Analysis of Deep Neural Networks for Military Target Classification using Synthetic Aperture Radar Images

Shan Jacob
Computer Science and Digital Technologies, UEL
London, United Kingdom
u2235626@uel.ac.uk

Julie Wall
Computer Science and Digital Technologies, UEL
London, United Kingdom
j.wall@uel.ac.uk

Mhd Saeed Sharif
Computer Science and Digital Technologies, UEL
London, United Kingdom
s.sharif@uel.ac.uk

Abstract—**Target detection and classification in the military is an area that is very significant in modern battlefields. Using Synthetic Aperture Radar images for classifying targets adds to its significance, as these images are high-resolution images of the surface of the earth created using microwave radiation and they can be used anytime, anywhere, and in any weather conditions. A target classification system using deep learning to classify military vehicles from Synthetic Aperture Radar images is proposed in this study. The system uses a baseline Convolutional Neural Network to classify the images of military vehicles from the MSTAR dataset, achieving a baseline accuracy of 90%. Further transfer learning was applied to the system by using 5 different pre-trained networks, namely the InceptionV3, VGG16, VGG19, ResNet50, and MobileNet. These models were analysed and evaluated using 3 different evaluation metrics, the Confusion matrix, Classification report, and Mean Average Precision to discover the most accurate and efficient model for this task. The models VGG16 and MobileNet displayed the best performance on the dataset achieving accuracies of 98% and 97%, respectively. The ResNet50 model displayed the worst performance among the models, achieving an accuracy of 82%. While the other models, InceptionV3 and VGG19, achieved accuracies of 92% and 96% respectively.**

Keywords— SAR, Target Classification, Deep Learning, Transfer learning, MSTAR dataset

I. INTRODUCTION

Target classification plays a crucial role in modern warfare as it enables the detection and identification of enemy assets [1]. Synthetic Aperture Radar (SAR) images, created using microwave radar signals, provide high-quality remote sensing images of the Earth's surface, making them valuable for military applications where optical imagery is limited The SAR is a technology that can produce images disregarding whatever time of the day it was taken, what the weather condition was, and the amount of lightning available and this makes it a very unique technology [2] SAR works by transmitting microwave radar signals to the Earth's surface and receiving the reflected signals [3]. By calculating the time delay and phase difference between transmitted and received signals, SAR can determine the position and distance of objects on the Earth's surface, creating detailed terrain photographs. SAR images capture the shape, size, and structure of potential target objects, including various polarizations, which aid accurate target classification and polarization discrimination [4].

However, SAR data analysis and processing can be challenging, due to factors such as noise (electrical, thermal, and speckle), coherence, complexity, and the vast volume of data [5]. Conventional techniques such as manual processing of these images using groups of humans, may struggle to interpret and analyse SAR data effectively, resulting in less accurate results and increased computational demands. Deep learning networks like CNNs have shown promise in mitigating these challenges. Deep learning models excel at handling large datasets, leveraging parallel processing capabilities, and automatically extracting features and patterns from SAR data [7].

Deep learning models offer numerous advantages in SAR image processing tasks. They can automatically extract relevant information, eliminate manual image processing, produce highly accurate results, interpret SAR images quickly, and scale up to handle larger datasets and more complex analysis tasks. Tasks like image classification, change detection, segmentation, and registration can be achieved, making deep learning models a flexible tool for SAR picture analysis [8].

Deep learning utilizes artificial neural networks, comprising multiple layers of connected nodes, to solve complex challenges. These models identify patterns and features in data without the need for manual extraction, making them suitable for various applications such as natural language processing, autonomous driving, speech recognition, and image recognition [7]. Training deep learning models typically requires large labelled datasets, where the model adjusts its biases and weights during training to minimize prediction errors. Once trained, the model can predict new data.

In this paper, target detection from SAR images using deep learning has been achieved using the MSTAR dataset, a dataset created by the US Department of Defense. MSTAR primarily focuses on target recognition and classification, featuring eight Russian military vehicles, including bulldozers, tanks, trucks, guns, and armoured carriers.

This study aims to aid the military in classifying enemy targets in SAR images by investigating the application of deep learning models, particularly CNNs, to this problem. Transfer learning techniques will be employed by utilizing various pre-trained models to determine the optimal model for military object classification.

