Aircraft Image De-noising and Identification using Deep Neural Network

Mridusmita Sharma, Kandarpa Kumar Sarma and Nikos Mastorakis
Department of Electronics and Communication Engineering, Gauhati University,
Guwahati-781014, Assam, India.

Abstract— Images corrupted during transmission and acquisition required de-noising for proper interpretation and reliable recognition. Though related traditional methods are known to be reliable in de-noising and identification, learning aided approaches have become popular recently. Subsequent, deep learning has been accepted to be an efficient mechanism and is found to be increasingly becoming integral element for a range of image processing and computer vision applications. This work deals with the formulation of a system based on Auto-encoder (AE) and Stacked Auto-encoder (SAE) configured for de-noising of certain military aircrafts as part of an automatic target recognition (ASR) system. The ASR is based on a class of classifiers that include the soft-max layer, conventional Artificial Neural Network (ANN) and Deep Neural Network (DNN) of the type convolutional neural network (CNN). The sample set includes five image types corrupted with Gaussian, Poisson, Speckle, Salt and Pepper noise for de-noising by AE and SAE topologies and identification by the CNN. Further, the image sets are subjected to signal to noise ratio (SNR) variation between -3 to 20 dB which increases the data volume for training which is necessary to make the system robust. Despite higher computational latency, the efficiency of the proposed approach is justified by the experimental results.

Keywords—. AE, SAE, DNN, Deep, Learning, De-noising.

I. INTRODUCTION

De-noising is an important image processing operation which determines the outcome of subsequent image interpretation techniques. A host to techniques performing image denoising have been reported and have proven their worth in different fields of application. Yet the requirement to automate the process and improve efficiency have laid stress on the design of learning based approaches. The advantages of such approaches is the fact these are learning based which means that these learn from the surrounding, retain it and use it subsequently. Further, immediately after the training phase is over, such techniques can be used repeatedly without much variations in the parameters and models. Throughout the last few decades neuro-computing based techniques have been extensively used for image restoration and de-noising applications. Of late, the shift has been towards the use of deep learning (DL). Deep Neural Networks (DNN) have been the preferred options for many real world situations [1] - [4] including image processing and computer vision applications. Several works related to the application of DL in high-level computer vision tasks have been reported. Works [5] [6], image classification [7], object detection [8], and semantic segmentation [9] are some of the examples. Deep learning has also been investigated for low level computer vision tasks such as image de-noising [10], [11], [12].

Automatic target recognition (ATR) systems with special regards to military aircrafts have adopted synthetic aperture radar (SAR) and Inverse SAR (ISAR) as reliable techniques. With SAR and ISAR methods of ATR, DNN approaches like Convolutional Neural Networks (CNN) have been used [13]. Improvement in performance of the DNN assisted ATR in case of military aircrafts is related to the reliable de-noising of the test samples. Among DNN techniques, auto-encoder (AE) and the stacked AE (SAE) [14] have been found to be fast and efficient among de-noising tools. Often, like all signals, images are corrupted during transmission or acquisition or editing. De-noising is a part of image restoration. Image restoration intends to recover back a signable portion of the signal with quality as close to the original. Image de-noising is an important preprocessing step in a host of applications and becomes essential when an image is corrupted by additive white Gaussian noise (AWGN) which commonly occurs in the communication channel or may creep into while recording the image due to erroneous sensors or calibration.

Efficiency of the ATR performance is linked to its ability to performance proper interpretation in presence of severe input degradation. This aspect is strengthened by dedicated de-noising blocks working in tandem with ATR units. The application of learning aided tools holds the key in deriving efficient performance as observed in a host of computer vision designs. This paper focuses on the design of AE and SAE based approaches for de-noising of certain military aircrafts as part of an automatic target recognition (ASR) system. The ATR is based on a class of classifiers that include the soft-max layer, conventional Artificial Neural Network (ANN), Support Vector Machine (SVM) and Deep Neural Network (DNN) of the type convolutional neural network (CNN).

Mridusmita Sharma and Kandarpa Kumar Sarma are with Department of Electronics and Communication Engineering, Gauhati University, Guwahati-781014, Assam, India (email- mriduzb@gmail.com, kandarpaks@gmail.com).

Nikos Mastorakis is with Technical University of Sofia, Sofia, Kliment Ohridski 8, Bulgaria (e-mail:, mastor@tu-sofia.bg)

The rest of the description includes certain basic considerations (Section II), details of the proposed work (Section III), experimental results (Section IV) and the conclusions derived.

II. Basic Considerations

In this section, we include certain relevant notions which are linked with the design of the system.

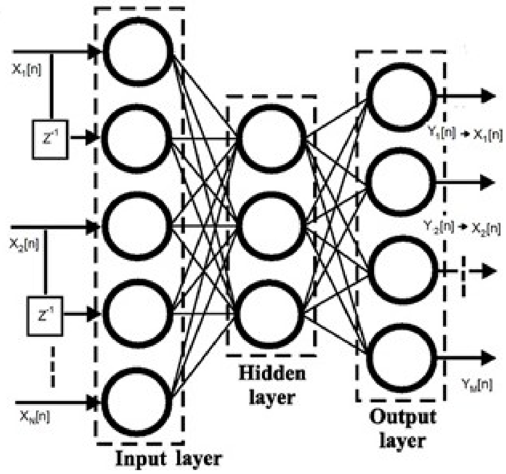

Fig. 1. Generic auto encoder

A. De-noising

The primary objective of the de-noising operation is to recover the image content from the surrounding noise. In the process noise gets subtracted and contributes towards the improvement of the quality of the image. A host of techniques are available through which de-noising can be performed but in the present work the focus in on the use of AE and SAE in case of a few military aircrafts.

