vector:

$$f = vec\left(\underset{i=1,d=1}{\overset{H,d}{\Theta_{\omega}}} \left(Y^{l}_{i,\omega,d} \right) \right) \tag{5}$$

where $\omega = [1,...,d]$. The fusion product f encodes the features of the complete 3D tensor $Y^{l}_{i,j,d}$ in a 1D-vector form encompassing both the feature responses and the topology of the features for the entire 3D tensor depth.

Then we exploit the appealing classification performance and robustness of the local feature based techniques [46] by feeding the 1D-feature vector into a local feature matching strategy. Hence, given a scene feature f^S and the template features $f^T_{ii}, ii = \{1,2,...,k\}$, with *k* the number of templates, the proposed feature matching strategy relies on the *Cosine* similarity measure (*C*) that is combined with a Nearest Neighbor matching scheme [47]:

$$\Im\left\{ f^T_{ii}, f^S \right\} = \frac{\sum_{ii=1}^{k} f^T_{ii} f^S}{\sqrt{\sum_{ii=1}^{k} (f^T_{ii})^2} \sqrt{\sum_{ii=1}^{k} (f^S)^2}} \tag{6}$$

$$matched\ class\ m = \arg\min_{ii} \left(\Im\left\{ f^T_{ii}, f^S \right\} \right) \tag{7}$$

3 Experiments

In this section we evaluate the robustness of the proposed architecture on the MSTAR dataset. In order to be consistent with current literature we challenge our techniques against contemporary solutions on the 3-class target classification problem, the 10-class problem and on a number of nuisance factors such as depression angle, resolution and noise variation. Finally, we also extend

the suggested architecture to facilitate current state-of-the-art CNNs and several local feature match metrics.

3.1 MSTAR Dataset

We evaluate the performance of the proposed architecture on the publicly available subset of the MSTAR database [48] that includes 10 classes of ground targets as presented in Fig. 3. Each class contains chirps of 15° and 17° depression angles using an X-band SAR sensor, while some classes contain views from additional depression angles. In any case, chirps cover a full 0°-360° azimuth orientation. Table 2 presents the number of targets per type and depression angle used in this paper.

For compatibility reasons with current literature we adopt [48] and establish a training set based on the 17° depression angle. To validate the effectiveness of the combined C-VGG and *Cosine* similarity measure, dubbed C-VGG-C, we compare the ATR performance achieved by our architecture against current algorithms. All trials are implemented in MATLAB on an Intel i7 with 16GB RAM and for VGG the MatConvNet [49] version is used.

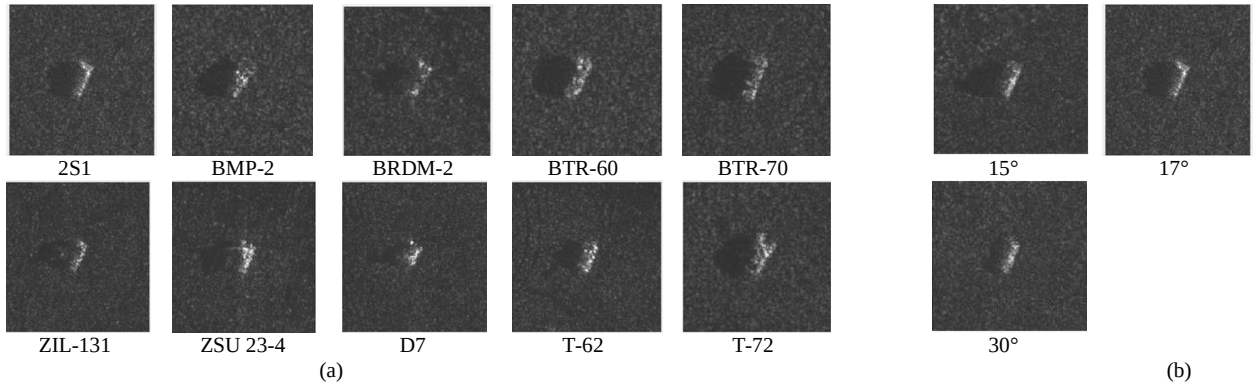


Fig. 3 (a) 10 classes of the MSTAR database at 17° depression angle (b) the 2S1 target at various depression angles

Table 2 MSTAR database used

Target	BMP-2			BTR-70	T-72			BTR-60	2S1	BRDM-2	D7	T-62	ZIL-131	ZSU 23/4	Sum
Serial Nº	9563	9566	c21	c71	132	812	s7	k1	b01	e71		a51	e12	d08	
train 17°	233	232	233	233	232	231	228	256	299	298	299	299	299	299	2747
test 15°	195	196	196	196	196	195	191	195	274	274	274	273	274	274	3203
test 30°	-	-	-	-	-	-	-	-	288	287	-	-	-	288	863

3.2 3-Class ATR

In this experiment, we aim at tuning the performance of our architecture by defining the optimal activation layer *l* of the C-VGG-C. The target classes used are the BMP-2, T-72 and BTR-70. For the former two we use all three variants namely for the BMP-2 the 9563, 9566 and c21 and for the T72 the 132, 812 and s7. For the BTR-70 we use the c71, which is the only variant available. As a reminder, images captured at 17° depression angle are used as training and images at 15° for testing.

Fig. 4 reveals that C-VVG-C excels at layer *l*=2 attaining a peak performance of 100% target recognition. The peak performance at this relatively shallow layer can be explained as follows. As the depth of the activated output layer increases, the extracted features are less generic e.g. corners, blobs, and become more complex. In parallel, as the depth of the output layer increases, features become more data specific for the training templates. Since the training and the testing data domain are substantially different (visual vs. SAR), our trials highlight that the balance between feature complexity and data training vs. testing variability is found in layer *l*=2. Driven by the high ATR

performance achieved, applying a dimensionality reduction strategy [50,51] to improve classification performance is not required.