In summary, this study proposes the classification of military objects in SAR images from the MSTAR dataset using a baseline CNN model. By leveraging transfer learning techniques and comparing the performance of six deep neural networks, including InceptionV3, VGG16, VGG19, ResNet50, and MobileNet, the study aims to determine the most effective model for military target classification. Evaluation metrics, such as mean average precision (mAP), classification reports, and confusion matrices, will be utilized to assess model performance, providing insights into the potential of deep learning for SAR image analysis and military target detection.

II. Literature survey

Soldin applied the ResNet-18 deep neural network on the MSTAR dataset and obtained achieved an accuracy of 99% on 10 classes of the MSTAR dataset using the confusion matrix [9]. They further extended the classifier on emerging targets and investigated its effect.

Gu, Tao, Feng and Wang developed a system using the VGG16 network on the MSTAR dataset to classify three categories of military vehicles, the BTR70 (armored transport vehicle), BMP2 (infantry fighting vehicle), and T72 (tank) using the VGG16 neural network [11]. They achieved an accuracy of 90% for both BMP2 (infantry fighting vehicle) and T72 (tank) while they achieved a low accuracy of 70% for the BTR70 (armoured transport vehicle), as the vehicles have similar appearances and can be confused with each other.

For observation of areas important for the military, Anishi Gupta and Uma Gupta developed a system that can survey the area in real-time [14]. Here they made use of a customized You Only Look Once (YOLO) model which has an enhanced CNN with 58 layers. They created a customized dataset for this model, which contains 22 classes of objects, 20 classes of the Pascal VOC dataset, and 2 classes of tanks and guns which they gathered from around the internet. Their model achieved an accuracy of 79.12% and 78.19% mAP. The paper suggests future improvements such as incorporating instance segmentation and masking to further enhance the precision and recall of the detection system.

Ouyang, Wang, Hu, Xu, and Shao proposed a method for vehicle detection in the military on the basis of hierarchical feature representation and reinforcement learning refinement localization [15]. They constructed a new dataset of military vehicle images sourced from the internet for the object detection task. The test set in the dataset was separated into 3 divisions samal scale, large scale, and full dataset. To evaluate, they used mAP and achieved an accuracy of 85.6% on the large scale division, 66.3% for the small scale and 81.1% for the full set. This method can support observation and monitoring of arms and other supplies in warfare based on information.

Coman and Thaens [12] investigated the use of phase information in deep learning models for SAR target classification using the MSTAR dataset. Their study demonstrated that incorporating phase information improved the accuracy, achieving 91% accuracy when both phase and amplitude were utilized, and 90% accuracy when using amplitude alone. The results emphasize the significance of phase information in enhancing the classification performance for SAR targets. The authors suggest two future research directions: exploring advancements in ensemble models, capsule networks, and transfer learning techniques, and enhancing the dataset for training by incorporating spectral images, polarimetry images, and other relevant data.

Shi [10] developed a multi-feature fusion-based approach for SAR target recognition using the MSTAR dataset. The proposed method integrated Hu moment (image moments for shape analysis), Harris corner point (corner detection algorithm), and Gabor features (linear filters used for texture analysis) to capture target shape, corner features, and texture. By addressing the limitations of single-feature descriptions, the approach achieved higher recognition rates. PCA (Principal Component Analysis) dimension reduction was applied to reduce data dimensionality, and three conventional classification techniques, Decision Trees, Support Vector Machines and the Multilayer Perceptron were employed. The integration of single and multiple features led to an enhancement of 2.4-2.9% in the accuracy of target recognition. Among the various classifiers utilized, the Decision Tree algorithm attained the highest recognition accuracy of 92.9237%. In the case of SAR target recognition, employing PCA for multi-feature fusion and reducing the dimensionality to 40% resulted in the highest recognition rates for all three classifiers, with the Decision Tree classifier achieving a recognition rate of 94.4134%. Overall, the study showcased the effectiveness of multi-feature fusion and PCA dimension reduction in SAR target recognition.