B. Auto-encoder (AE)

AEs are feed-forward (FF) Artificial Neural Networks (ANN) with three layers (input, hidden and output) that copy their inputs to the outputs. AEs can reduce dimensions and preserve as much content as possible. AEs can also be trained to derive fresh representation and generate new attributes. There are three primary segments of an auto-encoder. These are encoder, latent representation and decoder. The encoding section can be represented as $h = f(x)$ while the decoding function has an expression $r = g(h)$ such that input x is as close as output r. AEs learn better when there is noise in the input. This way the AE is found to be useful for de-noising applications. AEs can recover true signal content from noise mixed input. This is true with images as well. Auto-encoders are unsupervised networks that can make both linear and non-linear transformations. Figure 1 shows a generic AE.

C. Stacked AE (SAE)

The SAE is formed by combining multiple AEs together resulting in multiple layer network. It performs de-noising encoding by using unsupervised pre-training layers. Each layer is pre-trained to perform feature selection from the input using the preceding layer. The second stage is used for supervised fine-tuning where apriori knowledge maybe made available. Figure 2 shows a SAE.

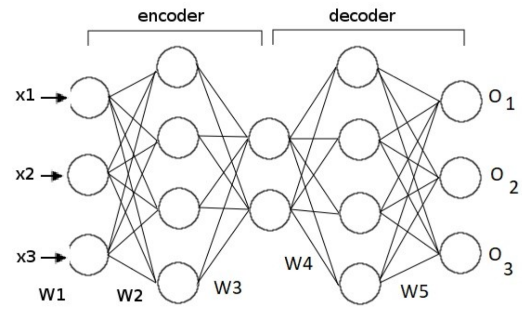

Fig. 2. Generic stacked auto encoder

For an input $[x^{(T)}]$, the output generated by the stacked auto-encoder is,

$$X_{A1} = x_1.w_{111}$$
$$X_{A2} = x_2.w_{112}$$
$$.$$
$$X_{AT} = x_T.w_{11T}$$

$$\sum_{k=1}^{T} X_{Ak} = f_k(\sum_{k=1}^{T} x^k.w_{11k}) \quad (1)$$

where $k = 1, 2, ...T$, T is the size of the input layer. For the hidden layer,

$$X_{B1} = X_{A1}.w_{211} + X_{A2}.w_{221} + \cdots + X_{AT}.w_{2T1} + b_{11}$$
$$X_{B2} = X_{A1}.w_{212} + X_{A2}.w_{222} + \cdots + X_{AT}.w_{2T2} + b_{12}$$
$$.$$
$$X_{BM} = X_{A1}.w_{21M} + X_{A2}.w_{22M} + \cdots + X_{AT}.w_{2TM} + b_{1M}$$

$$\sum_{i=1}^{M} X_{Bi} = f_i(\sum_{i=1}^{M}(\sum_{k=1}^{T} X^{Ak}.w_{2ki} + b_{1i})) \quad (2)$$

where $i = 1, 2, ...M$, M is the size of the hidden layer. For the output of the first auto-encoder,

$$X_{C1} = X_{B1}.w_{311} + X_{B2}.w_{321} + \cdots + X_{BM}.w_{3M1} + c_{11}$$
$$X_{C2} = X_{B1}.w_{312} + X_{B2}.w_{322} + \cdots + X_{BM}.w_{3M2} + c_{12}$$
$$.$$
$$X_{CN} = X_{B1}.w_{31N} + X_{B2}.w_{32N} + \cdots + X_{BM}.w_{3MN} + c_{1N}$$

$$\sum_{j=1}^{N} X_{Cj} = f_j(\sum_{j=1}^{N}(\sum_{i=1}^{M} X^{Bi}.w_{3ij} + c_{1j})) \quad (3)$$

where $j = 1, 2, ...N$, N is the size of the output layer of the first auto-encoder.

Equation 3 is also the input to the second auto-encoder and the hidden layer of the second auto-encoder is represented by,

$$X_{D1} = X_{C1}.w_{411} + X_{C2}.w_{421} + \cdots + X_{CN} \ .w_{4N1} + d_{11}$$
$$X_{D2} = X_{C1}.w_{412} + X_{C2}.w_{422} + \cdots + X_{CN} \ .w_{4N2} + d_{12}$$
$$.$$
$$X_{DP} = X_{C1}.w_{41P} + X_{C2}.w_{42P} + \cdots + X_{CN}.w_{4NP} + d_{1P}$$
$$\sum_{l=1}^{P} X_{Dl} = f_l(\sum_{l=1}^{P}(\sum_{j=1}^{N} X^{Cj}.w_{4jl} + d_{1l})) \qquad (4)$$

where $l = 1, 2, \ldots P$, P is the size of the hidden layer of the second auto-encoder.