The X-band SAR ATR concerned in this paper is envisaged to be applied on either ground based stations or aerial platforms such as large Unmanned Aerial Vehicles and aircrafts that have a sufficient size to host a standard CPU based processing architecture. In this context, current algorithms that are designed for low processing capability platforms [52,53] are not required.

We also compare the 3-class ATR performance achieved by our suggested architecture with current literature. From Table 3 it is evident that our proposed SAR ATR architecture gains both the highest overall ATR performance (100%) and achieves the highest inter-class ATR performance for each of the three classes.

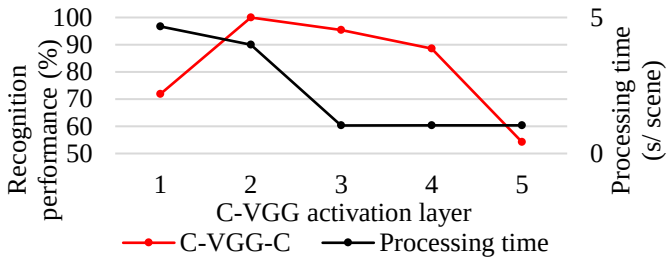


Fig. 4 Proposed architecture's SAR ATR performance on the 3-class problem

Table 3 3-class ATR (%)

Method	BMP-2	BTR-70	T-72	average
SRF [31]	93.56	96.43	96.91	95.63
Huang's [30]	94.38	98.47	96.91	96.04
DFSS [54]	91.65	99.48	96.04	95.72
ASC [33]	97.27	97.96	97.53	97.58
PCA [55]	97.44	99.49	95.92	97.61
BMO [32]	97.28	98.98	97.78	97.58
C-VGG-C	**100**	**100**	**100**	**100**

3.3 ATR Assessment Against Depression Variation

Next trial involves assessing the SAR ATR performance under various depression angles. Alike current literature [16,21,33,36,55–57], we use three similar targets, namely the 2S1, the BRDM-2 and the ZSU 23-4. Images at 17° depression angle are used as training images, while the 15° and 30° for testing. We intentionally do not evaluate the recognition performance at 45° as it is well known that SAR imagery is extremely sensitive to the depression angle variation and thus such an extensive depression variation cannot secure very high ATR rates that are mandatory for military applications.

From Table 4 it is evident that the suggested C-VGG CNN combined with a local feature-matching scheme based on the *Cosine* similarity measure can afford a higher ATR performance compared to current solutions. This is due to the low-level abstract features extracted by activating the $l = 2$ layer of the C-VGG CNN that are invariant to the large depression angle variations examined in this trial.

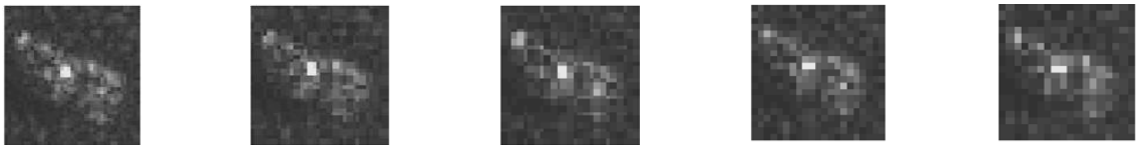
Table 4 3-class large depression variation ATR (%)

	ASC [33]	NNMF [21]	Bayesian CS [16]	JSRC [36]	PCA [55]	NMF [21]	EFS [56]	Zernike [57]	C-VGG-C
15°	99.15	98.91	99.20	99.50	98.65	99.25	97.88	96.46	**99.88**
30°	97.91	91.42	89.60	91.80	97.82	98.24	93.42	93.24	**99.88**

3.4 ATR Assessment Against Resolution Variation

We evaluate the robustness of our SAR ATR architecture under different resolution variations ranging from 0.3m×0.3m, which is the original resolution, down to 0.7m×0.7m. Fig. 5 shows a target from the MSTAR database under these resolutions. In Fig. 6, we show that our ATR architecture achieves 98.64% ATR even under the lowest resolution case of 0.7m×0.7m. This

performance is much higher compared to ASC [33] (85.9%). This is mainly because ASC performs feature matching based on the *Kullback-Leibler divergence* while the suggest feature matching scheme is based on the *Cosine similarity* metric. In Section 3.8 we demonstrate that the *Cosine similarity* metric is quite insensitive to several nuisance factors, including resolution variation, affording a robust ATR performance.



(a) (b) (c) (d) (e)

Fig. 5 MSTAR images (focusing on the target) at different resolutions (a) 0.3m×0.3m (original) (b) 0.4m×0.4m (c) 0.5m×0.5m (d) 0.6m×0.6m (e) 0.7m×0.7m.

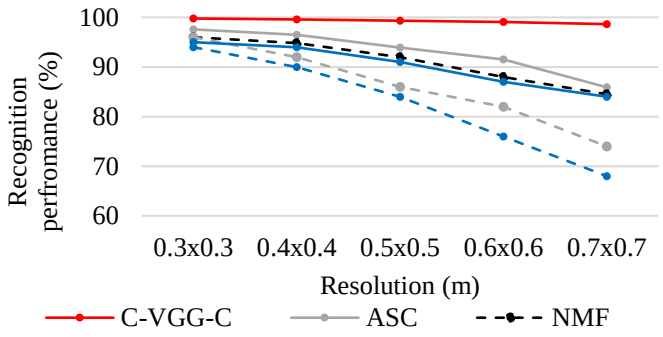


Fig. 6 Robustness in various scene resolutions

3.5 10-class ATR

For the 10-class ATR problem, the MSTAR related literature suggests various target set configurations. Two commonly used configurations are the Standard Operation Conditions 1 (SOC-1) and the SOC-2. Both exploit all 10 classes with the difference being that SOC-1 includes only the 9563 serial number for BMP-2 and serial number 132 for T-72, while SOC-2 all available

