

RESEARCH ARTICLE



OPEN ACCESS

Received: 12.08.2021

Accepted: 21.01.2022

Published: 05.03.2022

Citation: Selvam N, Nagesa Y, Nagesa F (2022) Deep Learning Approach with Optimization Algorithm for Reducing the Training and Testing Time in SAR Image Detection and Recognition. Indian Journal of Science and Technology 15(9): 371-385. <https://doi.org/10.17485/IJST/v15i9.1266>

* **Corresponding author.**

rajanksrcas@gmail.com

Funding: None

Competing Interests: None

Copyright: © 2022 Selvam et al. This is an open access article distributed under the terms of the [Creative Commons Attribution License](https://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Published By Indian Society for Education and Environment ([iSee](https://www.indjst.org/))

ISSN

Print: 0974-6846

Electronic: 0974-5645

Deep Learning Approach with Optimization Algorithm for Reducing the Training and Testing Time in SAR Image Detection and Recognition

Nagarajan Selvam^{1*}, Yonatan Nagesa², Fikiru Negesa³

¹ Faculty, Department of Computer Science, Ambo University, Ethiopia

² Research scholar, Department of Computer Science, Ambo University, Ethiopia

³ Research scholar, Department of Health Informatics, Ambo University, Ethiopia

Abstract

Objective: - To reduce training and testing time in SAR image detection and recognition using optimization algorithm. **Methods:** SAR images have achieved a prominent position in the arena of remote sensing and satellite technology. SAR image processing has many applications in different areas like agriculture, mineral exploration, resource management and environmental monitoring. To carry out this research works, MSTAR dataset was used with six classes. The collected dataset was preprocessed with filtering algorithms to remove the speckle noise from the image. Then, image segmentation which is essential expertise for image processing has been done with aim of rationalizing and changing image representation into more meaningful and easier to analyze. The characters of Hue, Intensity, Saturation (H, I, S) were applied to acquire the information of the pixels of the target image. By doing so, color information and edge extraction were done, since it was the basic idea to achieve the segmented image from its background. Next, feature extraction has been done using DNNs through three stages (Low, middle and high level feature extraction). At low level feature extraction, the image edge and lines were extracted while image front or noses were extracted at middle level feature extraction. Then, all image features were combined at the high level feature extraction and thus all features were combined to form high-level features, because they were primitive images features. After doing all the above, detection has been done to locate the presence of objects in an image using bounding box regression model. Finally, for the SAR image recognition, the pre-trained CNN models such as ResNet-50, AlexNet, and VGG16 were used to compare their performance with the proposed model. In SAR image detection and recognition, the high training and testing time is founded as a challenging. Thus, to reduce such long training and testing time of SAR image detection and recognition, optimization algorithms such as Stochastic Gradient Descent with Momentum (SGDM), RMSProp and Adam optimization methods were used with the pre-training and proposed CNN model. **Findings:** Preprocessing was carried out using Median,

Guided Filter (GF), Lee, Box, Adaptive or Wiener filter algorithms were used and their performances were also compared in PSNR, SNR and MSE values and from those all used algorithms, the GF achieves better performance in high PSNR value of 37.8342. The performance of the all three pre-trained models and the proposed models were compared in accuracy and speed. The AlexNet, ResNet-50, VGG16 and proposed models achieved accuracy of 89%, 92%, 86% and 95% respectively and proposed model achieved the best performance. Among the used four models with Optimization methods, the Proposed Model with SGDM took very least time for Training (26' and 49s) and for Testing (17s). **Novelty:** New Deep Network Model was successfully designed, developed and used along with Optimization algorithms for reducing the training and testing time in SAR image detection and recognition.

Keywords: ResNet50; VGG16; SAR image; Optimization algorithms

1 Introduction

Optimization is a very important part of machine learning has attracted abundant attention of researchers. With the exponential growth of knowledge quantity and also the increase of model quality, optimization strategies in machine learning face additional challenges⁽¹⁾. Through optimization algorithms, long train and test time can be reduced. In this, the model structure and optimization algorithms (Model-based and optimization-based method) should have been taken in a consideration as it stated in the study of Cui and et al⁽²⁾. Accordingly, in this research work, two main scenarios were proposed based on problem or gap identified by Cui and et al⁽²⁾ which is described as follows:

1.1 Model-based strategic optimization scenarios

The CNN pre-trained models are firstly used for transferring learning or for fine tuning methods. In this to transfer learning from pre-training models, the most powerful method that helps to learn from pre-trained model is performed. With the pre-trained models the optimization algorithms was applied. After this, for the concepts of CNN structure role in the model, the new CNN model structure were proposed and it gives very good performance from the used others model. In this case, the model structure also has an impact with the optimization algorithms to reduce the training and testing time.

1.2 Optimization-based strategic scenarios

In the optimization based strategy there are different ways to choose parameters those helps to speed up the model training progress. Based on this, different optimization algorithms with parameter training options are practiced and the best parameter performances were used. From the attitude of the gradient data in optimization, in style optimization strategy is divided into 3 categories: First-order optimization strategies, High-order optimizations and heuristic derivative-free optimization strategies⁽¹⁾.

1.1 Stochastic Gradient Descent (SGD Optimization Algorithm)

In SGD, all parameters are updated with the same learning rate αt in the t^{th} iteration as

$$\theta_{t+1,i} = \theta_{t,i} - \alpha_t x g_{t,i} \quad (1-1)$$

Where $\theta_{t,i}$ and $\theta_{t+1,i}$ are the earlier and updated values for the i^{th} parameter with $i = 1, 2, \dots, d$, where d is that the number of parameters, and $g_{t,i}$ is the gradient with respect to the parameter $\theta_{t,i}$ for a loss function ϵ , defined as

$$g_{t,i} = \frac{\partial(\epsilon, \theta)}{\partial(\theta_{t,i})} \quad (1-2)$$

Where ϵ, θ may be a loss function with reference to the parameters of the network (θ) in t^{th} iteration. In this, the cross entropy loss used for image categorization experiments is defined as

$$\epsilon_{t,\theta} = \frac{1}{Nb} \sum_{j=1}^{Nb} \epsilon_{t,\theta,j} + \sigma R_{t,\theta} \quad (1-3)$$

Where Nb is the number of training images in the batch, $\epsilon_{t,\theta,j}$ is the cross entropy data loss for j^{th} training image in t^{th} iteration, $R_{t,\theta}$ is the regularization loss in t^{th} iteration, and σ may be a regularization loss hyper-parameter.

