This is the Pre-Published Version.

The following publication F. N. Khan and A. P. T. Lau, "Robust and efficient data transmission over noisy communication channels using stacked and denoising autoencoders," in China Communications, vol. 16, no. 8, pp. 72-82, Aug. 2019 is available at https://doi.org/10.23919/JCC.2019.08.007.

Robust and efficient data transmission over noisy communication channels using stacked and denoising autoencoders

Faisal Nadeem Khan*, and Alan Pak Tao Lau
Department of Electrical Engineering, The Hong Kong Polytechnic University, Kowloon, Hong Kong (SAR)
*Corresponding author: fnadeem.khan@yahoo.com

Abstract: We study the effects of quantization and additive white Gaussian noise (AWGN) in transmitting latent representations of images over a noisy communication channel. The latent representations are obtained using autoencoders (AEs). We analyze image reconstruction and classification performance for different channel noise powers, latent vector sizes, and number of quantization bits used for the latent variables as well as AEs' parameters. The results show that the digital transmission of latent representations using conventional AEs alone is extremely vulnerable to channel noise and quantization effects. We then propose a combination of basic AE and a denoising autoencoder (DAE) to denoise the corrupted latent vectors at the receiver. This approach demonstrates robustness against channel noise and quantization effects and enables a significant improvement in image reconstruction and classification performance particularly in adverse scenarios with high noise powers and significant quantization effects.

Keywords: Communication channels; data compression; deep learning; autoencoders; denoising autoencoders.

1. Introduction

In communications research, machine learning techniques have appeared as a new direction of innovation to cope with many emerging challenges. These techniques have already started playing a pivotal role in various networking tasks such as traffic load forecasting, capacity optimization, autonomous fault detection and analysis, etc. [1]. Recent developments in deep machine learning have further motivated the researchers to fully exploit their potential in communication systems [2][3]. Deep learning architectures, e.g., deep autoencoders (AEs), deep neural networks (DNNs), deep convolutional neural networks (CNNs), deep recurrent neural networks (RNNs), etc., have been applied successfully for various tasks in communication networks and have achieved state-of-the-art results in many cases [4]-[9].

One of the major workhorses of communication systems is data compression. For images, discrete cosine transform (DCT) and wavelet transform are standard compression techniques to reduce the actual amount of data to be transmitted over the communication links [10][11]. Over the past few years, deep learning methods have also been considered for image compression task and have already shown quite impressive results [12][13]. Among these methods, reduced-complexity deep AEs have gained significant attention recently [14]-[20]. This is due to the fact that in many practical scenarios, the mobile devices or wireless sensor network (WSN) nodes may be resource-limited in terms of processing power, storage capacity (i.e., memory), battery life, etc., and thus cannot afford complex image compression/decompression techniques. The use of relatively simple deep AE architectures may be an attractive alternative in such cases.

In case of AEs, the latent representation, or latent vector, of the corresponding image can be simply seen as a compressed description of the original image. AEs-based compression approaches learn statistical regularities in the data and thus may hold key advantage when a pair of transmitter and receiver is communicating information that corresponds to a particular domain with certain specific characteristics/features, e.g., transmission of sensing data between two nodes in a WSN or communication of certain types of files/images between two dedicated servers in data centres. The previously reported works in this context assumed idealized communication

© 2019 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

scenarios such as

(i) Availability of original transmitted latent vectors (i.e., compressed representations) at the receiver for decompression while completely discarding the effects of real-world channel impairments such as noise.
(ii) Neglecting the effects of digitization/quantization of latent variables into binary data in practical communication systems.
(iii) Ignoring the effects of quantization of deep AEs' parameters (i.e., weights and biases), which might be inevitable while deploying these architectures on practical resource-limited (in terms of storage size and computational power) devices like mobile or embedded devices.

In this work, we consider the scenario in which modified National Institute of Standards and Technology (MNIST) images are compressed/transformed into their latent representations through AEs. The resulting latent vectors are digitized into binary data and transmitted over a communication channel corrupted by additive white Gaussian noise (AWGN) and the received latent vectors are then used for the recovery of original images. We then propose the use of an additional denoising autoencoder (DAE) to restore the fidelity of latent vectors at the receiver and hence improve image recovery and classification performance. More specifically, the main highlights of this work are as follows:

- We investigate quantitatively the effect of AWGN in communication channels on the image recovery performance of AEs-based compression with latent vectors of various sizes.
- We analyze the effect of latent variables quantization on image recovery performance. Such an analysis is particularly helpful in determining the trade-off between compression ratio and image reconstruction quality.
- We propose denoising of received latent vectors corrupted by AWGN and/or quantization noise through the use of an additional DAE at the receiver for enabling more robust image recovery and classification performance in noisy communication channels.
- We study the effect of quantization of AEs' parameters on image recovery performance and propose the use of nonuniform quantization for significantly reducing the storage requirements of AEs' parameters while still offering good image reconstruction quality.