serial numbers for these two targets as presented in Table 2. Thus, SOC-1 is essentially a 10-class and SOC-2 a 14-class ATR problem. For both target set configurations, the 17° depression angle is used for training and the 15° for testing. Table 5 compares the ATR performance achieved by C-VGG-C on SOC-1 against current literature and Table 6 presents the corresponding confusion matrix of our technique. Table 7 and 8 present the corresponding results for SOC-2. Results on both SOC trials highlight that C-VGG-C outperforms current ATR algorithms as it attains 99.71% ATR on SOC-1 and 99.79% on SOC-2.

Table 5 SOC-1 average ATR performance (%)

Chen's [8]	MtCS [35]	Bayesian CS [16]	SAE [20]	DNN [17]	C-VGG-C
84.70	84.00	92.60	95.40	96.00	**99.71**

Table 6 SOC-1 confusion matrix of the proposed C-VGG-C architecture

class	2S1	BMP-2	BRDM-2	BTR-60	BTR-70	D7	T-62	T-72	ZIL-131	ZSU 23-4	recognition (%)
2S1	**273**	0	0	0	0	0	0	0	1	0	99.64
BMP-2	0	**196**	0	0	0	0	0	0	0	0	100
BRDM-2	0	0	**269**	1	0	0	0	4	0	0	98.18
BTR-60	0	0	0	**195**	0	0	0	0	0	0	100
BTR-70	0	0	0	0	**196**	0	0	0	0	0	100
D7	0	0	0	0	0	**272**	0	0	0	2	99.27
T-62	0	0	0	0	0	0	**273**	0	0	0	100
T-72	0	0	0	0	0	0	0	**196**	0	0	100
ZIL-131	0	0	0	0	0	0	0	0	**274**	0	100
ZSU 23-4	0	0	0	0	0	0	0	0	0	**274**	100
average											**99.71**

Table 7 SOC-2 average ATR performance (%)

method	DNN [17]	IGT [58]	Morgan's [15]	BMO [32]	KM [29]	ASC [33]	EFS [56]	Zernike [57]	PCA [55]	NMF [21]	C-VGG-C
avg	95.00	95.00	92.30	95.74	84.58	95.41	94.10	93.46	90.24	93.76	**99.79**

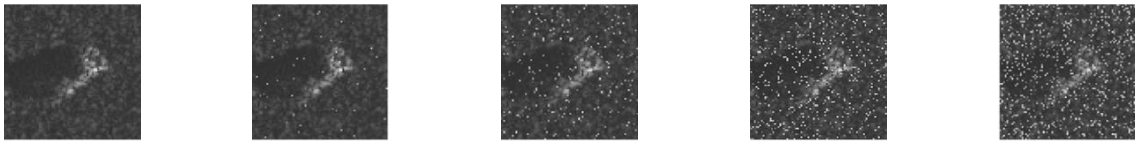
Table 8 SOC-2 confusion matrix of the proposed C-VGG-C architecture

target	2S1	BMP-2	BRDM-2	BTR-60	BTR-70	D7	T-62	T-72	ZIL-131	ZSU 23-4	recognition (%)
2S1	**273**	0	0	0	0	0	0	0	1	0	99.64
BMP-2 (9566)	0	**196**	0	0	0	0	0	0	0	0	100
BMP-2 (9563)	0	**196**	0	0	0	0	0	0	0	0	100
BMP-2 (c21)	0	**196**	0	0	0	0	0	0	0	0	100
BRDM-2	0	0	**269**	1	1	0	0	1	0	2	98.18
BTR-60	0	0	0	**195**	0	0	0	0	0	0	100
BTR-70	0	0	0	0	**196**	0	0	0	0	0	100
D7	0	0	0	0	0	**272**	0	0	0	2	99.27
T-62	0	0	0	0	0	0	**273**	0	0	0	100
T-72 (132)	0	0	0	0	0	0	0	**196**	0	0	100
T-72 (812)	0	0	0	0	0	0	0	**195**	0	0	100
T-72 (s7)	0	0	0	0	0	0	0	**191**	0	0	100
ZIL-131	0	0	0	0	0	0	0	0	**274**	0	100
ZSU 23-4	0	0	0	0	0	0	0	0	0	**274**	100
average											**99.79**

3.6 ATR Assessment Against Noise

In this trial, we evaluate the robustness of C-VGG-C against current methods on the SOC-1 dataset where noise is added. Noise simulation is consistent with the literature [6,11] ,i.e. we randomly select a percentage of pixels in the target scene and replace their value with samples generated from a Gaussian distribution. Fig. 7 presents a scene image from the MSTAR dataset under the various noise levels simulated in this trial. The performance gained on the SOC-1 dataset is presented in Fig. 8, where our suggested technique presents a considerable improvement over current methods. This is because the interpolation process presented in Eq. (1) smooths the noise nuisances, and combined with the C-VGG features and the *Cosine similarity* metric affords to our

suggested architecture a robust performance. Regarding the match metric, Section 3.8 demonstrates the effectiveness of the *Cosine similarity* metric.



