Online Signature Denoising using Deep Autoencoder

K. Sasirekha
Department of Computer Science
Periyar University, Salem
Tamilnadu, India
ksasirekha7@gmail.com

R. Ravikumar
Department of Computer Science
Periyar University, Salem
Tamilnadu, India
ravimphilcs@gmail.cim

K. Thangavel
Department of Computer Science
Periyar University, Salem
Tamilnadu, India
drktvelu@yahoo.com

***Abstract*- The efficiency of an online signature authentication system depends on the quality of the online signature. Denoising of the online signature is indispensable to get a noise reduced signature. Deep learning models have been applied to a wide variety of denoising problems in recent years with great success. In this paper, deep autoencoder is proposed to use for removing Gaussian noise present in the online signature. Signal denoising can be performed with autoencoders by distorting the original signal data and add some noise to it that help in generalizing over the test set. A stacked denoising autoencoder (deep autoencoder) is a denoising autoencoder with multiple hidden layers and is trained layer by layer, by trying to minimize the reconstruction error. In this research work, experiments have been conducted on the MCYT online signature dataset. The Peak Signal to Noise Ratio (PSNR), Root Mean Square Error (RMSE), Mean Square Error (MSE), Maximum Error rate (MAXERR) and L2RAT have been computed and compared in order to evaluate the performance of the proposed method.**

Keywords- MCYT, Online Signature, Denoising, Deep autoencoder

1. INTRODUCTION

In the current scenario, biometrics is widely exploited for human authentication which is associated with the use of unique physiological or behavioral characteristics of an individual (Sutcu, Tabassi, Sencar, & Memon, 2013). However, biometric identification has eventually a much broader relevance as computer interface more natural. Knowing the person with whom we are conversing is an important part of human interaction and one expects computers of the future to have the same capabilities (Sasirekha & Thangavel, 2016). A number of biometric traits have been developed and are used to authenticate the person's identity. The idea is to use the unique characteristics such as the face, iris, fingerprint, online signature, etc. The method of identification based on biometric characteristics is preferred over traditional passwords and token-based method as they provide non-repudiation, a higher level of security and are more convenient for the user. Biometric technologies are thus defined as the "automated methods of identifying or authenticating the identity of a living person based on a physiological or behavioral characteristic". Typically, signature verification is a common behavioral biometric to identify human beings for purposes of verifying their identity. Signatures are particularly useful for identification of a particular person because each person's signature is highly unique, especially if the dynamic properties of the signature are considered in addition to the static features of the signature (Nagasundara, Guru, & Manjunath, 2012). Even if skilled forgers can accurately reproduce the shape of signatures, but it is unlikely that they can simultaneously reproduce the dynamic properties as well. Signature verification is split into two categories according to the available data in the input: offline (static) and online (dynamic). The input of offline signature verification system is the image of signature and is useful in automatic verification of signatures found on bank checks and documents. Some of the examples of the offline signature are shown in figure 1.

Signatures that are captured by data acquisition devices like pressure-sensitive tablets are shown in figure 2. Generally, the online signature is used in real time applications like credit card transactions, protection of small personal devices (e.g. PDA), and authorization of computer users for accessing sensitive data or programs, and authentication of individuals for access to physical devices or buildings.

Figure 1. Offline Signatures

Figure 2. Online Signature

Noise reduction is the process of removing noise from an online signature. All recording devices, both analog and digital, have traits that make them susceptible to noise. Noise can be random or white noise with no coherence, or coherent noise introduced by the device's mechanism or processing algorithms. Denoising of the online signature is indispensable to get a noise reduced signature. Deep learning models are representation learning models that are composed of multiple information processing layers to learn representations of data with multiple levels of abstraction (Yann, Yoshua, & Geoffrey, 2015). Auto-associative neural networks or autoencoders were introduced for learning more about the structure of data without using any labels i.e. to enable unsupervised learning (Chen, Xu, Weinberger, & Sha, 2012).

1.1. Motivation

Recently, in the domain of signal processing deep learning models is one promising avenue of research which automatically extracts complex representations from raw data. In particular, the deep autoencoder model has been applied to a wide variety of denoising problems with great success.

1.2. Contribution

In this work, a stacked denoising autoencoder with multiple hidden layers is trained layer by layer to minimize the reconstruction error between the original input and reconstructed signature.

1.3. Paper Organization

Section 2 elaborately discusses the related work of denoising using deep autoencoder. In section 3, the proposed online signature denoising using deep autoencoder is presented. The experimental results are discussed in section 4. Finally, this paper concludes with some perspectives in section 5.