The annoyed entropy data loss $\epsilon_{t,\theta,j}$ for j^{th} training image is computed as:

$$\epsilon_{t,\theta,j} = -\log \left(\frac{e^{S_{oj}}}{\sum_{k=1}^{Nc} e^{S_k}} \right) \quad (1-4)$$

Where Nc is the total number of classes in the dataset, oj is the actual class (i.e., ground truth class) for j^{th} training image and S_k is the computed class score for k^{th} class for j^{th} training image.

The regularization loss $R_{t,\theta}$ is computed as

$$R_{t,\theta} = \sum_{i=1}^d (\theta_{t,i})^2 \quad (1-5)$$

1.4 Momentum

A very popular technique that used alongside SGD is named momentum. Instead of using only the gradient of the present step to guide the search, momentum also accumulates the gradient of the past steps to work out the direction to travel. Also bill that each gradient update has been determined into modules along with $w1$ (Weight 1) and $w2$ (Weight 2) direction. If individually sum these vectors up, their components along the direction $w1$ wipe out, while the component along the $w2$ direction is reinforced.

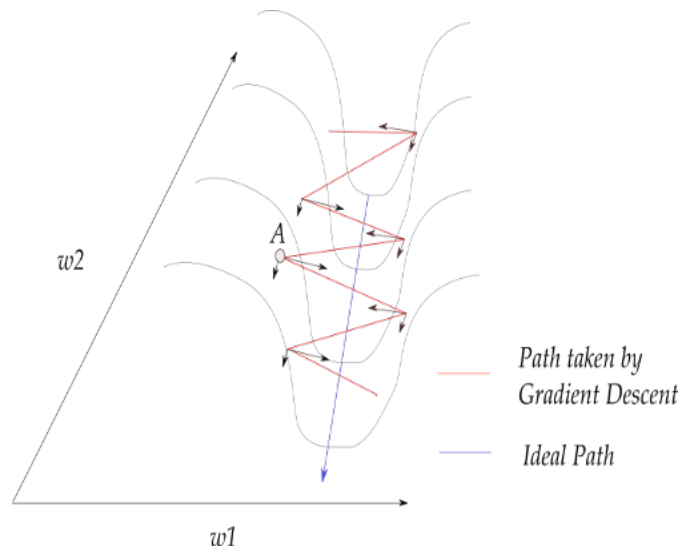


Fig 1. Momentum performance in the direction of gradient (Source: <https://blog.paperspace.com/intro-to-optimization-momentum-rmsprop-adam>)

For an update, this adds to the module along $w2$ (Weight 2), while zeroing out the component in $w1$ direction. This help us the move faster to the minima. For this reason, momentum is additionally mentioned as a way which dampens oscillations in

the search. In SGDM⁽³⁾, the gradient in each dimension is incorporated to realize moment for the parameters having consistent gradient, as follows: $m_{t,i} = \beta m_{t-1,i} + g_{t,i}$

$$\theta_{t+1,i} = \theta_{t,i} - \alpha m_{t,i}$$

Where $m_{t,i}$ is that the moment gained at t^{th} iteration for i^{th} parameter $\theta_{t,i}$ with $m_{t,i}=0$ for $t = 0$ and β may be a hyper-parameter to regulate the instant. In Adagrad⁽⁴⁾, the straightforward SGD style is revised by normalizing the learning rate α_t as:

$$\theta_{t+1,i} = \theta_{t,i} - \frac{\alpha_t \times g_{t,i}}{\sqrt{G_{t,i} + \epsilon}} \quad (1-6)$$

Where ϵ is a small value to avoid division by zero and $G_{t,i}$ is the sum of the squares of the gradients of t steps for the i^{th} parameter and given as:

$$G_{t,i} = \sum_{i=1}^t (g_{t,i})^2 \quad (1-7)$$

Where $g_{t,i}$ is given by Eq. (1-6). Over the iterations, the value of $G_{t,i}$ may become very large due to the positive accumulation of the square of the gradients and may decrease the effective learning rate α drastically, which successively can kill the training process.

1.5 RMSprop Optimization Algorithm

RMSprop can be compared very similarly to Adadelta. It attempts to extend Adagrad in a very similar way that Adadelta does. It maintains the per-weight learning rate while eliminating the decaying learning rate inherent in Adagrad. RMSprop preserves a "cache" of past weight values which decay over time given a decay parameter and accumulates the square gradient. The current gradient is divided by this "leaky" cache to modulate the learning rate per weight. RMSprop preserves a global learning rate parameter that Adadelta gets rid of. One prominent difference that is apparent based on the visualizations provided. RMSprop does not have the same initialization problem that Adadelta has. The problem addressed in Adadelta and RMSprop⁽⁵⁾ by leaking the accumulated square of gradients $G_{t,i}$ with a decay rate β is the $G_{t,i}$ in RMSProp is modified as:

$$G_{t,i} = \beta G_{t-1,i} + (1 - \beta) (g_{t,i})^2 \quad \text{Where}$$

$$G_{t-1,i} = 0 \text{ for } t = 1 \quad (1-8)$$

1.6 Adam Optimization Algorithm

Adam⁽⁶⁾ is another widely used gradient descent optimization technique that computes the learning rates at each step based on two vectors known as the 1st and 2nd order moments (i.e., mean and variance, correspondingly), which are recursively defined using the gradient and therefore the square of the gradient, respectively.

Here, the 1st and 2nd order moments are defined as:

$$m_{t,i} = \beta_1 m_{t-1,i} + (1 - \beta_1) g_{t,i} \quad (1-9)$$

$$v_{t,i} = \beta_2 v_{t-1,i} + (1 - \beta_2) g_{t,i}^2 \quad (1-10)$$

Where β_1 and β_2 are the decay rates for first and second moments, respectively, $m_{t-1,i}$ and $v_{t-1,i}$ are the mean and variance of the gradient of the previous steps, respectively. Both $m_{t-1,i}$ and $v_{t-1,i}$ are initialized with 0 at the first iteration, $t = 1$. It is observed that, initially, the value of first moment is small and the value of second moment is very small, thus leading to a very large step size. In order to solve this dispute, Adam has incorporated a bias correction of the 1st and 2nd order moments as:

$$\hat{m}_{t,i} = \frac{m_{t,i}}{1 - \beta_1^t} \text{ and } \hat{v}_{t,i} = \frac{v_{t,i}}{1 - \beta_2^t} \quad (1-11)$$

Where β_1^t is β_1 power t , β_2^t is β_2 power t , and $\hat{m}_{t,i}$ and $\hat{v}_{t,i}$ are the biased corrected first and second moments, respectively. Thus, the parameter update in Adam is assimilated as:

$$\hat{m}_{t,i} = \frac{m_{t,i}}{(1 - \beta_1^t)} \text{ and } \hat{v}_{t,i} = \frac{v_{t,i}}{(1 - \beta_2^t)} \quad (1-12)$$