(a) (b) (c) (d) (e)

Fig. 7 MSTAR images at different noise levels (a) 0% (original) (b) 1% (c) 5% (d) 10% (e) 15%

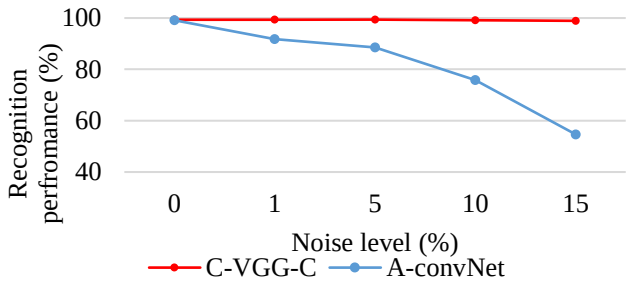


Fig. 8 Robustness in various noise levels

3.7 Extending to Other Mainstream CNNs

We also extend the suggested layer-clustering strategy to the AlexNet, GoogleNet and ResNet CNNs. For AlexNet we use MATLAB's implementation while for GoogleNet and ResNet their MatConvNet [49] implementations. The clustering methodology for each CNN is equivalent to the one used for C-VGG, i.e. each cluster contains one convolutional layer. Based on the tuning process presented in Section 3.2, the optimum activation layer for the clustered AlexNet (C-AlexNet) is *l=2* that concludes with the *MaxPool_2* layer and for clustered GoogleNet (C-GoogleNet) is *l=2* ending with the *Pool_2* layer. Finally, for the clustered ResNet (C-ResNet) optimum ATR is achieved at *l=3* that concludes with the *res2a_branch2b* layer.

The first trial involves evaluating the ATR performance in the 3-class recognition case of Section 3.2. Table 9 shows the coherency of our clustered CNN concept as it affords a high ATR performance for every mainstream CNN. This is also evident from the results in the 10-class SOC-1 dataset presented in Table 10.

Table 9 3-class ATR (%) per clustered CNN variant

	C-VGG-C	C-GoogleNet-C	C-AlexNet-C	C-ResNet-C
BMP-2	**100**	95.40	99.66	97.78
BTR-70	**100**	96.94	**100**	97.96
T-72	**100**	**100**	99.83	**100**
average	**100**	97.45	99.83	98.58

Table 10 SOC-1 10-class ATR (%) per clustered CNN variant

	C-VGG-C	C-GoogleNet-C	C-AlexNet-C	C-ResNet-C
average	**99.74**	98.2	99.68	98.06

3.8 Extending to Other Distance/ Similarity Metrics

We also extend the C-VGG concept to several distance/ similarity measures used by various computer vision algorithms. The measures evaluated are based on the grouping of [59] and are:

3.8.1 L_p Minkowski family

L1-norm, which measures the absolute value distance:

$$\Im\left\{ f_{ii}^{T}, f^{S} \right\} = \sum_{ii=1}^{k} \left| f_{ii}^{T} - f^{S} \right| \tag{8}$$

L2-norm or Euclidean, which measures the shortest distance:

$$\Im\left\{ f_{ii}^{T}, f^{S} \right\} = \sqrt{\sum_{ii=1}^{k} \left| f_{ii}^{T} - f^{S} \right|^{2}} \tag{9}$$

3.8.2 Intersection family

Tanimoto metric that compares the similarity and diversity of the features

$$\Im\left\{ f_{ii}^{T}, f^{S} \right\} = \sum_{ii=1}^{k} \frac{f_{ii}^{T} f^{S}}{\sqrt{(f_{ii}^{T})^{2} + (f^{S})^{2}} - f_{ii}^{T} f^{S}} \tag{10}$$

3.8.3 Inner Product family

Cosine, as introduced Eq. (6).

Jaccard

$$\Im\left\{ f_{ii}^{T}, f^{S} \right\} = 1 - \frac{\sum_{ii=1}^{k} f_{ii}^{T} f^{S}}{\sum_{ii=1}^{k} (f_{ii}^{T})^{2} + \sum_{ii=1}^{k} (f^{S})^{2} - \sum_{ii=1}^{k} f_{ii}^{T} f^{S}} \tag{11}$$

Fidelity family

that measures the similarity of two probability distributions

$$\Im\left\{ f_{ii}^{T}, f^{S} \right\} = -\log\left(\sum_{ii=1}^{k} \sqrt{f_{ii}^{T} - f^{S}} \right) \tag{12}$$

Hellinger,

$$H = \Im\left\{ f_{ii}^{T}, f^{S} \right\} = \sqrt{2 \sum_{ii=1}^{k} \left(\sqrt{f_{ii}^{T}} - \sqrt{f^{S}} \right)} \tag{13}$$

3.8.5 Shannon entropy family

Kullback-Leibler divergence, which measures the similarity by calculating the relative entropy*:*

$$\Im\left\{f_{ii}^{T}, f^{S}\right\} = \sum_{ii=1}^{k}\left(\left(f_{ii}^{T} - f^{S}\right)\log\left(\frac{f_{ii}^{T}}{f^{S}}\right)\right) \tag{14}$$

Shannon Entropy function, which measures the disorder of the features

$$\Im\left\{f_{i}^{T}, f^{S}\right\} = \sum_{ii=1}^{k}\left(-\left(f_{ii}^{T} - f^{S}\right)\log\left(f_{ii}^{T} - f^{S}\right)\right) \tag{15}$$

3.8.6 x^2 family

x^2 distance, which measures the underlying distance of the features and emphasizes their dissimilarity:

$$\Im\left\{f_{ii}^{T}, f^{S}\right\} = \sum_{ii=1}^{k}\left(\frac{f_{ii}^{T} - \frac{f_{ii}^{T} + f^{S}}{2}}{\frac{f_{ii}^{T} + f^{S}}{2}}\right) \tag{16}$$

In addition, we also investigate the SAR ATR performance by substituting the distance/ similarity measure with a Multi-class Support Vector Machine (M-SVM) scheme similar to the strategy suggested in [60].