For the output of the second auto-encoder,

$$O^1 = X_{E1} = X_{D1}.w_{511} + X_{D2}.w_{521} + \cdots + X_{DN}.w_{5N1} + e_{11}$$
$$O^2 = X_{E2} = X_{D1}.w_{512} + X_{D2}.w_{522} + \cdots + X_{DN}.w_{5N2} + e_{12}$$
$$.$$
$$O^{T}{}' = X_{EQ} = X_{D1}.w_{51Q} + X_{D2}.w_{52Q} + \cdots + X_{DN}.w_{5NQ} + e_{1Q}$$
$$\sum_{r=1}^{Q} X_{Er} = f_r(\sum_{r=1}^{Q}(\sum_{l=1}^{P} X^{Dl}.w_{5lr} + e_{1r})) \qquad (5)$$

where $r = 1, 2, \ldots Q$, Q is the size of the output layer of the second auto-encoder.

D. Convolutional Neural Network (CNN)

A CNN is a type of DNN which use the convolution operation to transform one volume of activations to another through certin types of layers namely convolutional layer, pooling layer, and fully-connected layer. The CNN transforms the original input volume using layer by layer processing to certain class scores. There can be several combinations of the basic architectural layers which generates considerable amount of processing but enhances efficiency and reliability. CNNs use a type of gradient descent algorithm to learn the input patterns and associate them with certain class labels without requiring human crafted features [5].

III. Proposed Work

The work has two components. First, there is a de-noising section and next, another segment performs recognition. For the de-noising part, AE and SAE blocks are trained and use. The ATR is designed two CNN based structures attached to soft-max layer, conventional ANN specially a Multi Layer Perceptron (MLP) and SVM used to trained to recognize the targets and work along with the de-noising layer. The learning based ATR is first trained extensively and subsequently subjected to a set of testing. Figure 3 summarizes the work-flow.

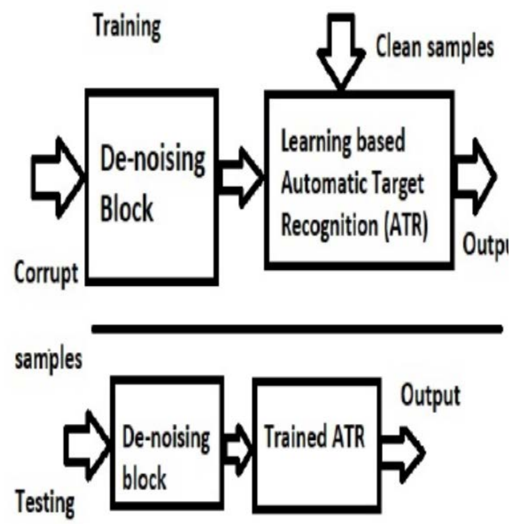

Fig. 3. Work flow of the proposed approach

A. Dataset

There are two stages of the work. First, de-noising is done using the SAE block. Next, the ATR is configured using two configurations of the CNN. The data sets used for both the work vary in size. Five image types are taken for the work which are mixed with Gaussian, Poisson, Speckle, Salt and Pepper noise. For each of these image sets signal to noise ratio (SNR) variation between -3 to 10 dB are taken. This gives a total of 280 images for performing the task. With variation in the illumination (one higher and one lower than the original), another set of 560 images are taken to test the trained networks to carry out the de-noising operation. Figure 4 shows a set of samples considered for training.

For the ATR, the image set is formed separately but includes all the samples used for the de-noising block. Five image types of military aircrafts corrupted with Gaussian, Poisson, Speckle, Salt and Pepper noise for de-noising by AE and SAE topologies and identification by the CNN. Further, the image sets are subjected to signal to noise ratio (SNR) variation between -10 to 20 dB which increases the data sets for training necessary to make the system robust. A set of 4320 images are taken for training. Another set of 4320 images with resolution variation are taken for testing.

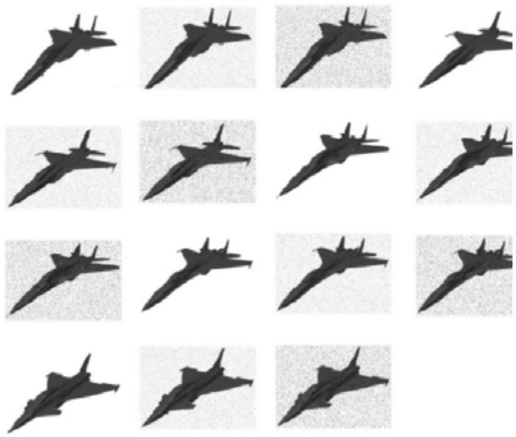

Fig. 4. A few training samples

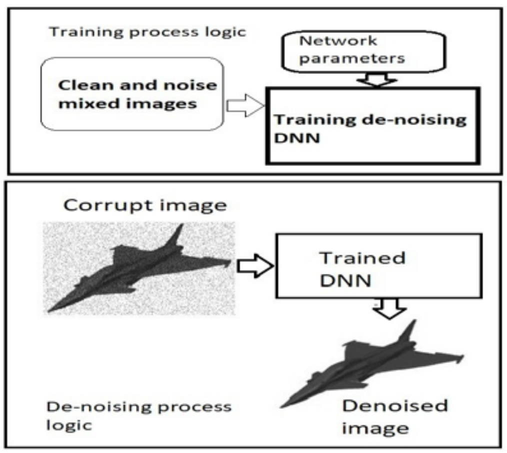

Fig. 5. Process logic

B. Methodology

The approach is based on collection of the data samples which in this case is formed by several sets of training and testing samples for the complete system covering both the de- noising blocks and the identification section. The configuration of the networks is another important task. During training, four different noise types and several cases of SNR variations are considered to make the systems robust. At the end of the training and testing, the performance is measured in terms of PSNR values of the input and output images. Further, the training latencies of the networks are also recorded. Extensive tests are performed to check the robustness of the ATR system under low and high SNR variation conditions. The ATR section is designed using two different configurations of the CNN.