2 Literature Review

Recently, many researchers were focused their thesis work on the optimization algorithm for reducing the training and testing time in SAR image detecting and recognition. Despeckling and colorization of SAR images simultaneously has been done, and they compare the performance of the method with that of other CNN methods (CNN⁽⁷⁾ and pix2pix⁽⁸⁾) and their combinations with the state-of-the-art despeckling algorithm SAR-BM3D⁽⁸⁾. Fully Convolutional Networks (FCNs) are a deep convolutional network that only consists of convolutional layers (and sub sampling) without any fully connected layer. It reduces the number of parameters by sharing weights and makes the learned features invariant to the location on the time-frequency plane of spectrograms, i.e., It provides advantages over hand-crafted and statistically aggregated features by allowing the networks to model the temporal and harmonic structure of audio signals⁽⁹⁾. SAR image classification based on unsupervised learning usually necessitates optimization of some metrics. Native optimization approaches frequently miscarry because functions of these metrics with respect to transformation parameters are generally non convex and irregular and, therefore, global methods are often compulsory⁽¹⁰⁾.

It is true that presenting the classifier with enough information is essential to achieving good performance, large training set sizes can be detrimental to generalization performance and invariably need significant training time. Such large training sets can often be redundant or noisy samples which only introduce unnecessary computations and could cause learning bias⁽¹¹⁾. Consequently, the deep neural net becomes more robust to appearance deviation of unseen new instances at the test set. In addition to the innovative application of the twofold classification and embedding loss to target recognition tasks, they also further improve the algorithm by inspiring the embedding loss at a later representation stage, in the classifier space, instead of the feature space that is commonly done in the Person-ReID⁽¹²⁾. The comparison of classification accuracy among different features shows that the baseline features and TPLBP (Three Patch Local Binary Pattern) features have good complementarities and the fused features have better segmentation⁽⁹⁾. More complete labeling might support better training, but it also can end in inferior training if the labeling used suboptimal parts. Automatic part labeling has the prospective to accomplish better performance by automatically finding effective parts. More elaborate labeling is also time consuming and expensive⁽¹³⁾. CNN has made amazing progress in image processing and has re-energized interest in ANNs. Up till now, a lot of research has been carried out to improve the CNN's performance on vision related tasks. The advancements in CNNs can be categorized in different ways including activation, and loss function, optimization, regularization, learning algorithms, and restructuring of processing units⁽⁴⁾. CNNs are well-organized models to execute nautical target classification in SAR images, and therefore the combination of various input resolutions within the CNN model improves its ability to derive features, increasing the overall classification score. In the study of Carlos Bentes, and et al perform on SAR input image is first calibrated to sigma zero (σ_0) and processed to detect all targets and obtain their centroids in image coordinates⁽¹⁴⁾. In the study of Li, et al, they designed a very elegant symmetric neural network named Deep U Net for pixel-level sea-land segmentation. Deep U Net is an end-to-end fully convolutional with two other kinds of short connections. These connections called U connections and Plus connections. In this, specifically the designed Down Block structure and the Up Block structure to adopt these connections. To verify the proposed network architecture, they collected dataset is of remote sensing Sea-land data RGB image sets Google-Earth and, those are manually labeled the ground truth. On the collected dataset, the Deep U net compared with SeNet and the Seg Net. Beside the experimental results show that the proposed Deep U Net outperformed the other networks significantly⁽¹⁵⁾. In⁽¹⁶⁾ the study compared with other algorithms and the proposed method simplifies the network structure and improves the recognition accuracy and time efficiency. The selection of features has a great impact on target recognition, in order to choose the features with more fusion value and further enhance the performance of recognition. The overall performance of the detection system will depend on both the quality of the change measure and the quality of thresholding. For addressing the aforementioned problems, a new measure is proposed by combining the fractal dimension and the intensity information of the original SAR images. To obtain a change map, the measure is partitioned into the changed and unchanged regions using some change detection methods, like Support Vector Machines (SVM), Fuzzy C-Means clustering (FCM) and artificial neural networks with a back propagation training algorithm⁽¹⁷⁾. As an early attempt to adapt CNN to pedestrian detection, the features generated by SCF+AlexNet were not so discriminative and produce relatively poor results. Based on multiple CNNs, Deep Parts and Comp ACT-Deep accomplish detection tasks via different strategies, namely local part integration and cascade network. The responses from different local part detectors make Deep Parts robust to partial blockings. However, due to complexity, it is too time-consuming to achieve real-time detection⁽¹⁸⁾. Evolutionary algorithms are computer-based solving technique, which use evolutionary computational models as vital element in their design and implementation⁽¹⁹⁾. The algorithm of synthetic aperture radar automatic target recognition (SAR-ATR) is generally calm of the withdrawal of a set of features that transform the raw input into a representation, followed by a trainable classifier⁽²⁰⁾. Feature extraction is a critical step for any automatic target recognition process, particularly in the interpretation of synthetic aperture radar (SAR) imagery⁽²¹⁾. The parameter identification of a polymer membrane fuel cell (PEMFC) is too important for designing, monitoring, and manufacturing. In most cases, its sensitive applications lead the researches to have

more attention to this subject⁽²²⁾. In this optimization-based method were used. SAR image speckle noises are image data that possess intensity related to the image and that also have an additive or multiplicative component. Accordingly, there were some filtrations algorithms are available to remove SAR image speckle noise⁽²³⁾. In this paper a variety of evolutionary algorithms have been proposed. They have a theoretical base of simulating the progression of individual structures via the Darwinian natural selection process. The training and testing time of SAR image detection and recognition is affected with very high training and testing time. And the optimization algorithms are respectively suggested for the problem of high train and test time on SAR image detection and recognition⁽¹⁾. Recently many researchers suggested this research gap especially on Optimization algorithms for reducing the training and testing time in SAR image detection and recognition⁽¹⁾.

3 Materials and Methods

The method or techniques used for the research work is a critical step that forwards the task of the proposed concepts to bring at the final step. Surprisingly methodology is a key for the research work. Methodology of this research works also clued very clearly as below in Figure 2.