Table 11 presents the ATR performance attained by each C-VGG vs. measure combination. From Table 11 the following conclusions can be made:

a. Even though the *Cosine* measure excels, the majority of the measures evaluated achieve a quite appealing SAR ATR performance.

b. The distance/ similarity measure has a substantial impact on the ATR performance. Nevertheless, the majority of the distance/ similarity measures attains a high ATR performance validating the robustness of the suggested clustering method.

c. The performance of each metric is associated with the distance/ similarity measure family that it belongs. From our trials this is clearly demonstrated as measures from the same family have a similar performance.

We also highlight the contribution of the *Cosine* similarity measure to achieve high performing ATR on SAR imagery that is affected by noise and subsampling nuisances. For that purpose, we corrupt a scene feature $f^{S}_{corrupted}$ with the noise and subsampling levels of Section 3.4 and 3.6 respectively, and calculate the feature distance/ similarity measure to the uncorrupted f^{S} scene feature. Both $f^{S}_{corrupted}$ and f^{S} are extracted using the C-VGG architecture and are then matched using the match/ similarity metrics presented. Fig. 9 shows the distance/ similarity per metric from which the following conclusions can be made:

a. The importance of the feature match metric is evident because the matching distance between $f^{S}_{corrupted}$ and f^{S} highly depends on the metric itself.

b. The *Cosine* similarity metric affords the smallest feature metric between $f^{S}_{corrupted}$ and f^{S}, with *Jaccard* to follow. It is worth noting that even when the level of corruption increases substantially, the *Cosine* based feature metric remains quite stable. Hence, feature matching is only minor affected by nuisance factors affording the high quality ATR performance demonstrated in Fig. 6 and Fig. 8. This can be explained as both *Cosine* and *Jaccard* involve the angular variation of the feature vectors $f^{S}_{corrupted}$ and f^{S} rather than their distance, which is the norm in a feature matching scheme.

c. The hierarchy of the noise nuisance trial (Fig. 9 (a)) and resolution variation trial (Fig. 9 (b)) is the same, enhancing the validity of selecting the Cosine metric.

Table 11 3-class ATR (%) for C-VGG and various feature distance/ similarity measures

	L_p Minkowski		Inter-section	Inner product		Fidelity		Shannon			x^2
	L_1-norm	L_2-norm	Tanimoto	Cosine	Jaccard	Bhattacha-ryya	Hellinger	Kullback-Liebler	Shannon	M-SVM	x^2
BMP-2	96.59	97.62	87.18	**100**	90.46	97.96	99.66	100	93.70	22.83	97.45
BTR-70	91.33	94.39	82.14	**100**	55.61	89.80	100	0	41.54	10.71	93.37
T-72	96.40	97.95	96.38	**100**	68.16	93.16	99.66	0	45.23	95.38	98.29
average	94.77	96.65	88.57	**100**	71.41	93.64	99.77	33.33	60.16	52.19	96.37

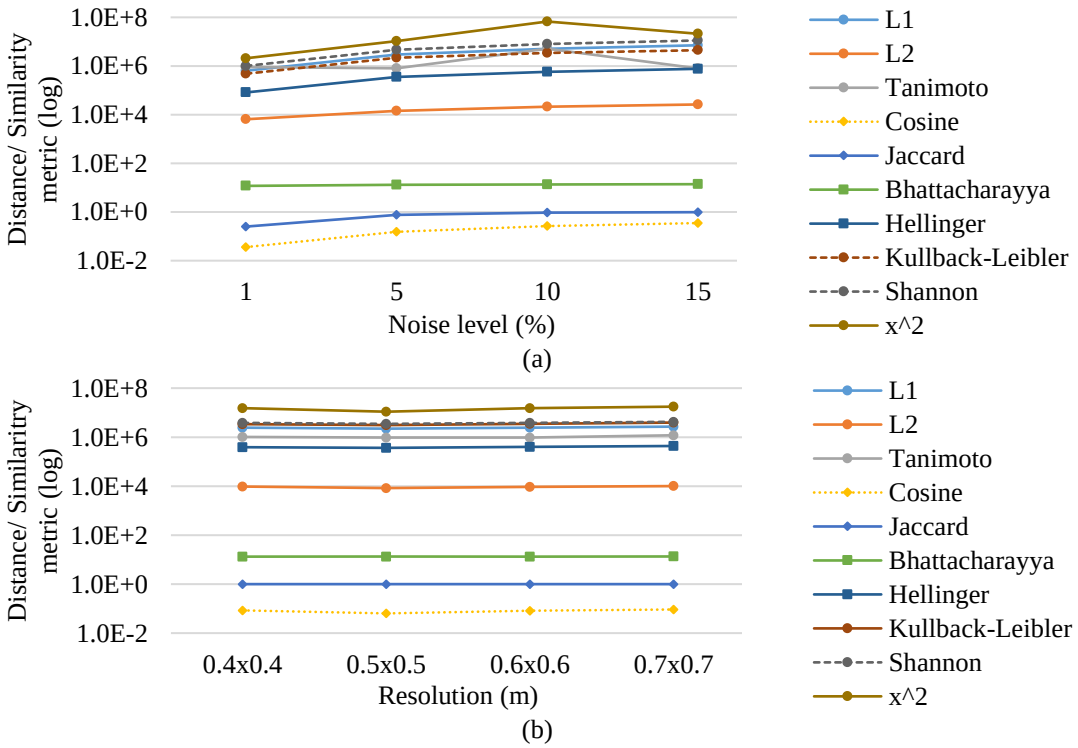


Fig. 9 Robustness in various nuisances and levels (a) noise (b) resolution

3.9 Discussion

The robustness of the proposed architecture is due to combining the suggested clustered CNN, the multi-dimensional vectorization process and the *cosine* similarity metric. Specifically, the 3D

response tensor Y^l of the activated clustered VGG layer reveals different and distinctive local patches of the target. In addition, applying the multi-dimensional vectorization process on Y^l exploits the entire the 3D tensor, i.e. the complete 3D response map, enhancing the distinctiveness and robustness of the 1D-feature vector to inter-class and intra-class variation as well as to nuisance factors. An additional advantage of the multi-dimensional vectorization process is converting the 3D topology of the features within Y^l into 1D-feature vector without any information loss. Finally, the performance of the suggested architecture is further enhanced by exploiting the *cosine* similarity metric that is appropriate for classification tasks taking full advantage the suggested highly descriptive 1D-feature vector.