Bandarupally, Talusani, and Sridevi proposed a method that combines Edge Boxes and CNNs for object detection, followed by image super resolution using Dense-Skip-Connections. The model effectively detects target patches and generates high-resolution outputs. The dataset used consists of approximately 500 annotated images, and the model's performance is evaluated by generating high-resolution target patches from the satellite images. The results demonstrate that the proposed model produces finer and clearer outputs compared to traditional methods such as bicubic interpolation. The paper concludes by highlighting the potential for further improving the system's performance in future work.

The existing research on military target detection and classification using deep learning and SAR images has showcased promising results and advancements in the field. Several studies have achieved high accuracy in classifying different categories of military vehicles using various deep learning models. The proposed methods have utilized techniques such as hierarchical feature representation, reinforcement learning, and multi-feature fusion to enhance detection and recognition performance. These approaches have demonstrated improved accuracy, especially when incorporating phase information in SAR target classification. However, some challenges remain, such as misclassification of similar-looking vehicles and the need for further improvement in precision and recall of detection systems.

This research aims to contribute to the field of military target detection by proposing a deep learning-based approach that leverages the strengths of existing methodologies while addressing their limitations. The focus will be on developing an efficient and accurate model for classifying military targets in SAR images. Transfer learning techniques will be utilized to evaluate and compare the performance of different deep learning models, including InceptionV3, VGG16, VGG19, ResNet50, and MobileNet. Evaluation metrics such as mAP, classification reports, and confusion matrices will be employed to assess model performance and determine the most effective model for military target classification. The research aims to provide insights into the potential of deep learning in SAR image analysis and contribute to advancements in military target detection systems.

III. Methodology

A. Dataset

The MSTAR (Moving and Stationary Target Acquisition and Recognition) [9] dataset is a collection of SAR images, which is widely used for benchmarking and evaluating SAR target recognition algorithms. The dataset was originally developed by the US Defence Advanced Research Projects Agency (DARPA) between the years 1995-1996. The dataset is mainly used for target recognition and classification. The dataset consists of 8 Russian military vehicles like Bulldozer, tanks, trucks and howitzers namely, the D7, T62, ZIL-131, ZSU-23-4, SLICY, 2S1, BRDM-2, and BTR-60.

SAR operates in the X-band microwave frequency (8 to 12 GHz) with HH (Horizontal-Horizontal, where radar waves are transmitted and received with a horizontal orientation.) polarization and uses a spotlight SAR working mode (where the radar beam is focused on a small area of the Earth's surface). The SAR system has a high resolution of 0.3 meters by 0.3 meters and uses a pixel size of 128 by 128. For each target category in the MSTAR database, the azimuth angle varies uniformly from 0 to 360 degrees. This enables fair performance evaluation of SAR target recognition algorithms across different viewing angles.

The MSTAR dataset includes SAR images with varying resolutions, aspect angles, and polarizations, as well as images with different operating frequencies and bandwidths. The dataset also includes images with various levels of noise and clutter, which makes it a challenging benchmark for SAR target recognition algorithms. The MSTAR dataset has been used extensively in research on SAR target recognition, and several benchmark studies have been conducted to compare the performance of different algorithms on the dataset. The dataset has also been used to evaluate the effectiveness of data augmentation and deep learning approaches for SAR target recognition.

B. Architecture

CNNs are a typical deep learning architecture type utilised in computer vision problems [16]. It is intended to automatically recognise and extract features from grid-like data, such as photographs. Convolutional, pooling, and fully linked layers are among the many layers that make up CNNs. Convolutional layers employ convolutions to extract local patterns and features by applying filters or kernels to the input data. By pooling layers, feature maps' spatial dimensions are reduced, which speeds up computation and extracts the most important features. High-level feature learning and classification are made possible by fully connected layers, which connect all neurons from the previous layer to the next one. CNNs are useful for a variety of visual identification applications, including image classification, object detection, image segmentation, and others. They are highly suited for analysing complicated visual data because they can automatically learn spatial hierarchies of features. CNNs have successfully transformed computer vision applications in a number of fields, including self-driving automobiles, medical imaging, and facial identification.

Deep learning architectures like VGG16, VGG19, InceptionV3, MobileNet, and ResNet were also used to perform transfer learning which is a machine learning approach that involves leveraging a pre-trained model as a foundation for training a new model on a different task. [19]. Frequently employed for a range of computer vision problems, the deep network structures of VGG16 and VGG19[17], which have small convolutional filters, are renowned for offering precise feature extraction. InceptionV3 makes effective use of inception modules to efficiently acquire data on a multi-scale. For mobile and embedded devices, MobileNet focuses on lightweight and effective models. ResNet introduces residual connections to solve the vanishing gradient issue in very deep networks, allowing enabling training of extremely deep models with improved performance [18].