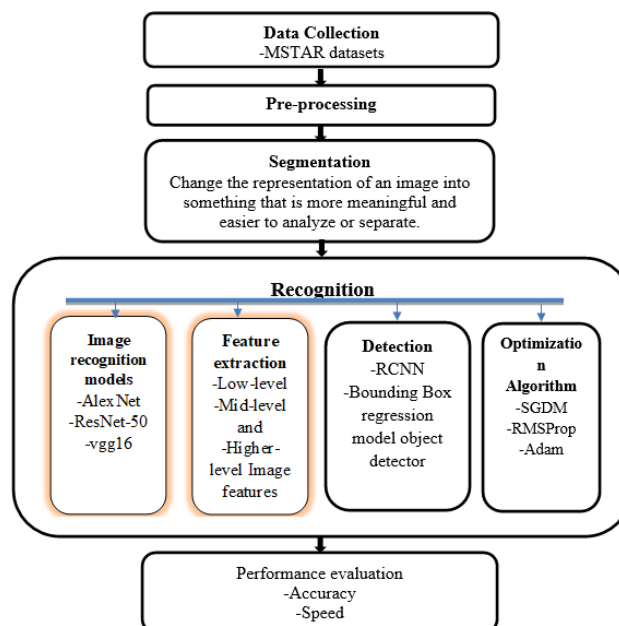


Fig 2. Architecture of research methodology

3.1 Data Collection

MSTAR database, is dataset that collected for the research work, which is provided by the Sandia National Laboratory (SNR) SAR sensor platform operating at X-band. The collection was cooperatively sponsored by Defense Advanced Research Projects Agency (DARPA) and Air force Research Laboratory as part of the Moving and Stationary Target Acquisition and Recognition (MSTAR) program.

Automatic Target Recognition (ATR) of Synthetic Aperture Radar (SAR) images is an area of continuing research by all branches of military and some research institutions⁽²⁴⁾. The available large database is very limited and the only publicly released database is MSTAR dataset, which only contained of several separate class of terrain military targets collected by an X-band SAR sensor⁽²⁵⁾.

Obvious that there are many mutual databases for close range digital photography, such as Image Net, PASCAL, and Label ME, etc. While for SAR, the available large database is very limited and the only public released database is MSTAR⁽²⁶⁾, which only contained of several separate class of terrain military targets collected by an X-band SAR sensor. The collected datasets was divided into training set and testing set, contain multiple types of ground military target, including BTR-60; tank: T-62 SLICY; air defense unit; ZSU-234; truck: ZIL-131; bulldozer: D7. The dataset contains 1746 of total target of SAR image data

and splitted in to training and testing set of 70% for training set and 30% for testing set respectively and the number of image observed in each class label of images are computed as shown in Table 1 below.

Table 1. Image number observed in each class labels

S/N	Label/Class	Count
1	BTR60	256
2	D7	299
3	SLICY	297
4	T62	299
5	ZIL131	299
6	ZSU234	296
	Total	= 1746

From the above Table 1 the dataset image labels of the minimum image set or fewest image number observed was 256. And to equalize all image numbers in each classes, it computed to minimum observed number images for preprocess dataset for the purpose of reducing redundancy and classification problems in the datasets and the computed results were shown in Table 2 below.

Table 2. Table of equalized image numbers to minimum set of image number observed for data preparation.

S/N	Label	Count
1	BTR60	256
2	D7	256
3	SLICY	256
4	T62	256
5	ZIL131	256
6	ZSU234	256
	Total	1536

Below Figure 3 shows corresponding sample MSTAR SAR images dataset used in this research work.

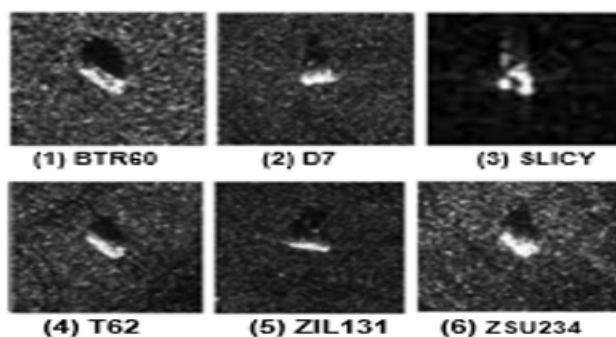


Fig 3. The SAR images used in the research work

3.2 SAR image Recognition models

SAR images recognition often proceeds in three stages: Feature extraction, detection, and recognition⁽²⁾. To talk about the recognition task, first image feature extraction has to be done. Although in deep learning, image feature extraction has three common procedures: low-level, middle-level and high-level feature extractions. In low-level image lines, shapes and edges are extracted. In middle-level images nose were extracted and in high level the images contextual information's are extracted. For the feature extraction the following template figures 6 bellows are gives more understandings. The recognition tasks of this thesis also proceeds as the followings. Consequently, let discuss some image recognition models as follows.

3.2.1 AlexNet Model

LeNet⁽²⁷⁾ though begin the history of deep CNNs but at that time CNN was limited to hand digit recognition tasks and not scaled to all classes of images. AlexNet⁽²⁷⁾ is deliberated as the first deep CNN architecture, which showed ground breach results for image classification and recognition tasks. Due to its competent learning approach⁽²⁶⁾ AlexNet has noteworthy importance in the new generation of CNNs and it has started a new era of research in CNNs. AlexNet model has 25 layers and required image sizes of [227x227x3].

3.2.2 ResNet-50 Model

ResNet was proposed by He et al.⁽²⁷⁾ which is considered as a continuation of deeper Nets, and it introduced an optimal methodology for the training of deeper Nets. The ResNet with 50/101/152 layers were more accurate than 34 layers plain Net. Worthy performance of ResNet on image recognition and localization tasks illustrated that depth is of central importance for many visual recognition tasks⁽²⁶⁾. ResNet-50 model has 177 layers and required image sizes of [224x224x3]. A residual network (ResNet) is 8 times deeper than VGG net but still having lower computational complexity compared to it⁽²⁸⁾.

3.2.3 VGG16 Model

With the successful use of CNNs for image recognition, Simonyan and Zisserman proposed an easy and effective design principle for CNNs. This new architecture was termed as VGG and was modular in layers pattern. Grounded on these outcomes, VGG substituted the 11x11 and 5x5 filters with a stack of 3x3 filters layer and experimentally confirmed that concurrent placement of 3x3 filters can induce the effect of the large size filter⁽²⁹⁾. Uses of small size filters give a further advantage of low computational complexity by reducing the amount of parameters. These findings set a replacement trend in research to figure with a smaller size filters in CNN.

The main drawback allied with VGG was high computational cost. VGG16 model has 41 layers and required image sizes of [227x227x3].

3.2.4 Proposed Model

The deep Convolutional Neural Networks (CNNs) gained a grand success on broad of computer visions. However, CNN model structures for training the data, were needs high CPU or GPU capacity of computer resources. The researcher, in this field was concerned on designing and developing CNN structures and optimization algorithm to reduce the training and testing time in SAR image detection and recognition. The two scenarios were proposed for the research works are basically model-based and optimization algorithms-based method from the Chui and et al study perspectives⁽²⁾. Accordingly, chromosome novel representation for the structure of CNN model was proposed and expected results were achieved. Unlike existing approaches, the proposed methodology was designed to work well in reducing the training and testing time in SAR image detection and recognition. It exploits advanced training methods to decrease the overhead on the computing resources that have been elaborated in the process. The experimental results denote the proposed model effectiveness over the related work methods. The model has 26 layers with the image input size of 128x128x3 as the architecture shown below from model input layer.