4 Conclusion

Deep learning techniques are widely used for SAR ATR and aim at extracting deep features that can uniquely describe a target within a SAR image. Instead of handcrafted CNNs, the suggested strategy extends the usability of the state-of-the-art pre-trained in the visual domain CNNs into the SAR data domain by clustering the CNN layers into feature-specific based layers. Specifically, SAR imagery is remapped to meet the requirements of the clustered VGG CNN, then a number of clusters are activated and the output response is transformed into a 1D-feature vector by applying a multi-dimensional tensor fusion. Template and scene feature vectors are matched based on the *Cosine* similarity measure.

Experimental results on the MSTAR data set under various configurations and nuisances such as 10-class and 3-class ATR problems with and without target variants, noise, large depression angle variation and resolution variation, illustrate the effectiveness of our suggested architecture against current ATR techniques. We also demonstrate that among current CNNs used by the

computer vision community, the combination of VGG with a *Cosine* measure can afford a highly appealing and robust ATR performance.

5 References

[1] Gray, G. J., Aouf, N., Richardson, M. A., Butters, B.., Walmsley, R., "An intelligent tracking algorithm for an imaging infrared anti-ship missile," Proc. SPIE 8543, Technol. Opt. Countermeas. IX **8543**, D. H. Titterton and M. A. Richardson, Eds., 85430L–85430L (2012).

[2] Gray, G. J., Aouf, N., Richardson, M. A., Butters, B., Walmsley, R.., Nicholls, E., "Feature-based recognition approaches for infrared anti-ship missile seekers," Imaging Sci. J. **60**(6), D. H. Titterton and M. A. Richardson, Eds., 305–320 (2012).

[3] Sun, S.-G., "Automatic target recognition using boundary partitioning and invariant features in forward-looking infrared images," Opt. Eng. **42**(2), 524, International Society for Optics and Photonics (2003).

[4] Gray, G., Aouf, N., Richardson, M. A., Butters, B., Walmsley, R.., Nicholls, E., "Feature-Based Target Recognition in Infrared Images for Future Unmanned Aerial Vehicles," J. Battlef. Technol. **14**(2), 27–36 (2011).

[5] Kechagias-Stamatis, O., Aouf, N.., Nam, D., "Multi-Modal Automatic Target Recognition for Anti-Ship Missiles with Imaging Infrared Capabilities," 2017 Sens. Signal Process. Def. Conf., 1–5, IEEE (2017).

[6] Chen, S., Wang, H., Xu, F.., Jin, Y.-Q., "Target Classification Using the Deep Convolutional Networks for SAR Images," IEEE Trans. Geosci. Remote Sens. **54**(8), 4806–4817 (2016).

[7] Zhong, Y.., Ettinger, G., "Enlightening Deep Neural Networks with Knowledge of Confounding Factors," arXiv Prepr. arXiv1607.02397, 1–10 (2016).

[8] Chen, S.., Wang, H., "SAR target recognition based on deep learning," 2014 Int. Conf. Data Sci. Adv. Anal., 541–547, IEEE (2014).

[9] Paladini, R., Martorella, M.., Berizzi, F., "Classification of Man-Made Targets via Invariant Coherency-Matrix Eigenvector Decomposition of Polarimetric SAR/ISAR Images," IEEE Trans. Geosci. Remote Sens. **49**(8), 3022–3034 (2011).

[10] Perissin, D.., Ferretti, A., "Urban-Target Recognition by Means of Repeated Spaceborne SAR Images," IEEE Trans. Geosci. Remote Sens. **45**(12), 4043–4058 (2007).

[11] Dong, G., Wang, N.., Kuang, G., "Sparse Representation of Monogenic Signal: With Application to Target Recognition in SAR Images," IEEE Signal Process. Lett. **21**(8), 952–956 (2014).

[12] Zheng, C., Jiang, X.., Liu, X., "Generalized synthetic aperture radar automatic target recognition by convolutional neural network with joint use of two-dimensional principal component analysis and support vector machine," J. Appl. Remote Sens. **11**(04), 1 (2017).

[13] Amrani, M.., Jiang, F., "Deep feature extraction and combination for synthetic aperture radar target classification," J. Appl. Remote Sens. **11**(04), 1 (2017).

[14] Profeta, A., Rodriguez, A.., Clouse, H. S., "Convolutional neural networks for synthetic aperture radar classification," SPIE Def. + Secur. Int. Soc. Opt. Photonics **9843**, E. Zelnio and F. D. Garber, Eds., 98430M (2016).

[15] Morgan, D. A. E., "Deep convolutional neural networks for ATR from SAR imagery," SPIE Def. + Secur. **9475**, 94750F (2015).

[16] Zhang, X., Qin, J.., Li, G., "SAR target classification using Bayesian compressive sensing with scattering centers features," Prog. Electromagn. Res. **136**, 385–407, EMW Publishing (2013).

[17] Doo, S. H., Smith, G. E., Baker, C. J., Cui, Z., Feng, J., Cao, Z., Ren, H.., Yang, J., "Aspect invariant features for radar target recognition," IET Radar, Sonar Navig. **11**(4), 597–604 (2017).

