

A Convolutional Neural Network Based Approach for SAR Image Classification of Vehicles

Abhishek Ameta 1*, Vinayak Singh¹, Veena Devi S.V.¹

¹Department of Electronics and Communication Engineering,
RV College of Engineering, Bengaluru -59

Abstract:- Synthetic aperture radar (SAR) picture classification is a key procedure for SAR picture comprehension and understanding. Inspired by neural network technology, a model is constructed which helps in classification the images by taking original SAR image as input using feature extraction which is convolutional neural network. In this paper the 1-D feature are extracted from using principle component analysis. The 1-D feature vector set is given as input to the CNN layer. A CNN model that is composed of various multiple layers which are simple and nonlinear modules and has the capability to learn representations from the data. Convolutional neural network can extract high-level information and discover intricate structure, which dramatically improve the performance of many computer vision tasks, such as object detection, speech recognition, and image classification. This paper describes a Convolutional Neural Network based approach for SAR image classification. The Synthetic Aperture Radar images are formed when microwave signals bounce back from a surface of an object. These images are used for the classification of different vehicles. As Convolutional Neural Networks have higher efficiency than other algorithms, hence it is used for image classification. The data set used in this project is MSTAR dataset. This contains the images of various military vehicles. It is collected using a X-band SAR sensor with a 1-ft resolution.

The MSTAR dataset is widely used for classification and to test the algorithms. This project aims to classify, recognize and detect military vehicles with the help of Convolutional Neural Network algorithm.

Keywords: Classification, detection, Convolutional Neural Network, SAR images.

1.0 INTRODUCTION

Synthetic Aperture Radar (SAR) images are formed when the microwave signals bounce back from the surface of an object [1]. These images find applications in detecting the military vehicles [2-7]. The technique used in this is the use of moving antenna and a stationary target. The echoes are correlated to form a high resolution image of the target. Fig.1. depicts the military vehicle images which are present in the dataset with their respective SAR Image. These images are generated from an X-band SAR radar with a 1-Ft resolution [8-11].

There are 3 types of interaction of a signal:-

1. Smooth surface.
2. Rough surface.
3. Double bounce.

These interactions give the topography of the surface of the

object. Fig.1. depicts the process flow diagram of image classification. The model is trained using the dataset of Moving and Stationary Target Acquisition and Recognition (MSTAR) dataset. This consists of all images of various types of military vehicles.

When a model is trained it can result in three types of fitting:-

1. Under fitting: - When the model tries to cover each and every point on the axis it is called under fitting. If less number of features are used then the accuracy reduces.
2. Good Fit/Robust:-When the model is trained with the optimum number of features it is called good fit.
3. Over fitting: - When the model tries to cover each and every point on the axis it leads to over fitting.

Over fitting increases the computation time of training and it provides less accuracy.

The feature extraction is performed using Principal Component Analysis. As not all the features can be taken into consideration from an image which cause over fitting, hence only the principal components are taken [12-15]. PCA helps in avoiding the issue of over fitting. The PCA is used to convert a high dimensionality to a set of low dimensionality in terms of attributes or features.

1.1 Principal Component Analysis

In order to reduce redundancy of the image the PCA is widely used to convert a high dimensional image to a set of low dimensional image [16][17]. The PCA process results in principal components. These principal components are validated based on the validation dataset images. PCA or Principal Component Analysis is a mathematical method that performs orthogonal transformations and converts a correlated dataset into a uncorrelated data. The data after PCA contains principal components which are selected from various components the image contains, basically the principal components are the ones where the variance of data is the highest to get highly uncorrelated data [18-20] Hence PCA can be used to compress an image. The eigen vectors, eigen values and covariance matrix are obtained from the input images and hence the principal components are found. Pixels which are redundant are removed and are replaced with available colours in the image. This way PCA performs efficient compression.

1.2 Convolutional Neural Network

A CNN is composed of various layers which comprises of convolutional layer, max pooling layer, sigmoid layer and various other layers. Each layer has its own functions and output. Initially the input to the convolutional layer is an image which is a matrix of form $a \times b \times c$ where a denotes the height of image, b denoted the width of image and c denotes the colour of image or channel count. A Gray-scale images has c as 1. Then comes the kernel which is of size $j \times j \times l$, j is the size of one side and l can vary but should be less than c . CNN is majorly used for classification and detection of object or in general image processing and detection related applications not just because it achieves good accuracy but also because its extracts features automatically and the helps the developer to escape from loss of pixel spatial interaction.