In the proposed model structure, the convolution layer, non-linear activation function (ReLU), pooling layer and batch normalizations layers are the basic CNN structures. Each convolution layer operation was used 5x5 which it is number of filter. Along with the MaxPool or pooling down the image size the Batch Norm layer was the most important layer that helps to increase the model performance. Accordingly, the proposed model were achieves best training and testing time reduction while using batch norm and maxpooling layers in the model structure. From this, using batch norm method shows that preferred model performance improvement in the proposed model structure. Although, resizing the input image size to 128x128x3 instead of 224x224x3 and 227x227x3 also gives more additional support in reducing the training and testing time. The purpose of ReLU was to announce non-linearity in the ConvNets, since most of the real-world data would want the ConvNets to learn would be non-linear and Convolution was a linear process element wise matrix multiplication and addition, so justification for non-linearity by introducing a non-linear function like ReLU has been used.

Batch normalization helps to initializing bias to zero or to other small constant random values and breaks the weight symmetry. Accordingly the down sampling method used was 3x3, which it was down sample size and gives additional benefit in the CNN structures of MaxPool Layer. So in this proposed model structure, the 3x3 maxpooling was used that gives more training time reduction. The softmax that helps to provide probability distribution of the given image to one of the label trained images. The final fc_6 is shows the number of class or labels used for the training model was six target class labels which are used in the model. As a general, the CNN structures starting from input layer to output layer were proposed and used as it has been shown below. Each and every layers are tends to their specific task to learn based on what input image given to the input layer of the proposed model. So, the proposed model has three convolutional layers followed by ReLU and batch norm layer instead

of dropout layer. From this, the batch norm layer used was found best performance rather than dropout layer in the network structure. With this, the input image size used also plays its own impact in the proposed model. The maxpool size of 3x3 were also used to sample down the image feature size in that it found best result in the new proposed model structure. In such a way that, the new proposed model with SGDM optimization method was achieved best training and testing time reduction than the used pre-training models.

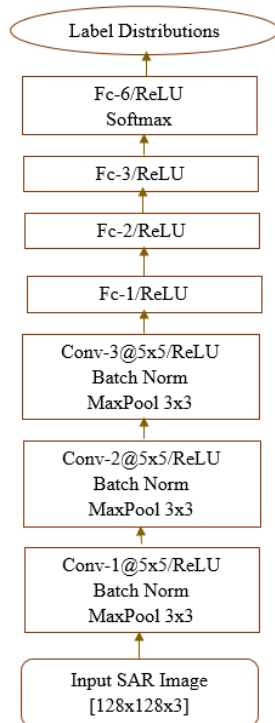


Fig 4. Architecture of the proposed model

The effectiveness of the proposed method has been evaluated on six different targets from public MSTAR SAR image dataset which are: armored personnel carrier: BTR60; tank: SLICY, T62; Air Defense unit: ZSU234; truck: ZIL131; bull dozer: D7 are used. Here, the below information was explained with Matlab for image data preparing for further detection and recognition. Managing image dataset for recognition tasks plays eyeful steps and it was done using IMDS (Image Data Store) method. IMDS was used to store the different categories of image with corresponding class label. To see number of images available within each category from IMDS, the count each label function was used. To maintain the equal number of images for the purpose of reducing the redundancy problems and classification problems in all categories in the IMDS, the min set count function was used to equalize the image numbers in each class labels. All image class label information was stored in IMDS which stores the image data properly. To find the location of image in various categories in IMDS, find method was used. Subsequently AlexNet or ResNet50 or Vgg16 Models or proposed model started. The input layer and output layers are specified with input image size and output label categories numbers for the models.

In IMDS the dataset was divided in to training and testing set by split each label function. In this training data set size was 70% and test set size was 30% as specified using split each label function. Many engineers mistakenly overwrite the images data source while processing. To solve this mistake; the method called augmented image data source store to resize and convert any gray scale image to RGB image totally image processing for further model understanding, on both training set and testing set were done. In computer vision, the feature vector was usually just the image pixel values. One special-purpose layer used commonly in computer vision was known as a “convolutional” layer. In its place of deriving optimal weights to multiply with all data point in the input vector, a convolutional layer derives an image kernel that it convolves with the input vector.

4 Results and Discussion

Within this research work, MATLAB tools were used, since MATLAB is easy or very accessible with deep learning tool box and MATLAB also accelerates deep learning.

4.1 Preprocessing result

After different filtration algorithm were used and compared with each other, the Guided filter algorithm achieves high PSNR value in SAR image speckle noise removal technique as shown in the Table 3 below.

Table 3. S4AR image Speckle noise removal algorithm comparison.

S/N	Filters	PSNR	SNR	MSE
1	Wiener Filter	14.7660	6.0111	2170.1087
2	Lee Filter	16.0543	4.5121	1613.0558
3	Box Filter	16.1029	4.5607	1595.1142
4	Median Filter	13.2904	4.5287	3021.9628
5	Guided Filter	37.8342	29.0716	10.7068

4.2 Segmentation result

The Figure 5 below shows the segmented and image surface edge detected results with bounding box for the process of detection of location of the target.



Fig 5. Segmentation method for image discrimination from its surface (a) original image (b) segmented image (C) segmented image with bounding box

4.3 Image feature extraction

Image features were extracted from image stored in IMDS and its information again stored as a matrix, and converted to gray scale image value to perform recognition task. For this work, the activation function was used. By stacking these layers together, a convolution neural network effectively implements a template-matching approach to recognize objects in an image, except that it creates hundreds of general templates and usually stacks multiple convolutional layers together as shown by below template. Early layers usually learn to recognize simple edges and lines (such as those shown in Figure 6) while later layers synthesize these results to recognize more complex concepts from the images.

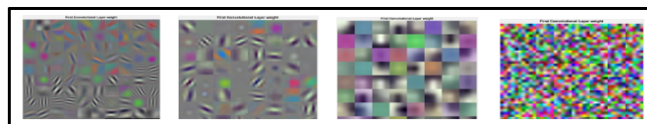


Fig 6. First Convolutional layer weights of (a) AlexNet, (b) ResNet-50, (c) vgg16 and (d) proposed model respectively

4.4 Accuracy assessment

A confusion matrix lists the values of known cover types of the reference data in the columns and of the classified data in the rows. The main diagonal of the matrix lists the properly classified pixels. One advantage of a confusion matrix is that it is easy

to see if the system is confusing two classes (i.e. commonly mislabeling one as another). A confusion matrix holds information about actual and predicted classifications done by a classification system on test data. Performance of such systems is usually evaluated using the data in the matrix. In this; the method called Error Correcting Output Code (ECOC) was used. These methods used $k(k-1)/2$ binary support vector machine binary classifier. Where k , is a number of unique class labels.