IV. Experiments

Firstly, 80% of the MSTAR dataset was used for training the models and the remaining 20% was used for validating the models. The size of the images to the models was set to a tuple of (224,224). This was done for easier transfer learning with pre-trained models. The batch size was set to 32 images in each batch.

Then a custom CNN model was created using Keras Sequential API. The first layer in the custom CNN model was a convolutional layer that consists of 32 filters, each with a size of 3x3. The activation function used is ReLU. After the convolutional layer, the subsequent layer employed max pooling with a pool size of 2x2. This pooling operation reduced the dimensional extent of the feature maps generated by the convolutional layer. The second layer of the network is equipped with 64 filters, each having dimensions of 3x3. Following this layer is a max pooling layer of 2x2 sized pool. The third layer is a convolutional layer with 128 filters of size 3x3 with ReLU activation functions followed by a max pooling layer. After the third max pooling layer, a GlobalAveragePooling2D layer was used instead of the flatten layer to reduce the spatial dimensions of the feature maps to a single value by averaging the values of each feature map. The output is passed to two fully connected (dense) layers, with ReLU activation functions, each layer having 1000 and 500 units, respectively. Finally, the softmax activation function is employed to generate a probability distribution across the classes as the output. The model is compiled utilizing the Adam optimizer and employs categorical_crossentropy as the chosen loss function.

The pre-trained deep learning models, vgg16, vgg19, InceptionV3, ResNet50, and MobileNet were trained on the dataset. The models were loaded with pre-trained weights from the ImageNet dataset using the Keras deep learning framework and the fully connected layer is excluded. To prevent the pre-trained weights from being updated during the training, the trainable attribute of the loaded models is set to False. An input layer is defined with the desired input shape and connected to the output of the pre-trained models. To reduce the spatial dimensions of the feature maps, a global average pooling layer is included, followed by the addition of fully connected layers for classification purposes. The Adam optimizer, categorical cross-entropy loss function, and accuracy metric were used for the optimizer, loss function, and evaluation metric, respectivey. Finally, the fit method was called to train the models using the training data generator and validate the efficiency of the models on the test data generator. The models underwent training for a total of 20 epochs, utilizing a learning rate of 0.001 and a batch size of 32.The performance of the models was evaluated using 3 evaluation metrics, namely the confusion matrix, the classification report, and the mean average precision.

V. Results and Discussion

Comparing the models, we can observe that all the models used in the experiment, except the ResNet50 model, performed well for the MSTAR dataset, see Table 1.

Table 1. Performance of Deep learning models

Models	Accuracy	Loss	Weighted average F1 score	Time for training per epoch (minutes)	Time for prediction (minutes)	Mean average Precision
CNN	0.9	0.2	0.91	6	2	0.92
InceptionV3	0.92	0.3	0.92	7	2	0.92
VGG16	0.98	0.04	0.99	30	6	0.98
VGG19	0.96	0.08	0.97	35	6	0.96
ResNet50	0.82	0.5	0.81	12	2	0.84
MobileNet	0.97	0.09	0.98	3	0.46	0.97

The basic CNN model showed an overall high performance in the experiment by correctly classifying all the samples for 3 classes. The model showed poor performance for the 2S1 and D7 classes by misclassifying a significant number of samples in each of the classes. The model made only a few misclassifications for the rest of the classes while achieving a low precision score for the T62 class.

The VGG16 model displayed the best performance out of all the models considering the evaluation metrics used. The model provided correct predictions for most of the classes, misclassifying very few samples and achieving high precision and recall values. The model achieved a high accuracy of 0.98 and a high mAP value of 0.98.

The VGG19 model displayed similar results to the VGG16 model The model displayed correct predictions for most of the classes while struggling with class D7. The model achieved high precision and recall values for most classes while scoring a lower recall value for class D7. The model scored a high accuracy and mAp score of 0.96. However, the VGG19 model was very slow to train and for inferencing, taking around 35 minutes per epoch for training and 6 minutes for testing.