2. RELATED WORK

The problem of personal verification and identification is an actively growing area of research. The methods are numerous and are based on different personal characteristics. In (Arora, Malik, & Anil, 2015) Vineeta et al. have extracted the writing speed, pressure points, strokes, acceleration as well as the static characteristics of signatures. This leads to better accuracy because the dynamic characteristics are very difficult to imitate, but the system requires user co-operation and complex hardware. A method for verifying handwritten signatures where various static (e.g., height, slant, etc.) and dynamic (e.g., velocity, pen tip pressure, etc.) signature features are extracted and used to train several network topologies is presented (Trevathan, Read, & McCabe, 2008). A method for automatic handwritten signature verification relies on global features that summarize different aspects of signature shape and dynamics of signature production is discussed in (Meghdadi, Guler, & Majid, 940-950). Jain et al. (Jain AK, 2002) have used pressure sensitive tablet to capture signatures. After a fair amount of pre-processing (resampling, smoothing, and size normalization), several local features were extracted (x, y) coordinate divergences between two consecutive points, curvature, gray values in 9 x 9 neighborhood, absolute and relative speeds, etc. A number of signatures stroke was the only extracted global feature, which was later incorporated to the overall dissimilarity value. Dynamic programming algorithm was applied to align two signatures. The overall dissimilarity value between a test and template signatures was then calculated by linearly incorporating the alignment score, the deference of stroke numbers between the signatures, and the normalization factor. Three deferent criteria were investigated to authenticate the test signature: the minimum, the maximum, and the average dissimilarity values to the reference set signatures.

Rigoll et al. (Rigoll & Kosmala, 1998) have provided a comparison between online and off-line signature verification using Hidden Markov Models. Signatures used for either of the systems were from the same data set; hence while using signatures for the online verification system, all dynamic features were discarded and only the image of the signature was used. Seven different feature types were empirically tested for their discriminative capabilities. Although Rigoll et al. used discrete Hidden Markov Models, they didn't mention about the structure of the models. The Viterbi algorithm was used to compute the likelihood probability of a test signature belonged to a claimed writer's model. The system was tested on very small data set: 14 writers contributed to the data set with 20 signatures each, 16 of which were used for training each writer's model, and the remaining 4 (56 total) were used for testing. As for the forgery set, 60 forgeries were supplied by 10 forgers, where 40 of them were skilled forgeries. Each feature was evaluated for its discriminative power. Then empirically combined feature sets were tested in the same manner. The feature set of the bitmap, velocity, Fourier transform and pressure features yielded the best performance results of 1% equal error for the online system. For the on-line case, an equal error rate of 1.9% was obtained. Although good performance results are reported for these systems, the data sets are too small to give reliable performance numbers.

The preprocessing of online signature is indispensable to get an enhanced and noise signature. Different types of Deep Neural Networks have been used to de-noise the signals. The de-noising auto-encoder is a special type of fully connected feedforward neural networks that takes noisy input signals and outputs their de-noised version. DAEs are common in deep learning. They are used to learn robust low-dimensional features even when the inputs are perturbed with some noise (Kim & Smaragdis, 2015) and (Vincent, Larochelle, Lajoie, Bengio, & Manzagol, 2010). Feng et al. (Xue, Yaodong, & James, 2014) trained the acoustic models using the reconstructed features from the DAE, and speech recognition is performed. The proposed approach is evaluated on the CHIME-WSJ0 corpus, and shows a 16-25% absolute improvement on the recognition accuracy. Peng et al. (Peng, Hongrui, Ming, & Xiuling, 2015) have used Stacked DAE to enhance Electrocardiogram (ECG) signal. The ECG signals used for DAE training is initially corrupted by means of a stochastic mapping and then reconstructed to the initial uncorrupted signals. DAEs are stacked to build deep architecture, which will improve the expression with multi-level feature extraction. The method is evaluated on ECG signals from the MIT-BIH Arrhythmia Database. MIT-BIH Noise Stress Test Database was used for generating noise. Computer simulation results show that the proposed system performed better than typically used ECG denoising methods with significant improvement on signal to noise ratio and root mean square error. Xiong et al. (Xiong, Wang, Liu, Lin, Hou, & Liu, 2016) have proposed a contractive de-noising technique to improve the performance of current de-noising auto-encoders for ECG signal de-noising. Based on the Frobenius norm of the Jacobean matrix for the learned features with respect to the input, we develop a stacked contractive de-noising auto-encoder to build a deep neural network (DNN) for noise reduction, which can significantly improve the expression of ECG signals through multi-level feature extraction. The proposed method is evaluated on ECG signals from the bench-marker MIT-BIH Arrhythmia Database, and the noises come from the MIT-BIH noise stress test database. The experimental results show that the new CDAE algorithm performs better than the conventional ECG de-noising method, specifically with more than 2.40 dB improvement in the signal-to-noise ratio and nearly 0.075 to 0.350 improvements in the root mean square error. Tan et al. (Man-WaiMak & Tan, 2015) explored the potential of using deep learning for extracting speaker-dependent features for noise robust speaker identification. More specifically, an SNR-adaptive de-noising classifier is constructed by stacking two layers of restricted Boltzmann machines (RBMs) on top of a de-noising deep auto-encoder, where the top-RBM layer is connected to a soft-max output layer that outputs the posterior probabilities of speakers and the top-RBM layer outputs speaker-dependent bottleneck features. Both the deep auto-encoder and RBMs are trained by contrastive divergence, followed by back propagation fine-tuning. The auto-encoder aims to reconstruct the clean spectra of a noisy test utterance using the spectra of the noisy test utterance and it's SNR as input. Experimental results based on a noisy YOHO corpus show that the bottleneck features slightly outperform the conventional MFCC under low SNR conditions and that fusion of the two features lead to further performance gain, suggesting that the two features are complementary with each other. Sasirekha et al. (Sasirekha & Thangavel, 2014), have applied the different kinds of wavelet filters such as db1, db2, db3, sym2 sym4, coif2 and coif4 with different noise levels and they observed that the db2 outperformed. The peak signal-to-noise ratio, signal-to noise ratio, root mean square error and mean square error measures have been used to analyse the above filters. Jain, et al has used FVC 2002 and estimated the best error rate for a common threshold is 3.3% false rejects and 2.7% false accepts. Using the minimum dissimilarity value plus user dependent offset results in 2.8% false rejects and 1.6% false accepts (Jain AK, 2002).