Generally, confusion matrix were done using AlexNet, ResNet-50, VGG16 and proposed models on testing data of 30% from the whole data used and accuracy of the models were done. Accuracy assessment is an essential step in the process of analyzing remote sensing data. Remote sensing products can serve as the basis for economic decisions. Potential users have to know about the steadfastness of the data when confronted with maps derived from remote sensing data. Accuracy expresses "exactness"; it dealings the agreement between a standard assumed to be correct and classified image of unknown quality. If the image classification agrees closely with the standard, it is said to be accurate. The most common way to express classification accuracy is the preparation of an error matrix also known as confusion matrix or contingency matrix. Diverse measures and statistics can be derived from the values in an error matrix. The basic form of an error matrix and non-statistical measures are described⁽³⁰⁾. In the following, to train CNN models the pre-trained Resnet-50, vgg16, AlexNet and proposed models were used. The classification method was performed on image class label and it's categorized under predicted images group. Now it shows here in the tables below result obtained from the computed confusion matrix done using Resnet-50, AlexNet, VGG16 and proposed models and the MAA (Mean Average Accuracy in %) of each class labels with trained models were shown as Table 4 below. For all models the used dataset was the MSTAR SAR image dataset where it explained in data collection methods.

Table 4. Mean Average Accuracy (MAA) of each class with the models.

Model	MAA (%)	BTR60	D7	SLICY	T62	ZIL131	ZSU234
AlexNet	0.89%	100	91	100	89	68	90
ResNet-50	0.92%	100	63	100	93	98	98
VGG16	0.86%	92	61	100	83	95	89
Proposed	0.95%	100	96	100	82	92	100

The performance of the all three pre-trained and proposed models was compared based accuracy. From those, the average accuracy of each classes were computed and the overall model accuracy performances were achieved. Accordingly, AlexNet, ResNet-50, vgg16 and proposed models achieved accuracy of 89%, 92%, 86% and 95% respectively. In this the proposed model were achieved better accuracy of 95%.

4.5 SAR image Detection

Detecting the target using bounding box regression model and RCNN (Region Based CNN) object detectors were used based embedded features representation. Convolutional Neural Network (CNN) is biologically inspired and multilayer deep learning models are trained end to end from raw image pixel values to the recognition outputs⁽⁴⁾.

The algorithm of synthetic aperture radar automatic target recognition (SAR-ATR) is generally calm of the withdrawal of a set of features that transform the raw input into a representation, followed by a trainable classifier⁽²⁵⁾. The detected image regions or chips, are then sent to the recognition stage, where a recognition algorithm predicts the target class label of each chip as shown in Figure 7 below. In this the sample2 appeared is used to provide rectangle value to bind the target or region of interest and there is sample1 which it is image label categories but it is not appeared as of sample2 and sample2 is dependent of this sample1.

Accordingly, the following detection results were done. In this input x is the image given to the model, m -d feature is massive number of images feature represented as $f(x)$, k -way classifier (wf) is ways in which extracted feature is going to classified, probability ($\hat{P} \propto e^{wf}$) is the probability in which the predicted and actual label is going to be determined and lastly the classified label is generated as of predicted output labels.

4.6 Optimization Algorithm for reducing the training and testing time

Batch normalization was employed to deal with the problems associated with internal covariance shift within feature maps. The internal covariance shift may be a change within the distribution of hidden units' values, which hamper the convergence (by forcing learning rate to small value) and requires careful initialization of parameters⁽²⁶⁾. Batch normalization unifies the distribution of feature map values by bringing them to zero mean and unit variance⁽³¹⁾. Performance evaluation of models with the used optimization methods, reduction of feature maps with max pooling method and parameter initialization methods was used to regulate computational complexity of training the network. Random estimate or randomization method with dataset

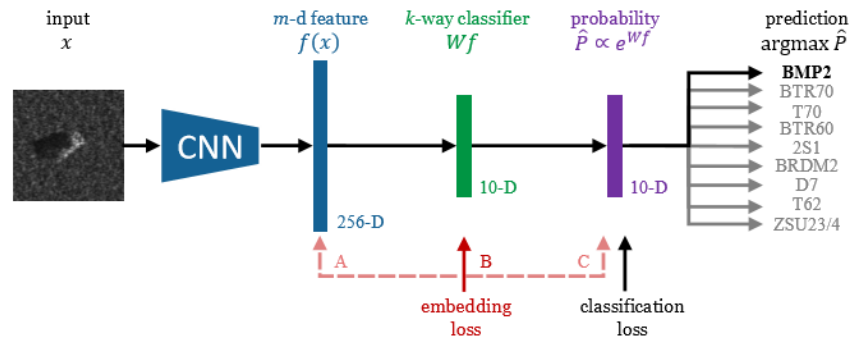


Fig 7. The input SAR image, x , into a series of embedded feature representations

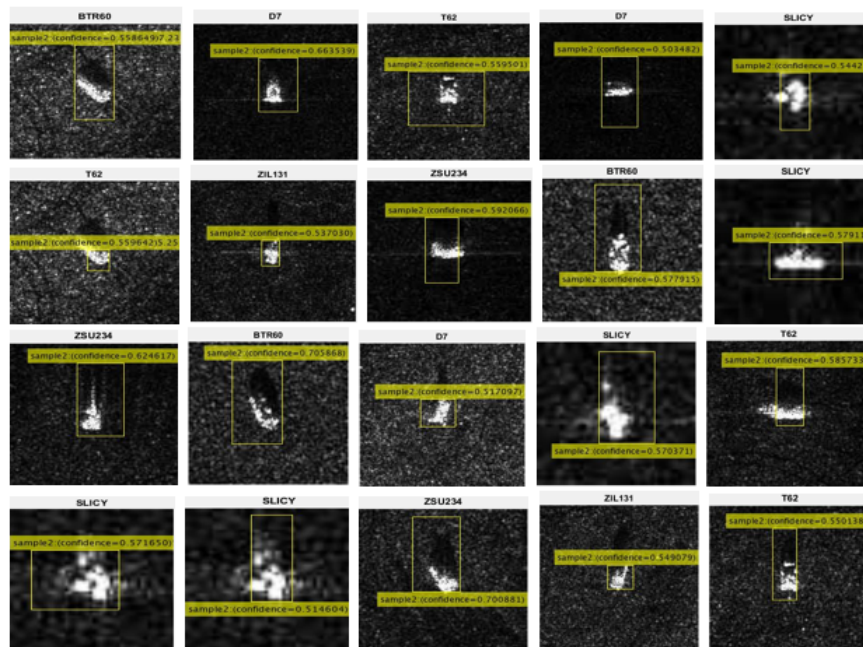


Fig 8. SAR image target detection with bounding box

for the model to train the data also gives additional support. With the optimization algorithms the parameter initialization was the core one. And in the model structure the image pooling size or sub-sampling method has major impact.