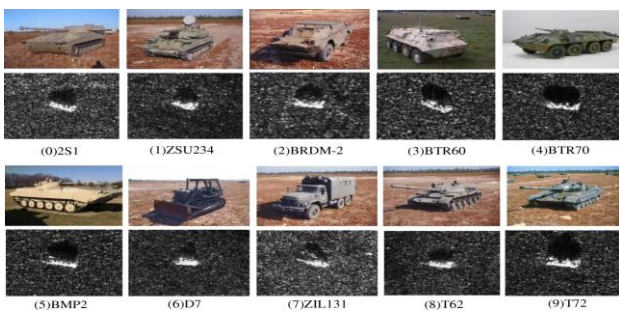


Fig.1. Military vehicles and their respective SAR Image.

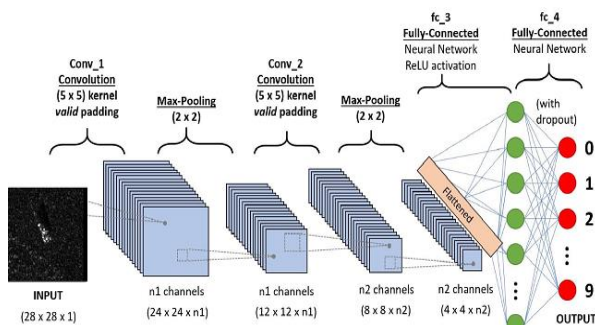


Fig.2. The convolutional Neural network layer.

2.1 Methodology adopted

The model is trained using the dataset which is Moving and the Stationary Target Acquisition and the Recognition (MSTAR) dataset. This consists of the images of various military vehicles which are 2S1, BMP2, BRDM_2, BTR60, D7, T62, ZIL131, and ZSU_23_4. It contains 2033 images each with a resolution of 158 x 158. The dataset comprises of images having resolution 158 x 158. The principal components are generated and the CNN model is trained. The final feature vector set is obtained after the fully convolutional layer. The feature vector is used to compare with the testing data. The model generates an error message if the image is not matching with the given image.

2.2 Experimental Details
 In order to solve the problem of over fitting, principal components must be calculated. Fig.3. explains the flow diagram of the principle component analysis. Principal Components are calculated with the help of Covariance

matrix which is mentioned in equation.1 and 2.

$$\text{Cov}(x,y) = \sum_{i=1}^n \frac{x^i - x'}{x^i + x'} \quad (1)$$

$$\text{Cov}(x,y) = \begin{bmatrix} \text{Cov}(x,x) & \text{Cov}(x,y) \\ \text{Cov}(y,x) & \text{Cov}(y,y) \end{bmatrix} \quad (2)$$

All the principal components are orthogonal i.e. they are independent of each other.

The negative value of covariance implies the two values are inversely proportional to each other.

The Eigen value can be obtained with the help of equation.3.

$$\text{Cov} - \lambda \cdot I = 0 \quad (3)$$

Where λ is the Eigen value and I is the identity matrix.

This will give more than one value of Eigen value which will have corresponding Eigen vector.

The Eigen vector can be calculated with the help of equation.4.

$$\text{Cov} - \lambda \cdot I \begin{bmatrix} x1 \\ y1 \end{bmatrix} = 0 \quad (4)$$

The principal component is the highest Eigen vector for the corresponding Eigen value.

The Significance of the parameters are:-

1. Covariance matrix:-It is a relation between how one variable is associated with other variable.
2. Eigen value:-Importance of these directions or how much amount of variance is explained in these directions.
3. Eigen vectors:-The direction in which the data is spread gives the eigen vectors.

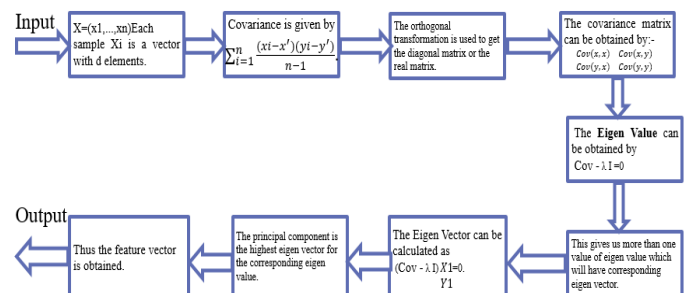


Fig.3. Process flow diagram of Principal Component Analysis.

$$\text{Accuracy} = \frac{\text{Total number of correct predictions.}}{\text{Total number of images used for testing.}} \quad (5)$$

Fig.7. explains the number of accurately predicted and incorrectly predicted values for the images classified.