Muramatsu, et al. (Muramatsu & Matsumoto, 2007) have taken SVC 2004 dataset for their experiment and have performed an evaluation of each distance from a feature and performed an evaluation of fusion models by combining different distances. These experimental results show that, by incorporating pen pressure and inclination features, one could improve the performance from an error rate of 5.79% to 3.61% for SVC task 2 with user-dependent threshold parameters, and from an error rate of 12.67% to 10.15% with a global threshold parameter. Gehring, et al. (Gehring, Miao, Metze, & Waibel, 2013) have used MFCC dataset and they performed experiments with bottleneck features and auto-encoder has been produced in terms of word error rate, relative improvements of 9.2% (Cantonese, ML training), 9.3% (Tagalog, BMMI-SAT training), 12% (Tagalog, confusion network combinations with MFCCs), and 8.7% (Switchboard) are achieved. They trained network with four autoencoders on subsets of the data and used the bottleneck features to set up a context-dependent system on the full 300-hour dataset. Using the feature from a network trained on 60 hours of speech lowered the word error rate from 39.0% to 36.1% (7.4% relative). Doubling the amount of training data for the neural network resulted in further improvement and produced a WER of 35.6%, which is an 8.7% relative gain over the MFCC baseline.

K. S. Manjunath et al. (Manjunath, Manjunath, Guru, & Somashekara, 2016) have analyzed MCYT and the experimental results confirm the effectiveness of writer dependent characteristics for online signature verification. The dataset contains a total of 2000 genuine signatures collected in two sessions and 1000 skilled forgeries which include 500 highly skilled forgeries. They have used 10 genuine signatures of every writer for training purpose and the remaining genuine and all skilled forgeries for testing. The obtained result was writer specific features and writer dependent classifier and common classifier for signature verification access of achieved result. D. S. Guru et al. (Guru, Nagasundara, & Manjunath, 2011)have used MCYT, for their experimentation, even if the number of features selected is increased in step of 1 up to 75, there was only marginal decrease in the error rate and hence they considered only 60 features and similarly with respect to skilled 20 and Random 20. They achieved a minimum error rate for 50 features. Nagasundara et al. (Manjunath, Guru, & Nagasundara, 2013) have used MCYT (online signature dataset) and speed up the identification processes which reduces the number of candidate hypotheses to be considered during matching by the identification algorithm. Kd-tree based indexing model is designed for online signature based person identification. The system is trained using 40%, 60% and 80% samples per user respectively.

3. DENOISING ONLINE SIGNATURE USING DEEP AUTOENCODER

3.1. Deep Learning Models

Deep learning models have been applied to a wide variety of signal processing problems in recent years with great success. At the heart of all deep learning, models are the domain-independent idea of using hierarchical layers of learned abstraction to efficiently accomplish high-level tasks. For a long time, neural network with more than one hidden layers was considered to be hard to train efficiently and gained popularity with the advent of various deep learning models such as

- Deep Autoencoders
- Restricted Boltzmann Machines (RBM)
- Convolutional Neural Networks (CNN)
- Recurrent Neural Networks (RNN)

3.2. Conventional Autoencoder

Autoencoder is trained to encode the input x into some representation r(x) so that the input can be reconstructed from that representation as in figure 3.