C. Training the De-noising networks

The process logic of the de-noising part carried out using AE and SAE is summarized in Figure 5. It has two compo- nents. First, two types of networks namely the AE and SAE are trained. Next, the trained networks are tested with the images in the data base. Several images of military aircrafts are taken for the work which are mixed with Gaussian, Poisson, Speckle, Salt and Pepper noise implying a range of de-noising condition. For each of these image sets SNR variation between -3 to 10 dB are taken. The work is assumed to be a part of an ATR system which requires a reliable de-noising stage to provide accurate assessment of the situation. Two de-nosing networks, the first one an AE and the other a SAE are used for the de-noising procedure. The parameters of the networks are summarized in Table I. The AE network is first configured by following a sequence of trail and error runs. The network with hidden layer size 75 with a decoder layer with purely linear activation function is found to be most efficient in terms of mean square error (MSE) and computational time. Similarly, a SAE with three hidden layers with details as shown in Table I is found to be suitable. For both the cases the MSE attained and the time in seconds taken at the end of 2000 epochs are taken to be two criteria for selection. Five image classes with four different SNR variations constituting 280 images are taken for training both the images. Such a set of images are shown in Figure 4.

D. Training the ATR Networks

The ATR is formed using a DNN with multiple layers as shown in Figure 6. The network type adopted here is based on the CNN topology. The two different configurations of the CNN are used to ascertain the performance improvement that can be derived in terms of lower computational complexity and better accuracy. Two networks have different combinations of several layers. First comes the input layers. Next is the layer of the convolution (CNV) layer followed by a max-pooling mechanism and the fully connected (FC) layers. At the end there is the classifier which is formed by a soft-max layer. The CNN based ATR is designed in two different configurations as outlined in Table II. The first network (Config1) is formed with one input, four middle and one classifier layers while the second block has a set of input, middle and final layers with convolutional, FC and rectified linear units (ReLU) the details of which are given in Table II.

In the first network, there are four convolutional layers of sizes 32×32, 16×16, 8×8 and 4×4. The input taken is of size 64×64. Subsequently, the learning takes place in all the convolutional layers which result in an optimal set of features which are used to train the soft-max layer. The CNN architecture ensures that the convolutional filter masks learn and produce the best activation so as to become spatially more relevant for use as local input to subsequent layers. It enables the learned filters to respond when known patterns (or features) occur in the training data within the active field. The larger filter masks (32×32) placed in the beginning takes in the pixels with spacing of one and two are used. In each subsequent stages, with the 16×16 sized filters, four numbers, with 8×8 eight and with 4×4 ones sixteen filter blocks are used to capture the feature details. This helps in extracting minor variations in the input despite the fact that

TABLE I
PARAMETERS OF DE-NOISING AE AND SAE NETWORKS

Sl. no.	Item	Parameters
1	Hidden Layers	AE one SAE three
2	Max. Epochs	AE = 3000 SAE = 3000
3	Hidden Layers Length	AE one with 75 neurons SAE three first with 75, second with 48, third with 42
4	Criteria for fixing hidden layer	MSE convergence and Computational time
5	Weight Regularization	0.005
6	Sparsity Regularization	2
7	Sparsity Proportion	0.05
8	Decoder activation	Purely linear

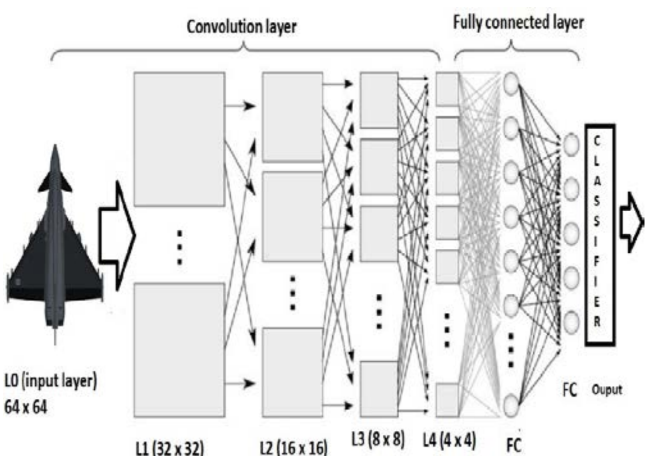

Fig. 6. CNN architecture used for designing the ATR

initially redundancy increases the computational complexity of the system. Out of the total of 4320 images, five batches are taken to train the ATR. Twelve hundred epochs are taken to train the ATR formed with this configuration. The next configuration is a bit complex but is more popular. It is formed by an input layer of size 64 × 64 × 3. Next the middle layers are formed. First, a convolutional block with 32 numbers of 5 × 5, followed by one ReLU and a max-pooling layer of size 3 × 3 are placed. Next, another convolutional layer of 32 numbers of 5 × 5 mask followed by an ReLU and a max-pooling layer of 3 × 3 is placed. The third stage is constituted by another identical combination of 32 numbers of 5 × 5 convolutional masks followed by ReLU and 3 × 3 max-pooling filter. The subsequent layer is a FC block with sixteen numbers of FC layers followed by ReLU another two FC and finally a soft-max layer.