The Convolutional Neural Network layer consists of one or more convolutional layers.

The convolutional neural network comprises of 4 layers:-

1. Convolutional layer.
2. ReLU Layer.
3. Pooling layer.
4. Fully connected layer.

1. Convolutional layer:-

The image is taken as a form of matrix in which the data present is marked with 1 and the other is marked with 0. The convolution is performed between the image matrix and the filter to get the convolved matrix. Fig.4. describes the output matrix using kernel.

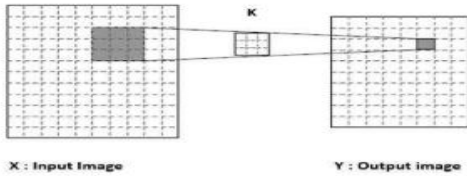


Fig.4. Generation of output matrix.

The output image is described as:-

$$M_{i+1} = M_i - p + 1.$$

Where, p denotes the size of kernel and M denotes the matrix.

2. ReLu Layer:-

This acts as an activation function. In this layer only the negative values are removed from the filtered image and replace it with zeros. This is done to avoid the values summing up to 0.

3. Pooling layer:-

Pooling layer is used to down sample the image. Pooling layers help in lessening the quantity of parameters required and subsequently this decreases the calculation required. It likewise helps in maintaining a strategic distance from over fitting.

There are two types of the pooling which is used:-

1. Maximum Pooling:-In this maximum value is selected for the matrix.
2. Average Pooling:- It involves the average of values of the entire set.

Here, in this project maximum pooling is being used. Fig.5. describes the output matrix after the pooling layer. The maximum pooling is used to calculate the output matrix.

$$Y(i,j) = \max(\sum_{u,v,k} Y(h_k \times i + u, w_k \times j + v))$$

Where Y denotes the matrix of order i and j and h_k and w_k are weights.

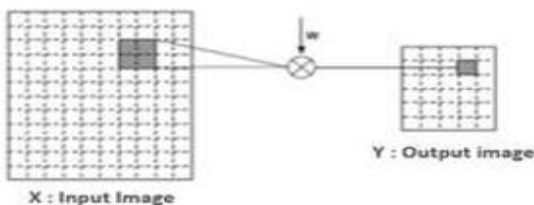


Fig.5. Generation of output matrix after pooling layer.

4. Fully connected layer:-

This is the layer in which the actual classification happens. Flattening process is employed to convert the resultant 2-dimensional clusters from pooled include set into a solitary long constant straight vector.

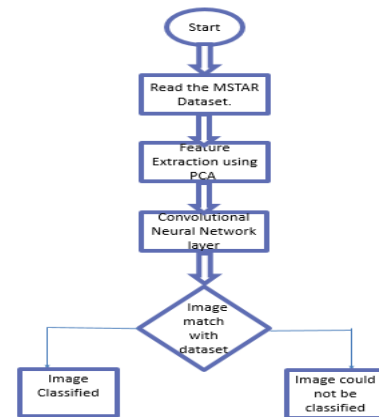


Fig.6. Process flow of SAR Image classification.

PROPOSED METHOD :-

CNN has been profoundly used for the image classification, however the computation time has been an issue as large amount of data is read. So in real time applications like SAR images target recognition need fast and effective computation and for that reducing the amount of data while not losing essential features for recognition is of primary importance. The proposed method first reads the image in form of matrix and further performs PCA image compression while reduces the size maintaining the principal components, removing the less important features. The images is then provided to the CNN model where the weights, biases, filter are initialized. The images in the form of matrix is passed onto various layers in CNN model. Convolutional layer performs the dot product between the filter and the image matrix. Then comes the ReLU layer, this layer removes the negative values from the filtered images and replace it with zero. Then the matrix is passed to the Pooling layer which is used to down sample the image. Max Pooling selects the maximum value in each patch. The Fully Connected layer performs flattening then. Flattening is the process of converting all the resultant 2-dimensional arrays from pooled feature map into a single long continuous linear vector. Adam optimizer is used for optimization. The network learns through various epochs and the network graph is saved after training of the model. The saved model is restored and new data is provided for prediction. The network calculates the probabilities of each outcome for each image. The outcome with highest probability is taken as the predicted outcome. The accuracy of the model is calculated based on the comparison of predicted and expected outcome.