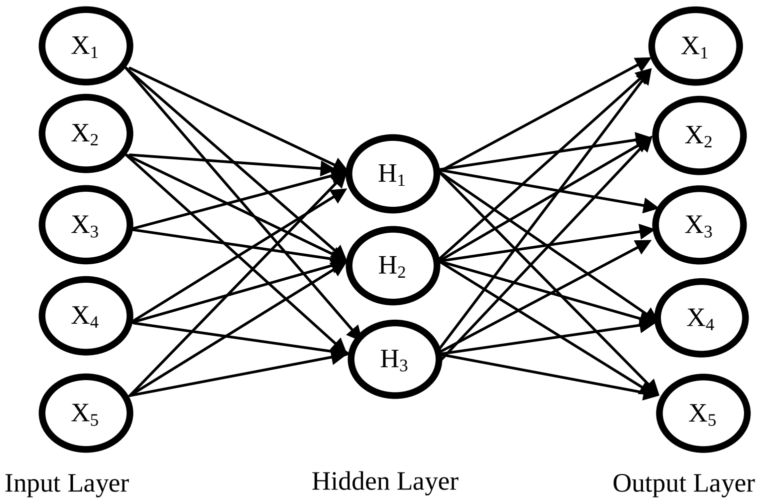

Figure 3. The Autoencoder architecture

Hence the target output of the autoencoder is same as the input itself. It consists of two parts, the encoder, and the decoder. The encoder part takes the raw input of the signal $x \in R^d$ and maps it into a hidden representation $h \in R^{d'}$ through a deterministic mapping as in equation (1).

$$h = \sigma(Wx + b) \tag{1}$$

where σ is the transfer function such as a sigmoid function or a rectified linear unit. W is a weight matrix and b is a bias vector. The latent representation h or code is then mapped back into a reconstruction r of the same shape as x. The mapping happens through a similar transformation as given in equation (2).

$$r = \sigma(W'h + b') \tag{2}$$

where, r should be seen as a prediction of x, given the encode h.

Autoencoders are trained to minimize reconstruction error between the raw inputs and reconstructed one. The reconstruction error can be measured in many ways, depending on the appropriate distributional assumptions on the input given the code. The conventional squared error is given as in equation (3).

$$L(x,r) = ||\mathrm{x} - \mathrm{r}||^2 \quad (3)$$

If the input x is interpreted as either binary or vectors of bit probabilities, then cross-entropy of the reconstruction can be used and is given as in the equation (4),

$$L(x,r) = -\sum_{k=1}^{d}[x_k log r_k + (1 - x_k) log(1 - r_k)] \quad (4)$$

3.3. Stacked Denoising Autoencoder

A denoising autoencoder is trained to reconstruct a clean input from a corrupted version of it as shown in figure 4. This is done by first corrupting the initial input S into N by adding noise to it. Then the corrupted input N then mapped, as with the basic autoencoder, to a hidden representation Y from which R is reconstructed. A schematic representation of the procedure is given in figure 4. As previously, the considered reconstruction error is either the squared loss or cross-entropy loss (Vincent, Larochelle, Lajoie, Bengio, & Manzagol, 2010). The denoising autoencoders are still minimizing the same reconstruction loss between a clean S and its reconstruction R from Y. So this still amounts to maximizing a lower bound on the mutual information between clean input S and representation Y. The difference is that Y is now obtained by applying deterministic mapping to a corrupted input.