TABLE II
DNNS IN TWO DIFFERENT CONFIGURATIONS- FIRST ONE (CONFIG1) WITH ONE INPUT, FOUR MIDDLE AND ONE CLASSIFIER LAYERS WHILE THE SECOND BLOCK HAS A SET OF INPUT, MIDDLE AND FINAL LAYERS WITH CONVOLUTIONAL, FC AND RECTIFIED LINEAR UNITS (RELU)

Sl No	Structure	Layer type	Size	Remark
		Input	64 × 64	L1, L2 L3, L4
		L1	32 × 32	are convolution
1	Config1	L2	16 × 16	layers
		L3	8 × 8	
		L4	4 × 4	
		Soft-max	5 × 1	
		Input	64 × 64 × 3	
			Convolutional	32 numbers,5 × 5
		Middle	ReLU	
			Max-pooling	3 × 3
2			Convolutional	32 numbers,5 × 5
		Middle	ReLU	
	Config2		Max-pooling	3 × 3
			Convolutional	32 numbers,5 × 5
		Middle	ReLU	
			Max-pooling	3 × 3
			FC	16 FC layers
		Last	ReLU	
			FC	2 FC
			Soft-max	

IV. RESULTS AND DISCUSSION

The results are discussed in reference to the two distinct aspects covered in the work. The first part is linked to the de-noising part carried out using the AE and the SAE while the second section is related to the DNN based ATR with which the de-noising networks are connected.

E. Results of the De-noising block:

The networks formed by AE and the SAE are trained upto 5000 epochs maximum. After the training is over, the networks are tested with noise corrupted images. Figure 7 shows a set of images used for testing. For varying epochs, the output of the AE and SAE varies. At the end of 1000 epochs the output from the AE is as in Figure 8. There is improvement in the performance of the SAE at the same level. This is obvious from Figure 9. It is observed that after the training is sustained for more number of epochs the overall appearance and the peak signal to noise ratio (PSNR) value improves considerably.

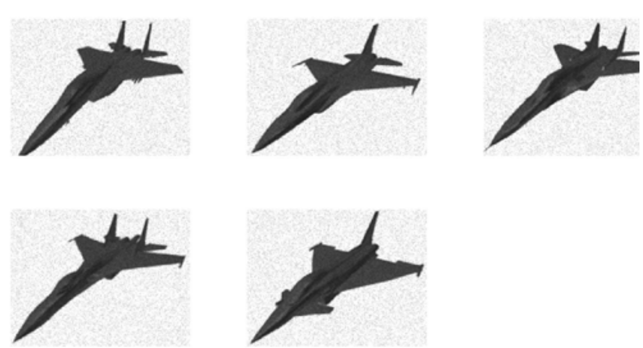

Fig. 7. Test samples

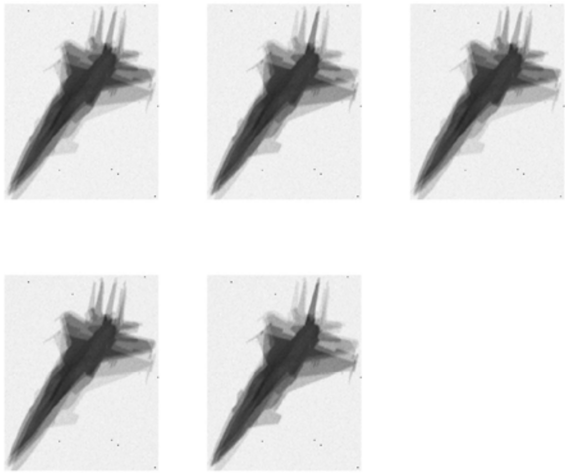

Fig. 8. Output of the trained AE after 1000 epochs

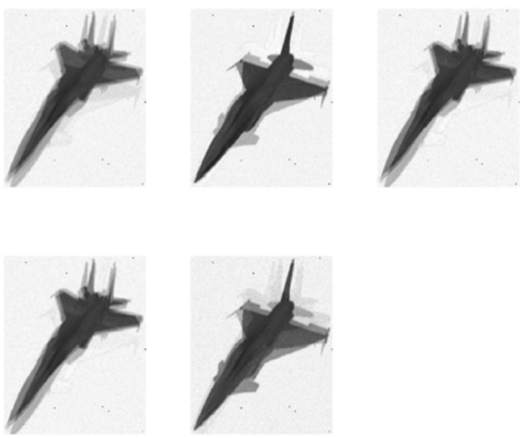

Fig. 9. Output of the trained SAE after 1000 epochs

Due to the use of the SAE, the PSNR shows an improvement between 8% to 27% compared to the AE network. Output PSNR from a AE and SAE trained upto 3000 epochs is shown in Figure 10. The output obtained from a AE trained upto 3000 epochs is shown in Figure 11. Similarly, the de-noising output from a SAE trained upto 3000 epochs is shown in Figure 12. The improvement provided by the SAE is clearly visible. A set of 280 images are taken for the training while a set of 560 images are used for testing. All the four different noise types are considered during testing along with clean samples. Two levels of illumination variation and the original intensity levels are taken for testing. The results depicted are the average performance of complete set of trails carried out. The training has been carried out using an Intel i-7 processor with 8 GB RAM.