4.7 Performance evaluation of models with optimization

In the model training progress, Mini-Batch Gradient optimizer used was 32, faster linear learning of Beta Tolerance (InitialLearnRate) was = 0.001 and 1.0000e-04 as it seen on the model training progress, Momentum= 0.900, Learn rate drop factor=0.2, epoch=10 were used.

The following discussions shows that the performance comparison of the three pre-trained and proposed models used was compared in their accuracy and speed of training and testing time to recognize the SAR image using optimization algorithm method. To reduce high train and test time, optimization algorithms such as SGDM, RMSProp and Adam optimization algorithms with parameter initialization methods were used with the pre-trained model and from the scratch the proposed models were also used. The used models are AlexNet + SGDM, AlexNet + RMSprop, AlexNet + Adam, VGG16 + SGDM, VGG16 + RMSProp, VGG16 + Adam, proposed model + SGDM, proposed model + RMSProp, proposed model + Adam. The training time of AlexNet + SGDM: 32'and 4s and testing time was 24s, ResNet-50 + SGDM:174' and 5s and testing time was

27s, VGG16 + SGDM: 324' and 21s and testing time was 44s, proposed model + SGDM: 26' and 49s and testing time was 17s, AlexNet + RMSProp: 41' and 5s and testing time was 26s, ResNet-50 + RMSProp: 190' and 36s and testing time was 29s, VGG16 + RMSProp: 338' and 45s and testing time was 51s, proposed model + RMSProp: 30' and 14s and testing time was 18s, AlexNet + Adam: 40' and 48s and testing time was 27s, ResNet-50 + Adam: 177' and 24s and testing time was 36s, VGG16 + Adam: 444' and 2s and testing time was 1' and 33s, proposed model + Adam: 28' and 53s and testing time was 21s. In the network using larger pooling size generally results in worse performance, since it throws away too much information during sub sampling⁽¹⁾. Feature extraction was a critical step for any automatic target recognition process, particularly in the interpretation of synthetic aperture radar (SAR) imagery⁽¹⁾.

Table 5. Performance evaluation of models with SGDM optimization algorithm.

	AlexNet + SGDM		ResNet + SGDM		Vgg16 + SGDM		Proposed model + SGDM	
	Minutes	Seconds	Minutes	Seconds	minute	seconds	Minute	seconds
Train time	32	4	174	5	324	21	26	49
Test time	0	24	0	27	0	44	0	17

Table 6. Performance evaluation of models with RMSProp optimization algorithm.

	AlexNet + RMSProp		ResNet + RMSProp		Vgg16 + RMSProp		Proposed model + RMSProp	
	Minutes	Seconds	Minutes	Seconds	minute	seconds	Minute	seconds
Train time	41	5	190	36	338	45	30	14
Test time	0	26	0	29	0	51	0	18

Table 7. Performance evaluation of models with Adam optimization algorithm.

	AlexNet + Adam		ResNet + Adam		Vgg16 + Adam		Proposed model + Adam	
	Minutes	Seconds	Minutes	Seconds	minute	seconds	Minute	seconds
Train time	40	48	177	24	444	2	28	53
Test time	0	27	0	36	1	33	0	21

Note: Zero value under minute shows that the testing time was finished in seconds. Testing time was achieved after the model completes all the training image data and it were possible to test the image by sending the new image to the model. The time that it takes to test the new given image to the model with predicted image data for the recognition was going to be elapsed. Based on this the total time that it takes for testing time was achieved. Accordingly, the above Tables 5, 6 and 7 shows the training and testing time taken during the recognition of the SAR images.

5 Conclusion

In this research work some contributions have been presented that advances in the field of SAR image detection and recognition. The contributions are focused on optimization algorithm for reducing the training and testing time for SAR image detection and recognition based on optimization-based and model-based method. Along with this, the problems like over fitting have been limited with the method, image data augmentation and manage number of parameters. Data augmentation can help CNN in learning varied internal illustrations, which eventually leads to improve performance. Experiments are implemented with Matlab R2019a on a computer equipped with CPU of 3.2 GHz and 8 GB RAM memory. The experimental results consistently show that the used scenarios and algorithm with the parameter initialization method has improved the model performance and reduction of training and testing time in SAR image detection and recognition. Generally the optimization algorithms of SGDM with proposed CNN model structure were shows that preferred training and testing time reduction than others models with the used optimization algorithms. The parameter initialized with SGDM + proposed model were achieves expected result from those pre-trained models of AlexNet, ResNet-50, and vgg16 with SGDM, RMSProp and Adam.

6 Future Scope

In the future may consider further enhance the performance of detection and recognition, we will conduct further studies on an appropriate optimization algorithm with different deep learning models.