[18] El-Darymli, K., McGuire, P., Gill, E. W., Power, D.., Moloney, C., "Holism-based features for target classification in focused and complex-valued synthetic aperture radar imagery," IEEE Trans. Aerosp. Electron. Syst. **52**(2), 786–808 (2016).

[19] Deng, S., Du, L., Li, C., Ding, J.., Liu, H., "SAR Automatic Target Recognition Based on Euclidean Distance Restricted Autoencoder," IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. **10**(7), 3323–3333 (2017).

[20] Kang, M., Ji, K., Leng, X., Xing, X.., Zou, H., "Synthetic Aperture Radar Target Recognition with Feature Fusion Based on a Stacked Autoencoder," Sensors **17**(12), 192, Multidisciplinary Digital Publishing Institute (2017).

[21] Cui, Z., Feng, J., Cao, Z., Ren, H.., Yang, J., "Target recognition in synthetic aperture radar images via non-negative matrix factorisation," IET Radar, Sonar Navig. **9**(9), 1376–1385 (2015).

[22] Martorella, M., Giusti, E., Capria, A., Berizzi, F.., Bates, B., "Automatic Target Recognition by Means of

Polarimetric ISAR Images and Neural Networks," IEEE Trans. Geosci. Remote Sens. **47**(11), 3786–3794, IEEE (2009).

[23] Kechagias-Stamatis, O., Aouf, N.., Richardson, M. A., "3D automatic target recognition for future LIDAR missiles," IEEE Trans. Aerosp. Electron. Syst. **52**(6), 2662–2675 (2016).

[24] Vasile, A.., Marino, R., "Pose-independent automatic target detection and recognition using 3D laser radar imagery," Lincoln Lab. J. **15**(1), 61–78 (2005).

[25] Kechagias-Stamatis, O.., Aouf, N., "Evaluating 3D local descriptors for future LIDAR missiles with automatic target recognition capabilities," Imaging Sci. J. **65**(7), 428–437 (2017).

[26] Kechagias-Stamatis, O., Aouf, N., Gray, G., Chermak, L., Richardson, M.., Oudyi, F., "Local feature based automatic target recognition for future 3D active homing seeker missiles," Aerosp. Sci. Technol. **73**, 309–317 (2018).

[27] Kechagias-Stamatis, O., Aouf, N.., Nam, D., "3D Automatic Target Recognition for UAV Platforms," 2017 Sens. Signal Process. Def. Conf., 1–5, IEEE (2017).

[28] Brown, W. M.., Swonger, C. W., "A Prospectus for Automatic Target Recognition," IEEE Trans. Aerosp. Electron. Syst. **25**(3), 401–410 (1989).

[29] Clemente, C., Pallotta, L., Gaglione, D., De Maio, A.., Soraghan, J. J., "Automatic Target Recognition of Military Vehicles With Krawtchouk Moments," IEEE Trans. Aerosp. Electron. Syst. **53**(1), 493–500 (2017).

[30] Huang, X., Nie, X., Wu, W., Qiao, H.., Zhang, B., "SAR target configuration recognition based on the biologically inspired model," Neurocomputing **234**, 185–191, Elsevier B.V. (2017).

[31] Yang, S., Wang, M., Long, H.., Liu, Z., "Sparse Robust Filters for scene classification of Synthetic Aperture Radar (SAR) images," Neurocomputing **184**, 91–98, Elsevier (2016).

[32] Ding, B., Wen, G., Ma, C.., Yang, X., "Target recognition in synthetic aperture radar images using binary morphological operations," J. Appl. Remote Sens. **10**(4), 046006 (2016).

[33] Ding, B., Wen, G., Zhong, J., Ma, C.., Yang, X., "A robust similarity measure for attributed scattering center sets with application to SAR ATR," Neurocomputing **219**, 130–143, Elsevier (2017).

[34] Gorovyi, I. M.., Sharapov, D. S., "Efficient object classification and recognition in SAR imagery," 2017 18th Int. Radar Symp., 1–7, IEEE (2017).

[35] Liu, S., Zhan, R., Zhai, Q., Wang, W.., Zhang, J., "Multi-view radar target recognition based on multitask

compressive sensing," J. Electromagn. Waves Appl. **29**(14), 1917–1934, Taylor & Francis (2015).

[36] Zhang, H., Nasrabadi, N. M., Zhang, Y.., Huang, T. S., "Multi-view automatic target recognition using joint sparse representation," IEEE Trans. Aerosp. Electron. Syst. **48**(3), 2481–2497 (2012).

[37] Krizhevsky, A., Sutskever, I.., Hinton, G., "Imagenet classification with deep convolutional neural networks," Adv. Neural Inf. Process. Syst., 1097–1105 (2012).

[38] Simonyan, K.., Zisserman, A., "Very Deep Convolutional Networks for Large-Scale Image Recognition," arXiv Prepr. arXiv1409.1556v6, 1–14 (2014).

[39] Szegedy, C., Wei Liu., Yangqing Jia., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V.., Rabinovich, A., "Going deeper with convolutions," 2015 IEEE Conf. Comput. Vis. Pattern Recognit., 1–9, IEEE (2015).

[40] He, K., Zhang, X., Ren, S.., Sun, J., "Deep Residual Learning for Image Recognition," 2016 IEEE Conf. Comput. Vis. Pattern Recognit., 770–778, IEEE (2016).

[41] Pan, S. J.., Yang, Q., "A Survey on Transfer Learning," IEEE Trans. Knowl. Data Eng. **22**(10), 1345–1359 (2010).

[42] Kang, C.., He, C., "SAR image classification based on the multi-layer network and transfer learning of mid-level representations," 2016 IEEE Int. Geosci. Remote Sens. Symp., 1146–1149, IEEE (2016).