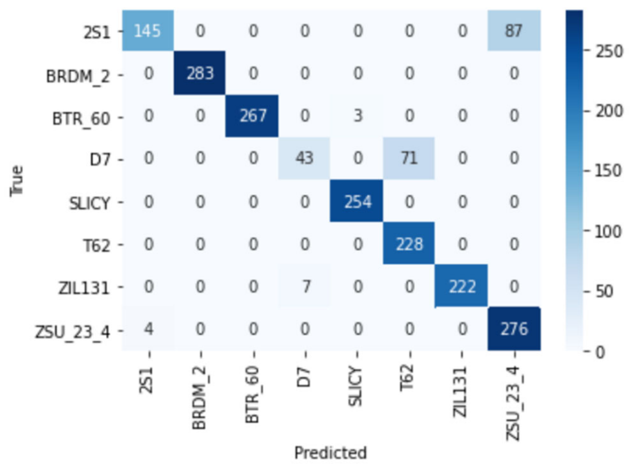

Fig. 2. CNN confusion matrix

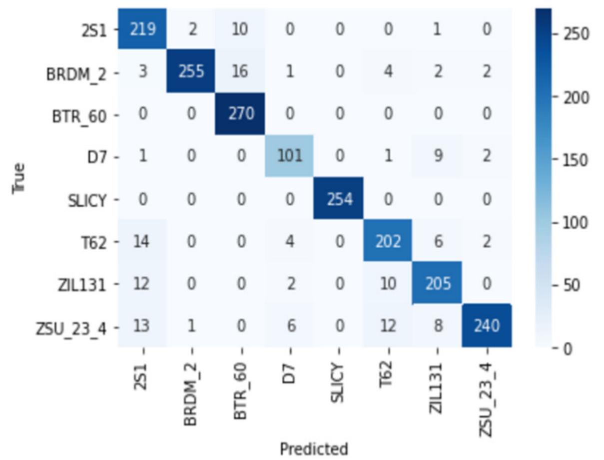

Fig. 3. InceptionV3 confusion matrix

The InceptionV3 model showed better performance on the dataset for all the classes, correctly classifying more classes with higher precision and recall values.

The ResNet50 model displayed the worst performance in the experiment in comparison to the other models. This model showed significant misclassifications for most classes, performing the worst on the 2S1 and the D7 classes. The model achieved a lower accuracy of 0.82 compared to the other models and a mAp value of 0.84.

The MobileNet model is another model that showed great performance on the MSTAR dataset. The model correctly predicted almost all the samples of all the classes achieving very high precision and recall values, similar to the VGG16 model. The model achieved a very high accuracy and mAP value of 0.97 each. It is also very important to note the time the model took for training and testing, which was 3 minutes per epoch for training and 46 seconds for testing. The model took the least time for training and prediction compared to all the other models.

In summary, all the models performed well on the MSTAR dataset except the ResNet50 model. This may be due to the high number of layers in the ResNet50 model. From the experiments, we can observe that the performance of the models on the dataset decreases with the higher number of layers of the models. This might be due to the small size of the dataset. All of the models performed the best for the SLICY class as all the models made correct predictions on all the samples in this class while achieving a precision and recall score of 1 each. The classes that all the models displayed the least performance on were the 2S1 and D7 classes which might be due to the smaller size of samples available for testing for the D7 and the similarity in appearance for 2S1 with the ZSU_23_4.

The model that displayed the highest performance was the VGG16 achieving high values in all the evaluation metrics. But comparing the time taken for training and testing it can be observed that the VGG16 is very slow. Thus, comparing the overall performance of the model on all the evaluation metrics and the time taken for training and testing (3 minutes for training and 46 seconds for testing) it can be concluded that the MobileNet model displayed the best performance on the MSTAR dataset that was used.