Though there is a significant increase in the PSNR values due to the de-noising carried out by the SAE, the computational time increases considerably. There is an increase between 52% to 87% in time complexity due to the use of the SAE. This is seen in Figure 13. The training latency makes the technique disadvantageous. Otherwise, with an unsupervised approach the AE and SAE based approaches are considerably reliable.

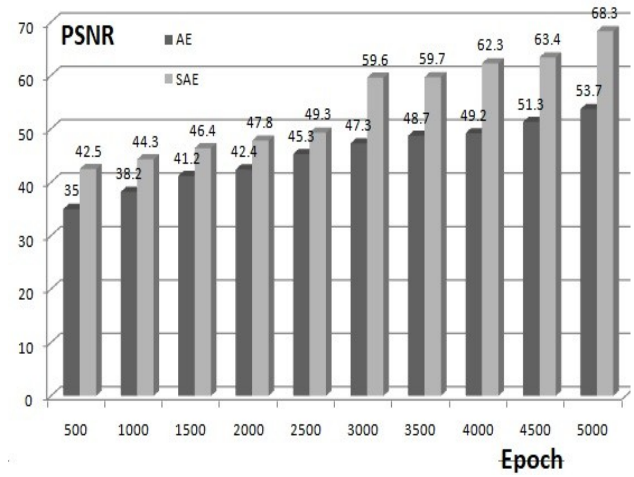

Fig. 10. Output PSNR from a AE and SAE trained upto 3000 epochs

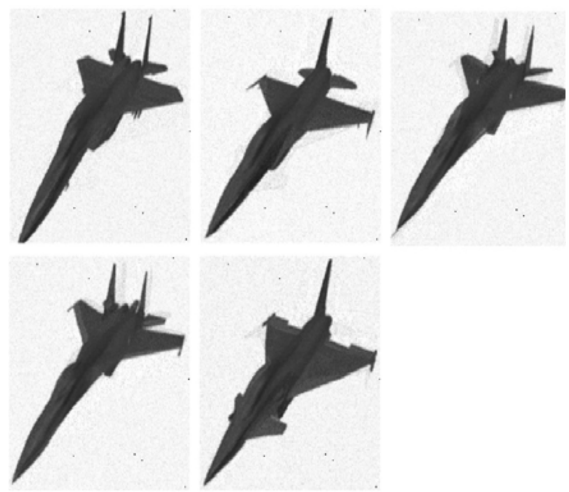

Fig. 11. Output obtained from a AE trained upto 3000 epochs

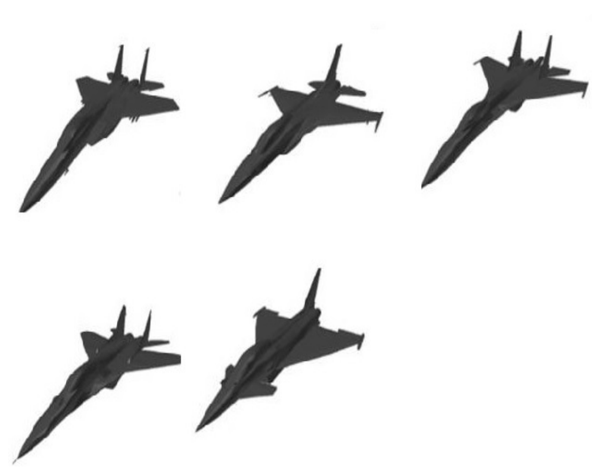

Fig. 12. Output obtained from a SAE trained upto 3000 epochs

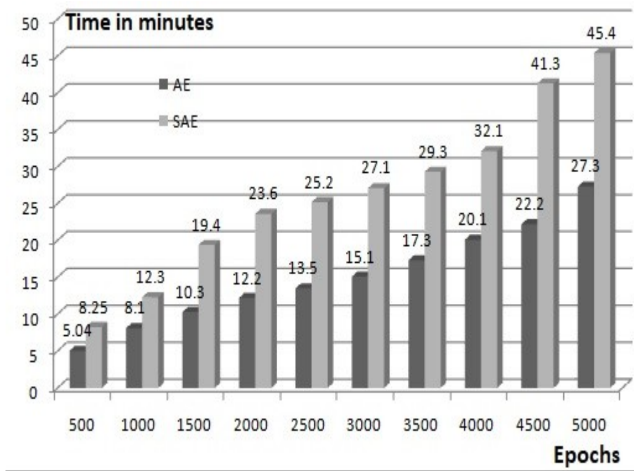

Fig. 13. Comparison of time in minutes between AE and SAE trained upto 3000 epochs