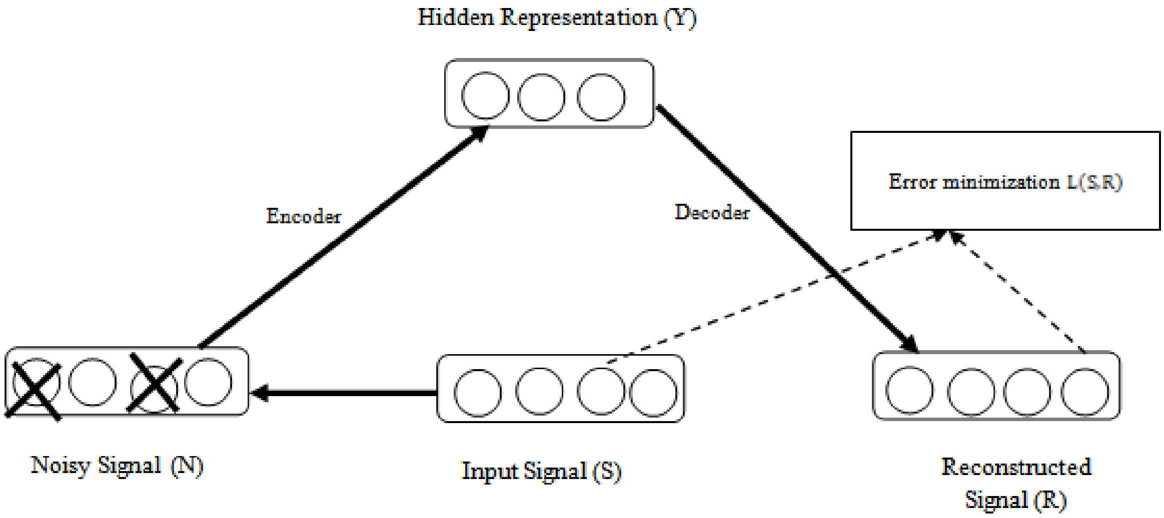

Figure 4. Denoising Autoencoder

The complete procedure for learning and stacking several layers of denoising autoencoders is shown in figure 5. After training a first level Denoising autoencoder its learned encoding function is used on clean input (5 (a)). The resulting representation is used to train a second level Denoising autoencoder (5(b)) to learn a second level encoding function. From there, the procedure can be repeated (5(c)).

3.4. Online Signature Denoising using Deep Autoencoders

Signature denoising can be performed with deep autoencoders by distorting the original signature data and add some noise to it that help in generalizing over the test set. The corrupted signature (N) is mapped to a hidden representation Y from which the noise reduced signature (R) is reconstructed. A schematic representation of the procedure is given in figure 6. The squared error is used to compute the error between corrupted signature (N) and reconstructed signature (R). Denoising procedure of Online Signature using Deep Autoencoder is presented in Algorithm 1