3.0 RESULTS AND DISCUSSION

The Fig.7. Shows the graph of count and result. The accuracy is calculated using equation 5. The value of each dataset is obtained. Table.1. shows the value for each dataset. The model is trained on an Intel i5 processor and a RAM of 8GB and it is trained for 13 Epochs to achieve high accuracy. The feature extraction is done using PCA. Table 1 shows the value for each dataset. The network calculates the probabilities of each outcome for each image. The outcome with highest probability is taken as the predicted outcome.

The accuracy of the model is calculated based on the comparison of predicted and expected outcome. This technique aims to provide the image classification with a high accuracy and computation time as compared to other algorithms. The algorithm helps us to achieve image classification using neural network. The project finds application in wide areas such as surveillance, defense purpose, in radars etc. The accuracy and computation is better compared to other algorithms and hence it is suitable for wide range of applications. True indicates the number of images predicted correctly and False indicates wrong Predictions. The figure shows that the algorithm has good efficiency as count of True is much higher than count of False. Number of True =1729, False=109, Accuracy = 94.06%.

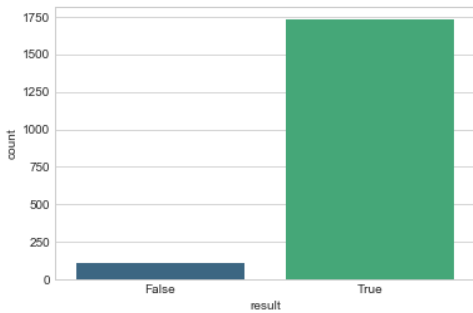


Fig.7. Count versus result.

TABLE.1. The value for each dataset.

	ZS1	BMP2	BRDM_2	BTR60	D7	T62	ZIL131	ZSU_23_4	max	target	result
0	0.755446	0.005034	0.175989	4.780000e-15	1.560000e-06	0.000894	0.062635	2.600000e-12	ZS1	ZS1	True
1	0.942875	0.002231	0.001249	1.030000e-14	5.422240e-04	0.001978	0.053125	2.080000e-10	ZS1	ZS1	True
2	0.931645	0.007806	0.059642	3.190000e-16	1.029270e-04	0.000743	0.000261	7.720000e-12	ZS1	ZS1	True
3	0.988896	0.022407	0.006909	5.220000e-18	6.790000e-08	0.001458	0.000330	3.160000e-13	ZS1	ZS1	True
4	0.997717	0.000013	0.000573	1.040000e-10	2.570000e-07	0.001066	0.000631	9.540000e-09	ZS1	ZS1	True
5	0.986682	0.000025	0.024483	2.540000e-17	5.630000e-07	0.007285	0.001545	7.810000e-10	ZS1	ZS1	True
6	0.281842	0.000295	0.729638	3.140000e-16	2.380000e-06	0.000059	0.008165	1.940000e-11	BRDM_2	ZS1	False
7	0.156931	0.000388	0.836625	6.270000e-17	2.000000e-07	0.000214	0.000582	4.530000e-12	BRDM_2	ZS1	False
8	0.981659	0.000007	0.315000	3.670000e-19	4.920000e-07	0.001054	0.002280	1.770000e-12	ZS1	ZS1	True
9	0.929298	0.000007	0.069762	4.050000e-19	2.750000e-08	0.000791	0.000143	2.620000e-11	ZS1	ZS1	True

4.0 CONCLUSIONS

The approach to the problem of SAR Image classification is described using Convolutional Neural Network which has proved to be very successful in the recent times in image classification and detection applications. The image processing helps in the organization of the data which can be then used for feature extraction. The feature extraction and compression done through PCA helps further in reducing the computational time. The 1-D features extracted using PCA are used to provide input to the CNN layer. The MSTAR dataset is utilized to prepare the model. The high numbers of samples are utilized to give a precision and accuracy. The model also laid focus on computation time by using Principal Components Analysis to reduce the data used for computation. The neural network is trained using samples and convolutional layers with leaky ReLU activation functions and accuracy of 94.06% is achieved.