F. Results of the ATR block integrated with the AE-SAE de-noising system:

The summary results of the ATR block integrated with the AE-SAE de-noising system is shown in Table III. The results are derived using data base images applied to the trained ATR system working together with the AE-SAE system. The training and testing times are noted. It is found that after the training is over, the testing phase takes times which are nearly similar for all the methods adopted including the benchmark classifier techniques namely the soft-max, ANN and SVM classifiers. The training time associated with the CNN based classifiers is much more compared to the benchmark classifier techniques namely the soft-max, ANN and SVM classifiers which is due to the configuration, nature of processing and structure of the systems. However, the benefits are in terms of better accuracy (atleast 13% more) compared to the benchmark classifier techniques namely the soft-max, ANN and SVM classifiers. Also, the learning is extensive and resilient showing no change in accuracy despite variations in SNRs in the input image. This is seen in the performance plot depicted in Figure 14. The CNN based classifiers show consistent accuracy (in the 90s range) despite variations in SNR which is desirable for a learning aided ATR. The CNN block labeled as *CNN-Config2-Softmax* in combination with SAE acts as a much reliable ATR despite extensive variations in the SNR.

In -15dB SNR, the AE-aided blocks show an accuracy of around 81% while the SAE driven CNN based ATRs show an improvement of atleast 2% (Figure 14). With images corrupted by -12dB SNR, the *SAE-CNN-Config2-Softmax* combination shows a performance improvement of around 4%. At -4dB SNR, this improvement is around 7%. This is significant as the degradation in image quality has no effect and the performance is found to be nearly consistent. The experiments also show that the de-noising blocks aid the learning of the CNN based ATR blocks. However, the absence of the AE/SAE de-noising blocks don't degrade the performance significantly yet their performance is highly desirable as it improves the accuracy. This is obvious from the results depicted in Table IV. The presence of the AE/SAE de-noising blocks improve the accuracy by atleast 7% but adds more computation cycles. With previously trained AE/SAE blocks, this training latency shall be an insignificant factor.

TABLE III
SUMMARY RESULTS OF THE ATR BLOCK LINKED WITH THE AE-SAE DE-NOISING SYSTEM

De-noising block.	Classifier Structure	Training Time	Testing time in secs	Accuracy %
AE	Softmax	155 secs	1.2	63
	ANN	210 secs	1.1	71
	SVM	95 secs	1.1	73
	CNN-Config1-Softmax	125 mins	1.2	86
	CNN-Config2-Softmax	1055mins	1.2	92
SAE	Softmax	187 secs	1.1	67
	ANN	315 secs	1.1	73
	SVM	94 secs	1.1	75
	CNN-Config1-Softmax	133 mins	1.2	91
	CNN-Config2-Softmax	1092 mins	1.2	98

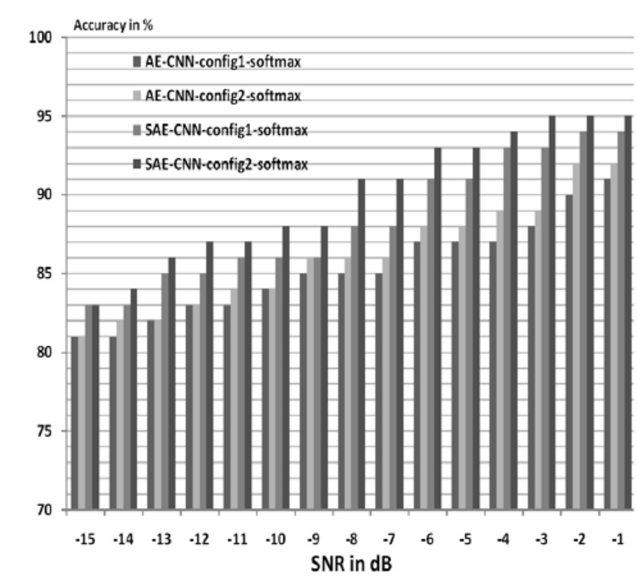

Fig. 14. Accuracy v/s SNR for the AE and SAE de-noising aided ATR blocks using two different configurations of CNN

TABLE IV
PERFORMANCE OF THE ATR WITH AND WITHOUT THE DE-NOISING BLOCKS

Sl No.	Item	Average accuracy (%)
1	ATR without de-noising	89
2	ATR with de-noising	96

IV. Conclusion

In this paper, we have focused on the design of DNN based ATR integrated to AE and SAE based approaches for de-noising of certain military aircrafts. The system is configured to make identification of the military aircrafts. Several images of military aircrafts are taken for the work which are mixed with Gaussian, Poisson, Speckle, Salt and Pepper noise implying a range of de-noising condition. For each of these image sets a range of SNR variations are considered both during training and testing. Experimental results have show that the SAE based approach is more reliable during de-noising though it has a higher computational latency. Due to the use of the SAE, the PSNR shows an improvement between 8% to 27% compared to the AE network. Though the PSNR values show improvements, there is a significant increase in time complexity due to the use of the SAE which is between 52% to 87%. The results for the ATR are derived from the trained system working together with the AE-SAE blocks. The testing phase has response latency which are nearly similar for the approaches considered including classifiers like the soft-max, ANN and SVM. The training time associated with the CNN based classifiers is much more compared to the benchmark classifier techniques which is due to the configuration, nature of processing and structure of the systems. However, the benefits are in terms of better accuracy (atleast 13% more) compared to the benchmark classifier techniques. The proposed approach can be effective used in actual situation to provide reliable identification and consistent accuracy even under severe SNR variations.