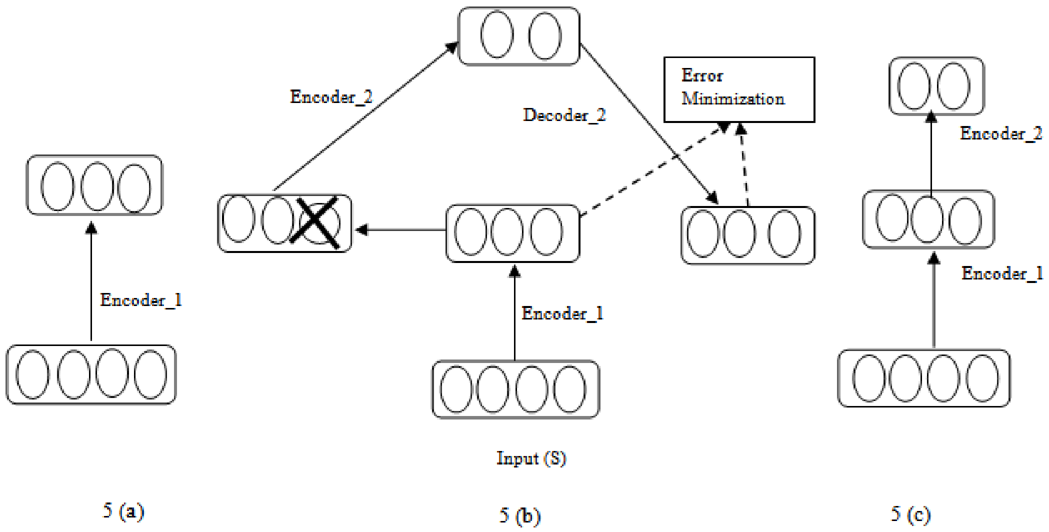

Figure 5. Stacking Denoising Autoencoders

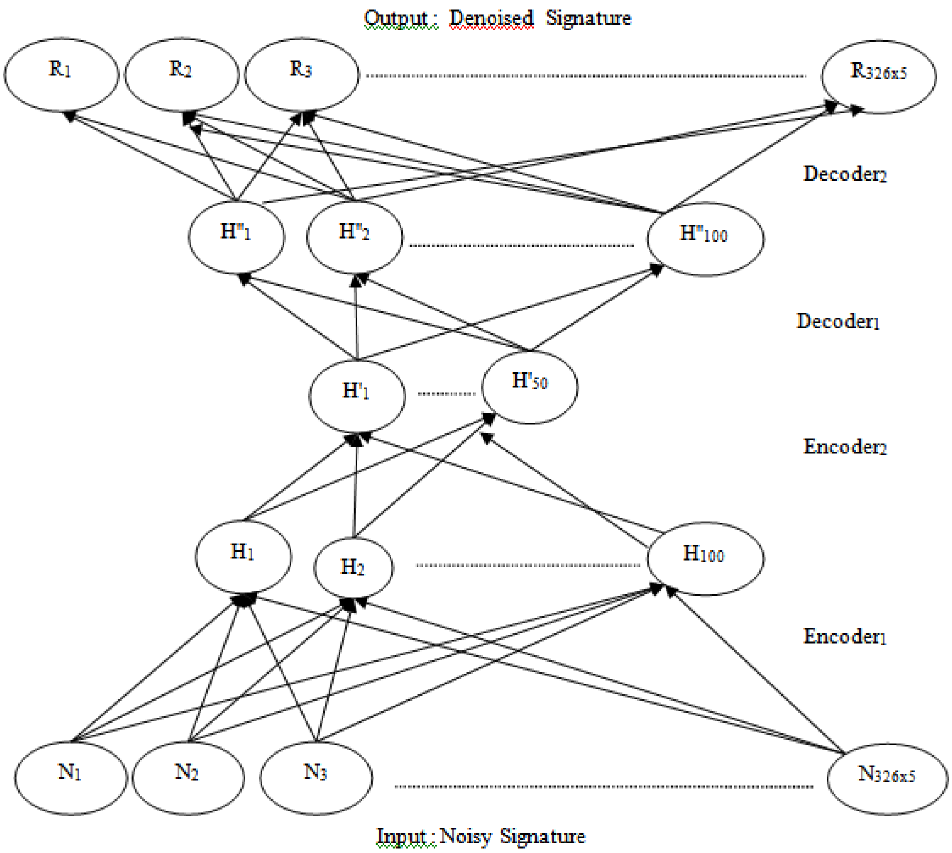

Figure 6. Online Signature Denoising using Deep Autoencoder

Algorithm 1: Denoising procedure of Online Signature using Deep Autoencoder

Input: Noisy Signal (n); Output: Denoised Signal (r)

Step 1: Corrupt the original input signal (Gaussian Noise)

Step 2: Encode (5) the corrupted signal to a hidden representation with an autoencoder through a deterministic mapping,

$$Encode\_1 = \sigma(W * n + b) \tag{5}$$

Step 3: The encoded signal from step 2 is then given as an input to the second autoencoder (6) for further compression through a deterministic mapping,

$$Encode\_2 = \sigma(W * Encode\_1 + b) \tag{6}$$

Step 4: The latent representation $Encode\_2$, or code is then mapped back into a reconstruction r of the same shape as n using the decoder function (7)

$$r = \sigma(W' * Encode\_2 + b') \tag{7}$$

Step 5: The squared error (8) is computed to minimize the reconstruction error between the original input and reconstructed one

$$L(n, r) = ||\mathrm{n} - \mathrm{r}||^2 \tag{8}$$

Step 6: Evaluate the performance using signal quality metrics such as PSNR, MSE, RMSE, MAXERR, and L2RAT

4. EXPERIMENTAL RESULTS AND DISCUSSION

The performance of the proposed Denoising Autoencoder for online signature denoising has been computed and compared in terms of encoder and decoder transfer functions. The proposed method has been implemented in MATLAB.

4.1. Dataset

The proposed method is tested on MCYT online signature dataset. It contains 16,500 signature files collected from 330 persons (Signature).

4.2. Performance Measures

The signal quality metrics such as Peak Signal to Noise Ratio, Mean Square Error, Root Mean Square Error, Maximum Error rate and Ratio of the squared norm are used to evaluate the performance (Sasirekha & Thangavel, 2014). The metrics are shown in table 1.

Table 1. Performance Measures

MEASURE	FORMULA
PSNR	$10\log_{10}\left(\frac{\mathrm{R}^2}{\mathrm{MSE}}\right)$
MSE	$\frac{\sum_{\mathrm{m,n}}[\mathrm{I}_1(\mathrm{m,n}) - \mathrm{I}_2(\mathrm{m,n})]^2}{\mathrm{m} * \mathrm{n}}$
RMSE	$\sqrt{\frac{\sum_{\mathrm{m,n}}[\mathrm{I}_1(\mathrm{m,n}) - \mathrm{I}_2(\mathrm{m,n})]^2}{\mathrm{m} * \mathrm{n}}}$
MAXERR	$\frac{\sum_{\mathrm{m,n}}[\mathrm{I}_1(\mathrm{m,n}) - \mathrm{I}_2(\mathrm{m,n})]}{\mathrm{m} * \mathrm{n}}$
L2RAT	$\frac{Norm(\mathrm{I}_2(\mathrm{m,n}))^2}{Norm(\mathrm{I}_1(\mathrm{m,n}))^2}$

Where I1 is the input signal, I2 is the denoised signal, m and n are the number of rows and columns in the signal; respectively, and R is the maximum fluctuation in the input signal data type.

4.3. Results of Online Signature Denoising using Deep Autoencoders

The transfer function for the autoencoder is specified in table 2. The mean value of PSNR, MSE, RMSE, MAXERR, and L2RAT of MCYT signature database is given in table 3 and table 4 respectively.

Table 2. Transfer Functions for Autoencoder

TRANSFER FUNCTION	FORMULA
Logsig	$f(z) = \frac{1}{1+e^{-z}}$
Satlin	$f(z) = \begin{cases} 0, & if\ 0 \leq 0 \\ z, & if\ 0 < z < 1 \\ 1, & if\ z \geq 1 \end{cases}$
Purelin	$f(z) = z$

The quantitative measures such as PSNR, MSE, RMSE, MAXERR, and L2RAT of DAE for Logsig encoder function with various decoder functions such as Logsig, Satlin and Purelin has been computed and compared in table 3.