References

- 1) Sun S, Cao Z, Zhu H, Zhao J. A Survey of Optimization Methods From a Machine Learning Perspective. *IEEE Transactions on Cybernetics*. 2020;50(8):3668–3681. Available from: <https://dx.doi.org/10.1109/tcyb.2019.2950779>.
- 2) Cui Z, Dang S, Cao Z, Wang S, Liu N. SAR Target Recognition in Large Scene Images via Region-Based Convolutional Neural Networks. *Remote Sensing*. 2018;10(5):776–776. Available from: <https://dx.doi.org/10.3390/rs10050776>. doi:1.
- 3) Sutskever I, Martens J, Dahl G, Hinton G. On the importance of initialization and momentum in deep learning. *International Conference on Machine Learning*. 2013;p. 1139–1147. Available from: <http://proceedings.mlr.press/v28/sutskever13.html>.
- 4) Duchi J, Hazan E, Singer Y. Adaptive sub gradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*. 2011;12:2121–2159. Available from: <https://escholarship.org/uc/item/4ck5k544>.
- 5) Parks F. A study of the Exponentiated Gradient +/- algorithm for stochastic optimization of neural networks. *Powered by the California Digital Library University of California*. 2019. Available from: <https://escholarship.org/uc/item/4ck5k544>.
- 6) Chatfield K, Simonyan K, Vedaldi A, Zisserman A. Return of the Devil in the Details: Delving Deep into Convolutional Nets. *Proceedings of the British Machine Vision Conference 2014*. 2014. doi:10.5244/c.28.6.
- 7) Zhang K, Zuo W, Chen Y, Meng D, Zhang L. Beyond a Gaussian Denoiser: Residual Learning of Deep CNN for Image Denoising. *IEEE Transactions on Image Processing*. 2017;26(7):3142–3155. Available from: <https://dx.doi.org/10.1109/tip.2017.2662206>.
- 8) Girshick R, Donahue J, Darrell T, Malik J. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. *2014 IEEE Conference on Computer Vision and Pattern Recognition*. 2014;1:5000–5000. doi:10.1109/CVPR.2014.81.
- 9) Kang M, Ji K, Leng X, Xing X, Zou H. Synthetic Aperture Radar Target Recognition with Feature Fusion Based on a Stacked Autoencoder. *Sensors*. 2017;17(12):192–192. Available from: <https://dx.doi.org/10.3390/s17010192>.
- 10) Xiao-Dan Z, Xuan-Chi-Cheng L, Mei L. The Implementation of Wiener Filtering Deconvolution Algorithm Based on the Pseudo-Random Sequence. *American Journal of Circuits*. 2016;2(1):1–5. Available from: <http://www.aiscience.org/journal/allissues/ajcsp.html?issueId=70090201>.
- 11) Kabkab M, Alavi A, Chellappa R. DCNNs on a Diet: Sampling Strategies for Reducing the Training Set Size. 2016. Available from: <http://arxiv.org/abs/1606.04232>.
- 12) Wang J, Virtue P, Yu SX. Joint Embedding and Classification for SAR Target Recognition. 2017. Available from: <http://arxiv.org/abs/1712.01511>.
- 13) Felzenszwalb PF, Girshick RB, McAllester D, Ramanan D. Object Detection with Discriminatively Trained Part-Based Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2010;32(9):1627–1645. Available from: <https://dx.doi.org/10.1109/tpami.2009.167>.
- 14) Bentes C, Velotto D, Tings B. Ship Classification in TerraSAR-X Images With Convolutional Neural Networks. *IEEE Journal of Oceanic Engineering*. 2018;43(1):258–266. Available from: <https://dx.doi.org/10.1109/joe.2017.2767106>.
- 15) Li R, Liu W, Yang L, Sun S, Hu W, Zhang F, et al. DeepUNet: A Deep Fully Convolutional Network for Pixel-Level Sea-Land Segmentation. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*. 2018;11(11):3954–3962. Available from: <https://dx.doi.org/10.1109/jstars.2018.2833382>.
- 16) Ali W, Wegner AE, Gaunt RE, Deane CM, Reinert G. Comparison of large networks with sub-sampling strategies. *Scientific Reports*. 2016;6(1):24–29. Available from: <https://dx.doi.org/10.1038/srep28955>.
- 17) Aghababae H, Amini J, Tzeng YC. Improving change detection methods of SAR images using fractals. *Scientia Iranica*. 2013;20(1):15–22. Available from: <https://dx.doi.org/10.1016/j.scient.2012.11.006>.
- 18) Zhao ZQ, Zheng P, Xu ST, Wu X. Object Detection With Deep Learning: A Review. *IEEE Transactions on Neural Networks and Learning Systems*. 2019;30(11):3212–3232. Available from: <https://dx.doi.org/10.1109/tnnls.2018.2876865>.
- 19) Nedjah N, Mourelle LDM. Fast Pre-Processing for the Sliding Window Method Using Genetic Algorithms. *International journal of computers, Systems, Signals*. 2003;4(2):11–21. Available from: <http://citeseerx.ist.psu.edu/viewdoc/citations?doi=10.1.1.19.8327>.
- 20) Chen S, Wang H, Xu F, Jin YQ. Target Classification Using the Deep Convolutional Networks for SAR Images. *IEEE Transactions on Geoscience and Remote Sensing*. 2016;54(8):4806–4817. Available from: <https://dx.doi.org/10.1109/tgrs.2016.2551720>.
- 21) Kang M, Ji K, Leng X, Xing X, Zou H. Synthetic Aperture Radar Target Recognition with Feature Fusion Based on a Stacked Autoencoder. *Sensors*. 2017;17(12):192–192. Available from: <https://dx.doi.org/10.3390/s17010192>.
- 22) Guo H, Tao H, Salih SQ, Yaseen ZM. Optimized parameter estimation of a PEMFC model based on improved Grass Fibrous Root Optimization Algorithm. *Energy Reports*. 2020;6:1510–1519. Available from: <https://dx.doi.org/10.1016/j.egy.2020.06.001>.
- 23) Nagesa Y, Nagarajan S, Negesa F. Performance Comparison of SAR Image Speckle Noise Removal Algorithms. *International Journal of Computer Applications*. 2021;183(18):14–19. Available from: <https://dx.doi.org/10.5120/ijca2021921525>.
- 24) Sun Y, Wang X, Tang X. Deep Learning Face Representation from Predicting 10,000 Classes. *2014 IEEE Conference on Computer Vision and Pattern Recognition*. 2014;p. 1891–1898. doi:10.1109/CVPR.2014.244.
- 25) Wang Y, Han P, Lu X, Wu R, Huang J. The Performance Comparison of Adaboost and SVM Applied to SAR ATR. *2006 CIE International Conference on Radar*. 2006. doi:10.1109/ICR.2006.343515.
- 26) Khan A, Sohail A, Zahoor U, Qureshi AS. A survey of the recent architectures of deep convolutional neural networks. *Artificial Intelligence Review*. 2020;53(8):5455–5516. Available from: <https://dx.doi.org/10.1007/s10462-020-09825-6>.
- 27) Chuang KS, Tzeng HL, Chen S, Wu J, Chen TJ. Fuzzy c-means clustering with spatial information for image segmentation. *Computerized Medical Imaging and Graphics*. 2006;30(1):9–15. Available from: <https://dx.doi.org/10.1016/j.compmedimag.2005.10.001>.
- 28) Donahue J. DeCAF: A deep convolutional activation feature for generic visual recognition. *31st International Conference of Machine Learning*. 2014;2:988–996. Available from: <http://proceedings.mlr.press/v32/donahue14.html>.
- 29) Symonian K, Zisserman A. Very deep convolutional networks for large-scale image recognition. 2015. Available from: <https://arxiv.org/abs/1409.1556>.
- 30) Pang S, Meng F, Wang X, Wang J, Song T, Wang X, et al. VGG16-T: A Novel Deep Convolutional Neural Network with Boosting to Identify Pathological Type of Lung Cancer in Early Stage by CT Images. *International Journal of Computational Intelligence Systems*. 2020;13(1):771–771. Available from: <https://dx.doi.org/10.2991/ijcis.d.200608.001>.

- 31) Lakide V. Classification Of Synthetic Aperture Radar Images Using Particle Swarm Optimization Technique Classification. 2009. Available from: <https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.330.6185>.