References

[1] G. Devi, K. K. Sarma, P. Datta, and A. K. Mahanta, “Prediction of High Energy Particle Shower Sizes and Core Location using Artificial Neural Networks”, Springer Indian Journal of Physics, vol. 86, no. 1 pp. 77-84, 2012.

[2] K. K. Sarma and N. Mastorakis, “MIMO System with GA DFE-ANFIS Framework”, *NAUN International Journal of Fuzzy Systems and Advanced Applications*, vol. 1, pp. 55-60, 2014.

[3] D. S. Swami, K. K. Sarma and N. Mastorakis, “A Chaos based PN Sequence Generator for Direct-Sequence Spread Spectrum Communication System”, *NAUN International Journal of Circuits, Systems and Signal Processing*, vol.8, pp. 351-360, 2014.

[4] M. Baruah, K. K. Sarma and N. Mastorakis, ”NARMA Equalizer for Nonlinear ITU channels with Modified Feedback”, NAUN International Journal of Communications, vol. 10, pp. 49- 56, 2016.

[5] Y. LeCun,Y. Bengio and G. Hinton, “Deeplearning”, Nature, vol. 521 no. 7553, pp: 436444, May 2015.

[6] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks”, In Advances in Neural Information Processing Systems (NIPS) 25, pp. 10971105, 2012.

[7] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition”, in proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770778, 2016.

[8] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards real-time object detection with region proposal networks”, In Advances in Neural Information Processing Systems (NIPS) 28, pp. 9199, 2015.

[9] J. Long, E. Shelhamer, and T. Darrell, “Fully convolutional networks for semantic segmentation” in proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 34313440, 2015.

[10] H. C. Burger, C. J. Schuler, and S. Harmeling, “Image denoising: Can plain neural networks compete with BM3D?”, in proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), pp. 23922399, 2012.

[11] K. Zhang, W. Zuo, Y. Chen, D. Meng, and Lei Zhang, “Beyond a Gaussian Denoiser: Residual Learning of Deep CNN for Image Denoising”, IEEE Transactions on Image Processing, vol. 26, no. 7, pp. 31423155, May 2017.

[12] S. Lefkimmiatis, “Non-local Color Image Denoising with Convolutional Neural Networks”, in proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 58825891, 2017.

[13] X. Li, C. Li, P. Wang, Z. Men and H. Xu, “SAR ATR based on dividing CNN into CAE and SNN”, in proceedings of IEEE 5th Asia-Pacific Conference on Synthetic Aperture Radar (APSAR), pp. 676-679, 2015.

[14] V. Pascal et al., “Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion”, Journal of Machine Learning Research, vol. 11, pp. 3371-3408, 2010.

Mridusmita Sharma, currently Senior Visvesvaraya Scholar at the Department of Electronics and Communication Engineering, GUIST, Gauhati University, Guwahati, Assam, India is awaiting the award of PhD degree from the same university. Earlier she completed her MTech from Gauhati University with specialization in Signal Processing in 2014. Previously she completed BE in Electronics and Telecommunication Engineering from the same university in 2012. She has co-authored a number of research papers published in peer reviewed journals and proceedings of international conference. Her areas of interest includes speech processing, soft-computational modeling of speech and related phenomena, human computer interaction etc.

Dr. Kandarpa Kumar Sarma (SM IEEE, FIETE), currently Professor and Head, Department of Electronics and Communication Engineering, GUIST, Gauhati University, Guwahati, Assam, India obtained PhD in the area of Wireless Communication from Indian Institute of Technology Guwahati, Assam, India in 2011. Earlier, he obtained MTech from the same institute in 2005. His areas of interest includes deep learning, wireless and mobile communication, speech processing and antenna design.

NIKOS E. MASTORAKIS (SM’99) received the B.Sc. and M.Sc. (Diploma) degrees in electrical engineering and the Ph.D. degree in electrical engineering and computer science from the National Technical University of Athens, Greece, and the B.Sc. (Ptychion) degree in pure mathematics from the National University of Athens, Greece. He has also studied Medicine from the Medical School of Athens, National University of Athens. He has served as a Special Scientist on Computers and Electronics in the Hellenic Army General Staff from1993to 1994 and taught several courses in the Electrical and Computer Engineering Department, National Technical University of Athens, from 1998 to 1994. He was a Visiting Professor with the University of Exeter, U.K., from 1997 to 1998, the Technical University of Sofia, Bulgaria, from 2003 to 2004, and the University of Salerno, Italy, from 2014 to 2015. Currently, he is a Professor with the Industrial Engineering Department, Technical University of Sofia, and the Electrical Engineering and Computer Science Department, Hellenic Naval Academy, Greece. He is the Author of many papers in international journals and conferences. He is also an Honorary Professor with the Technical University of Cluj-Napoca, Romania, and Obuda University, Budapest, Hungary. He has also received the Prize of Excellence from the Romanian Academy of Science, Bucharest, Romania. He is the Evaluator of many European research projects and the Editor in various international journals.