Table 3. Results of DAE with Logsig Encoder Function

		PSNR	MSE	RMSE	MAXERR	L2RAT
Decoder Transfer Function	Logsig	38.90	8.36	2.89	64.88	0.99
	Satlin	25.13	199.10	14.11	31.10	1.00
	Purelin	22.11	399.17	19.97	41.63	0.98

Table 4. Results of DAE with Satlin Decoder Function

		PSNR	MSE	RMSE	MAXERR	L2RAT
Decoder Tranfer Function	Logsig	25.03	203.93	14.28	49.40	0.99
	Satlin	18.32	956.90	30.93	89.42	0.97
	Purelin	19.72	692.78	26.32	85.94	0.98

The quantitative measures such as PSNR, MSE, RMSE, MAXERR, and L2RAT of DAE for Satlin encoder function with various decoder functions such as Logsig, Satlin and Purelin has been computed and compared in table 3.

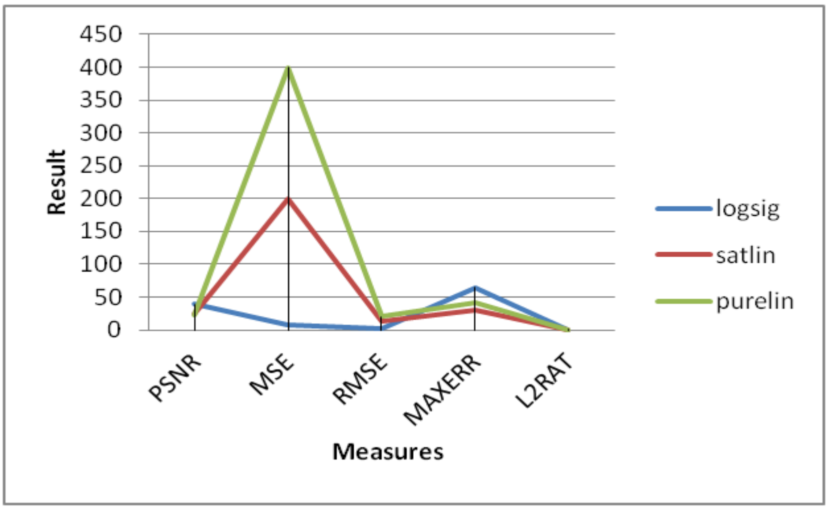

Figure 7. Results of DAE with Logsig Encoder Function

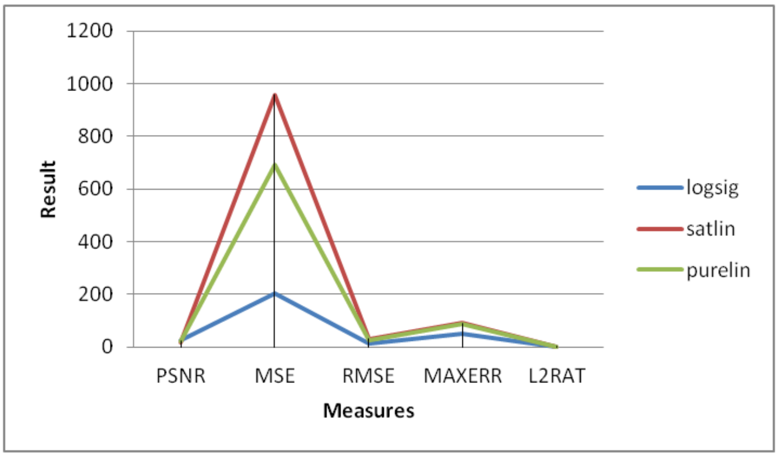

Figure 8. Results of DAE with Satlin Decoder Function

Figure 7and Figure 8 shows that the performance of the DAE with Logsig transfer function for both encoder and decoder gives better performance than other transfer functions. The PSNR of the DAE with logsig encoder and decoder function is 38.90% which is higher than other transfer functions. The DAE provide RMSE of 2.89% for logsig, 14.11% for satlin, and 19.97% for purelin transfer function.

5. CONCLUSION

Deep learning models have been applied to a wide variety of denoising problems in recent years with great success. In this research work, deep autoencoder is used for removing Gaussian noise present in the online signature. Online signature is denoised with deep autoencoders by distorting the original signature data and add some noise to it that help in generalizing over the test set. In this research work, a two-layer stacked denoising autoencoder is used with multiple hidden layers and is trained layer by layer, by trying to minimize the reconstruction error. Experiments have been conducted on the MCYT online signature dataset. The Peak Signal-to-Noise Ratio, Root Mean Square Error, Mean Square Error, Maximum Error rate and L2RAT have been computed and compared. The quantitative measures show that DAE with Logsig transfer produced better results than Stalin and Purelin transfer functions.