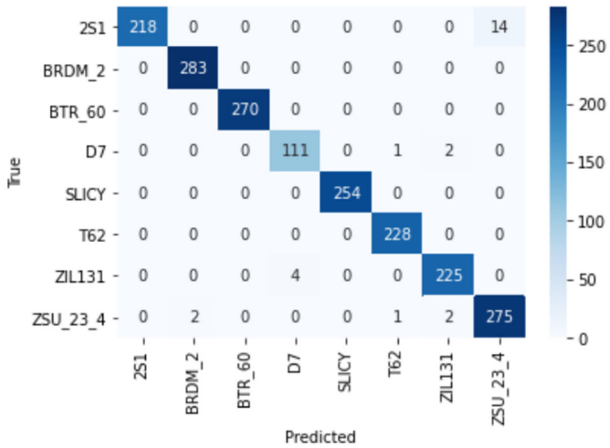

Fig. 4. VGG16 confusion matrix

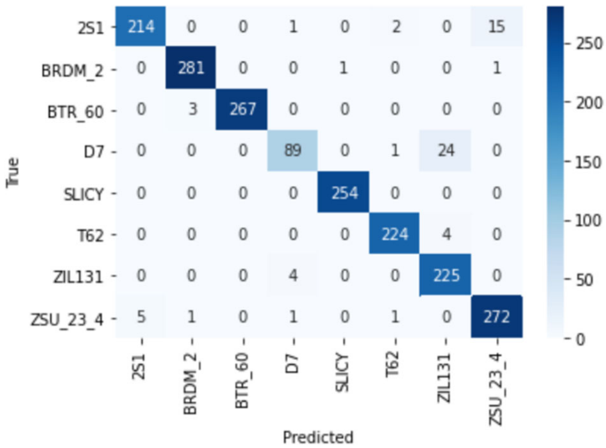

Fig. 5. VGG19 confusion matrix

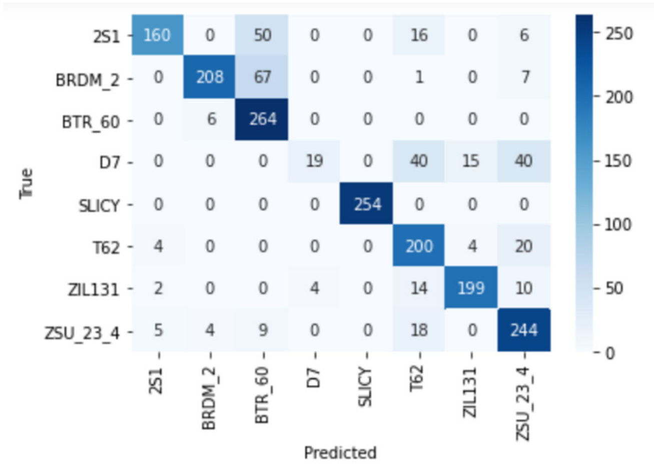

Fig. 6. ResNet50 confusion matrix

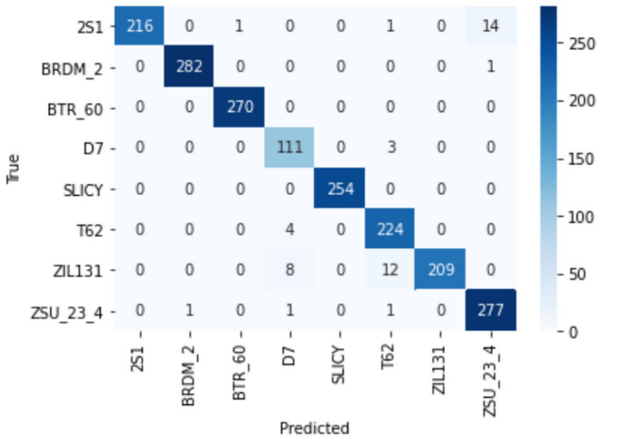

Fig. 6. MobileNet confusion matrix

VI. Conclusion

In modern warfare, it has become crucial to accurately identify and track targets. SAR systems can create high-resolution images of the surface of the Earth by making use of microwave radiation and can be used in all types of weather conditions, day or night. They can also access areas that are remote, densely forested, and under cloud cover. Using SAR images for detecting enemy targets in the military can help to detect and classify enemy targets in broad daylight and at night, in all kinds of weather conditions. They can also be used to find enemy targets hiding undercover in forests, remote areas, and cloud cover.