[43] Lindeberg, T., "Scale-space theory: a basic tool for analyzing structures at different scales," J. Appl. Stat. **21**(1), 225–270 (1994).

[44] Glorot, X., Bordes, A.., Bengio, Y., "Deep Sparse Rectifier Neural Networks," Proc. Fourteenth Int. Conf. Artif. Intell. Stat. **15**, G. Gordon, D. Dunson, and M. Dudík, Eds., 315–323, PMLR, Fort Lauderdale, FL, USA (2011).

[45] Wright, J., Yang, A. Y., Ganesh, A., Sastry, S. S.., Yi Ma., "Robust Face Recognition via Sparse Representation," IEEE Trans. Pattern Anal. Mach. Intell. **31**(2), 210–227 (2009).

[46] Mikolajczyk, K.., Schmid, C., "Performance evaluation of local descriptors.," IEEE Trans. Pattern Anal. Mach. Intell. **27**(10), 1615–1630 (2005).

[47] Mikolajczyk, K.., Schmid, C., "A performance evaluation of local descriptors," 2003 IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognition, 2003. Proceedings. **2**, II-257-II-263, IEEE Comput. Soc.

[48] Ross, T. D., Worrell, S. W., Velten, V. J., Mossing, J. C.., Bryant, M. L., "Standard SAR ATR Evaluation

Experiments using the MSTAR Public Release Data Set," SPIE Conf. Algorithms Synth. Aperture Radar Imag. V **3370**, E. G. Zelnio, Ed., 566–573 (1998).

[49] Vedaldi, A.., Lenc, K., "MatConvNet," Proc. 23rd ACM Int. Conf. Multimed. - MM '15, 689–692, ACM Press, New York, New York, USA (2015).

[50] Zhang, J., Yu, J.., Tao, D., "Local Deep-Feature Alignment for Unsupervised Dimension Reduction," IEEE Trans. Image Process. **27**(5), 2420–2432 (2018).

[51] Gao, Y., Beijbom, O., Zhang, N.., Darrell, T., "Compact Bilinear Pooling," 2016 IEEE Conf. Comput. Vis. Pattern Recognit., 317–326, IEEE (2016).

[52] Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M.., Adam, H., "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications" (2017).

[53] Zhang, X., Zhou, X., Lin, M.., Sun, J., "ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices" (2017).

[54] Liu, H.., Li, S., "Decision fusion of sparse representation and support vector machine for SAR image target recognition," Neurocomputing **113**, 97–104, Elsevier (2013).

[55] Zhang, R., Hong, J., Ming, F., Rui Zhang., Jun Hong., Feng Ming., "An improved PCA based features for SAR ATR," IET Conf. Publ.(551 CP), 313–313, IET (2009).

[56] Anagnostopoulos, G. C., "SVM-based target recognition from synthetic aperture radar images using target region outline descriptors," Nonlinear Anal. Theory, Methods Appl. **71**(12), 2934–2939 (2009).

[57] Rezai-rad, G.., Amoon, M., "Automatic target recognition of synthetic aperture radar (SAR) images based on optimal selection of Zernike moments features," IET Comput. Vis. **8**(2), 77–85 (2014).

[58] Srinivas, U., Monga, V.., Raj, R. G., "SAR Automatic Target Recognition Using Discriminative Graphical Models," IEEE Trans. Aerosp. Electron. Syst. **50**(1), 591–606 (2014).

[59] Cha, S., "Comprehensive Survey on Distance / Similarity Measures between Probability Density Functions," Int. J. Math. Model. Methods Appl. Sci. **1**(4), 300–307 (2007).

[60] Wagner, S. A., "SAR ATR by a combination of convolutional neural network and support vector machines," IEEE Trans. Aerosp. Electron. Syst. **52**(6), 2861–2872 (2016).

Odysseas Kechagias-Stamatis received the MSc degree in Guided Weapon Systems and the PhD degree in 3D ATR for missile platforms from Cranfield University, U.K. in 2011 and 2017 respectively. His research interests include 2D/3D object recognition and tracking, data fusion and autonomy of systems.

Caption List

Table 1 C-VGG layers

Table 2 MSTAR database used

Table 3 3-class ATR (%)

Table 4 3-class large depression variation ATR (%)

Table 5 SOC-1 average ATR performance (%)

Table 6 SOC-1 confusion matrix of the proposed C-VGG-C architecture

Table 7 SOC-2 average ATR performance (%)

Table 8 SOC-2 confusion matrix of the proposed C-VGG-C architecture

Table 9 3-class ATR (%) per clustered CNN variant

Table 10 SOC-1 10-class ATR (%) per clustered CNN variant

Table 11 3-class ATR (%) for C-VGG and various feature distance/ similarity measures

Fig.1 Proposed SAR ATR architecture showing activated cluster 2

Fig. 2 Examples of 2D response maps of clustered layer $l=2$. The 3D tensor descriptor is the ensemble of the 2D matrices. (as an example only five out of the 128 response maps are shown)

Fig. 3 (a) 10 classes of the MSTAR database at 17° depression angle (b) the 2S1 target at various depression angles

Fig. 4 Proposed architecture's SAR ATR performance on the 3-class problem

Fig. 5 MSTAR images (focusing on the target) at different resolutions (a) 0.3m×0.3m (original) (b) 0.4m×0.4m (c) 0.5m×0.5m (d) 0.6m×0.6m (e) 0.7m×0.7m.

Fig. 6 Robustness in various scene resolutions

Fig. 7 MSTAR images at different noise levels (a) 0% (original) (b) 1% (c) 5% (d) 10% (e) 15%

Fig. 8 Robustness in various noise levels

Fig. 9 Robustness in various nuisances and levels (a) noise (b) resolution