ACKNOWLEDGMENT

Authors would like to thank UGC, New Delhi for the financial support received under UGC Major Research Project No. 43-274/2014(SR).

REFERENCES

Arora, Malik, V., & Anil. (2015). Review Paper on Signature Recognition. International Journal for Research in Applied Science & Engineering Technology (IJRASET) , 3 (4).

Chen, M., Xu, Z., Weinberger, K., & Sha, F. (2012). Marginalized denoising autoencoders for domain adaptation. ICML .

Gehring, J., Miao, Y., Metze, F., & Waibel, A. (2013). Extracting deep bottleneck features using stacked auto-encoders. Proceedings of ICASSP, (pp. 3377–3381).

Guru, D., Nagasundara, K., & Manjunath, S. (2011). Indexing of online signatures. International Journal of Machine Intelligence , 289-294.

Jain AK, G. F. (2002). On-line Signature Verification. Pattern Recognition , 35, 2963–2972.

Jain, A., Griess, F., & Connell, F. (2002). On-line Signature Verification. Pattern Recognition , 35, 2963-2972.

Kim, M., & Smaragdis. (2015). Adaptive denoising autoencoders: A fine-tuning scheme to learn from test mixtures. Proc. LVA/ICA, (pp. 100–107).

Manjunath, K., Manjunath, S., Guru, D., & Somashekara, M. (2016). Online signature verification based on writer dependent feature and classifiers. Pattern Recognition letters , 80, 129-136.

Manjunath, S., Guru, D., & Nagasundara, K. (2013). User dependent features in online signature verification. Multimedia Processing, Communication and Computing Applications , 229-240.

Man-WaiMak, & Tan, Z. (2015). Bottleneck Features From SNR-Adaptive Denoising Deep Classifier For Speaker Identification. Signal and Information Processing Association Annual Summit and conference (APSIPA).

Meghdadi, Guler, I., & Majid. (940-950). A different approach to off-line handwritten signature verification using the optimal dynamic time warping algorithm. Digital Signal Process , 18 (6), 2008.

Muramatsu, D., & Matsumoto, T. (2007). Effective of Pen Pressure, Azimuth, and Altitude Features for Online Signature Verification. Proceedings of 2007 International Conference on Advances in Biometrics, (pp. 503–512).

Nagasundara, K., Guru, D., & Manjunath, S. (2012). Feature selection and indexing of online signatures. 408-414.

Peng, X., Hongrui, W., Ming, L., & Xiuling. (2015). Denoising Autoencoder for Electrocardiogram Signal Enhancement. Journal of Medical Imaging and Health Informatics , 5 (8), 1804-1810.

Rigoll, G., & Kosmala, A. (1998). A Systematic Comparison Between On-Line and Off-Line Methods for Signature Verification with Hidden Markov Models. In Proceedings of International Conference on Pattern Recognition, (pp. 1755-1757).

Sasirekha, K., & Thangavel, K. (2016). A Novel Feature Extraction Algorithm from Fingerprint Image in Wavelet Domain. Intelligence, Cyber Security, and Computational Models, (pp. 135-143).

Sasirekha, K., & Thangavel, K. (2014). A novel wavelet-based thresholding for denoising fingerprint image. International Conference on Electronics, Communication and Computational Engineering (ICECCE), (pp. 119-124).

Signature . Database: https://atvs.ii.uam.es/atvs/mcyt100s.html.

Sutcu, Y., Tabassi, E., Sencar, H., & Memon, N. (2013). What is biometric information and how to measure it? IEEE International Conference on Technologies for Homeland Security (HST) , 12 – 14.

Trevathan, J., Read, W., & McCabe, A. (2008). Neural network-based handwritten signature verification. Journal of Computers , 3 (8), 9-22.

Vincent, P., Larochelle, H., I, L., Bengio, Y., & Manzagol, P. (2010). Stacked DenoisingAutoencoders: learning useful representations in a deep network with a local denoising criterion. Journal of Machine Learning Research , 11, 3371 – 3408.

Xiong, P., Wang, H., Liu, M., Lin, F., Hou, Z., & Liu, X. (2016). A stacked contractive denoising auto-encoder for ECG signal denoising. Physiological Measurement , 37 (12).

Xue, F., Yaodong, Z., & James, G. (2014). Speech Feature Denoising And Dereverberation Via Deep Autoencoders For Noisy Reverberant Speech Recognition. IEEE International Conference on Acoustic, Speech and Signal Processing (ICASSP).

Yann, L., Yoshua, B., & Geoffrey, H. (436–444). Deep learning. Nature , 2015.