5.0 REFERENCES

- [1] D. E. Dudgeon and R. T. Lacoss, "An overview of automatic target recognition," Lincoln Lab. J., vol. 6, no. 1, pp. 3–10, 1993.
- [2] ZHAI Jia*, DONG Guangchang, CHEN Feng, XIE Xiaodan, Qi Chengming, Li Lin A Deep Learning Fusion Recognition Method Based On SAR Image Data, ICKIA International Conference, 2018.
- [3] Ali El Housseini, Abdelmalek Toumi, Ali Khenchaf Deep Learning for Target recognition from SAR images, 7th SEMINAR ON DETECTION SYSTEMS: ARCHITECTURES AND TECHNOLOGIES (2017)
- [4] Jie Geng, Hongyu Wang Member IEEE, Jianchao Fan, Member IEEE, and Xiaorui Ma, Member IEEE, SAR Image Classification via Deep Recurrent Encoding Neural Networks, IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, VOL. 56, NO. 4, APRIL 2018.
- [5] Sizhe Chen Student Member IEEE, Haipeng Wang Member IEEE, Feng Xu Senior Member IEEE and Ya-Qiu Jin Fellow IEEE, Target Classification Using the Deep Convolutional Networks for SAR Images, IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, VOL. 54, NO. 8, AUGUST 2016.
- [6] Jong-Il Park, Sang-Hong Park, and Kyung-Tae Kim, Member, IEEE, New Discrimination Features for SAR Automatic Target Recognition, IEEE GEOSCIENCE AND REMOTE SENSING LETTERS, VOL. 10, NO. 3, MAY 2013.
- [7] Y. Cui, G. Zhou, J. Yang, and Y. Yamaguchi, "On the iterative censoring for target detection in SAR image," IEEE Geosci. Remote Sens. Lett., vol. 8, no. 4, pp. 641–645, Jul. 2011. 4816 IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, VOL. 54, NO. 8, AUGUST 2016
- [8] J. I. Park and K. T. Kim, "Modified polar mapping classifier for SAR automatic target recognition," IEEE Trans. Aerosp. Electron. Syst., vol. 50, no. 2, pp. 1092–1107, Apr. 2014.
- [9] L. M. Novak, G. J. Owirka, W. S. Brower, and A. L. Weaver, "The automatic target-recognition system in SAIP," Lincoln Lab. J., vol. 10, no. 2, pp. 187–201, 1997.
- [10] L. M. Novak, "State-of-the-art of SAR automatic target recognition," in Proc. IEEE Int. Radar Conf., 2000, pp. 836–843.
- [11] T. D. Ross, J. J. Bradley, L. J. Hudson, and M. P. O'Connor, "SAR ATR: Sowhat's the problem? AnMSTARperspective," in Proc. 6th SPIE Conf. Algorithms SAR Imagery, 1999, vol. 3721, pp. 566–579.
- [12] E. R. Keydel, S. W. Lee, and J. T. Moore, "MSTAR extended operating conditions: A tutorial," in Proc. 3rd SPIE Conf. Algorithms SAR Imagery, 1996, vol. 2757, pp. 228–242.
- [13] A. Hirose, Ed., Complex-Valued Neural Networks: Advances and Applications. Hoboken, NJ, USA: Wiley-IEEE Press, 2013.
- [14] Q. Zhao and J. C. Principe, "Support vector machines for SAR automatic target recognition," IEEE Trans. Aerosp. Electron. Syst., vol. 37, no. 2, pp. 643–654, Apr. 2001.
- [15] Y. J. Sun, Z. P. Liu, S. Todorovic, and J. Li, "Adaptive boosting for SAR automatic target recognition," IEEE Trans. Aerosp. Electron. Syst., vol. 43, no. 1, pp. 112–125, Jan. 2007.
- [16] C. F. Olson and D. P. Huttenlocher, "Automatic target recognition by matching oriented edge pixels," IEEE Trans. Image Process., vol. 6, no. 1, pp. 103–113, Jan. 1997.
- [17] N. M. Sandirasegaram, "Spot SAR ATR using wavelet features and neural network classifier," Def. R&D Canada, Ottawa, ON, Canada, DRDC Ottawa TM 2005-154, Tech. Memorandum, 2005.
- [18] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," Science, vol. 313, no. 5786, pp. 504–507, 2006.
- [19] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in Proc. Adv. Neural Inf. Process. Syst., 2012, pp. 1097–1105.
- [20] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," Proc. IEEE, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.