A deep learning system using a baseline CNN for classifying military targets using the MSTAR dataset was created for the automatic classification of military targets. The system can correctly classify the 8 different military vehicles available in MSTAR dataset. Further transfer learning was applied to the system by experimenting with the dataset on 5 different pre-trained models namely, the InceptionV3, VGG16, VGG19, ResNet50, and MobileNet. The performances of all the models were compared using 3 different evaluation metrics which are confusion matrix, classification report, and mAP to find the more accurate and efficient target detection algorithm on the dataset. All the models displayed good performance on the model, whie the best performance was achieved while using VGG16 and the MobileNet models. Considering the time taken for training and prediction, it was concluded that the best model for the task was the MobileNet model.

VII. References

[1] Jensen, J.R. (2016). Introductory Digital Image Processing: A Remote Sensing Perspective. Routledge.

[2] Guttman, C. (2022) Satellite revolution empowered by cloud watches for dangerous weather, The Forecast By Nutanix. Available at: https://www.nutanix.com/theforecastbynutanix/industry/satellites-using-synthetic-aperture-radar-to-see-through-clouds-and-predict-floods (Accessed: 09 May 2023).

[3] Editor (2021) Deep learning and the future of Artificial Intelligence, AltexSoft. Available at: https://www.fcc.gov/sites/default/files/foia-consumer-complaints-09142017-565-577-privacy-1.pdf (Accessed: 08 May 2023).

[4] Njambi, R. (2022) How SAR data is complementary to optical, UP42 Official Website. Available at: https://up42.com/blog/sar-data-complementary-optical optical (Accessed: 09 May 2023).

[5] Deep Block (2023) How AI can help overcome SAR imagery analysis challenges., how-ai-can-help-overcome-sar-imagery-analysis-challenges. Available at: https://www.linkedin.com/pulse/how-ai-can-help-overcome-sar-imagery-analysis-challenges (Accessed: 09 May 2023).

[6] Kumar, V. et al. (2022) Agricultural SANDBOXNL: A national-scale database of parcel-level processed sentinel-1 sar data, Nature News. Available at: https://www.nature.com/articles/s41597-022-01474-4 (Accessed: 09 May 2023).

[7] Deep Learning (no date) What Is Deep Learning? Available at: https://uk.mathworks.com/discovery/deep-learning.html

[8] Soenen, S. (2019) Deep learning and sar applications, Towards Data Science. Available at: https://towardsdatascience.com/deep-learning-and-sar-applications-81ba1a319def

[9] J. Soldin, R. (2018) SAR Target Recognition with Deep Learning. tech. IEEE.

[10] Shi, J. (2022) SAR target recognition method of MSTAR data set based on multi-feature fusion. rep. Beijing: IEEE.

[11] Gu, Y. et al. (2021) Using VGG16 to Military Target Classification on MSTAR Dataset. rep. ShangHai: IEEE.

[12] Coman, C. (2018) A Deep Learning SAR Target Classification Experiment on MSTAR Dataset. rep. Bonn: IEEE.

[13] Bandarupally, H., Talusani, H.R. and Sridevi, T. (2020) Detection of Military Targets from Satellite Images using Deep Convolutional Neural Networks. rep. Hyderabad: IEEE.

[14] Gupta, A. and Gupta, U. (2018) Military Surveillance with Deep Convolutional Neural Network. rep. Hyderabad: ICEECCOT.

[15] OUYANG, Y. et al. (2022) Military Vehicle Object Detection Based on Hierarchical Feature Representation and Refined Localization. rep. Nanjing: IEEE access.

[16] Convolutional neural network: Benefits, types, and applications (2023) Datagen. Available at: https://datagen.tech/guides/computer-vision/cnn-convolutional-neural-network.

[17] Gary, Chang,C.-C.(2018) CNN architectures - vggnet, Medium. Available at: https://medium.com/deep-learning-g/cnn-architectures-vggnet-e09d7fe79c45

[18] Das, S. (2019) CNN architectures: Lenet, alexnet, VGG, googlenet, ResNet and more, Medium. Available at:https://medium.com/analytics-vidhya/cnns-architectures-lenet-alexnet-vgg-googlenet-resnet-and-more-666091488df5.

[19] Baheti, P. (no date) What is transfer learning? [examples & newbie-friendly guide], Machine Learning. Available at: https://www.v7labs.com/blog/transfer-learning-guide#:~:text=In%20other%20words%2C%20transfer%20learning,when%20modeling%20the%20second%20task.