The rest of the paper is organized as follows. Section 2 presents the proposed stacked AE and DAE based system model. The architectures and mathematical overview of stacked AE and DAE, their training and optimization procedures, and their specific functionalities within the proposed framework are also discussed in detail. In Section 3, we provide simulation results and quantitatively as well as visually compare the image reconstruction performances of systems without and with DAE. Section 4 concludes the paper and outlines potential future work.

2. System Model and Proposed Approach

The system model employed in our analysis is shown in Fig. 1. As clear from the figure, we use two different AEs in the proposed approach. The first AE is a stacked AE with two hidden layers while the second is a DAE. In this work, we employ MNIST database comprising of 55,000 training and 10,000 testing images for the learning and evaluation of AEs, respectively. Each 28 × 28 image in the training or testing data set is represented as a one-dimensional vector $\mathbf{x}$ of length L = 784 by concatenating all of its columns. The training data set $\{\mathbf{x}(1), \mathbf{x}(2), ..., \mathbf{x}(N)\}$ comprising of N = 55,000 image vectors is first utilized for the unsupervised training of stacked AE in a greedy layer-wise manner. The training of stacked AE is assumed to take place offline and in one location without the detrimental effects of quantization and channel noise. After

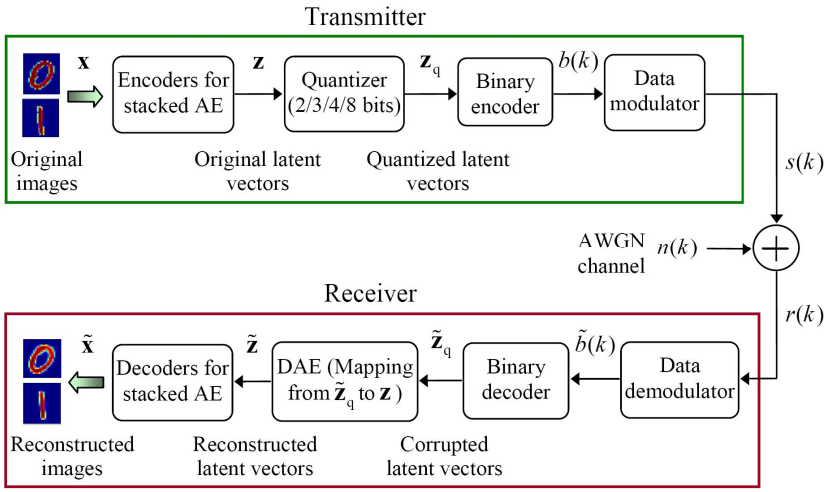


Fig. 1. Schematic diagram of system model used for the analysis.

convergence, the encoders and decoders of stacked AE are separated to form the components of transmitter and receiver, respectively. The first part of stacked AE, namely autoencoder-1, is a three-layer neural network that attempts to reconstruct its input vectors **x** at its output as shown in Fig. 2(a). For this purpose, the encoder of autoencoder-1 maps an input vector $\mathbf{x}(i) \in \mathbb{R}^L$ to a hidden representation $\mathbf{z}_1(i) \in \mathbb{R}^K$ (where $K << L$) while the decoder (shown in grey color) aims to reverse this mapping such that the output vector $\tilde{\mathbf{x}}(i) \in \mathbb{R}^L$ is an approximation of the original input vector $\mathbf{x}(i)$. The vectors satisfy

$$\mathbf{z}_1(i) = \sigma_1\left(\mathbf{W}_1\mathbf{x}(i) + \mathbf{b}_1\right) \qquad (1)$$

$$\tilde{\mathbf{x}}(i) = \sigma_1'\left(\mathbf{W}_1'\mathbf{z}_1(i) + \mathbf{b}_1'\right) \qquad (2)$$

where $\mathbf{W}_1$ and $\mathbf{W}_1'$ are the weight matrices for the encoder and decoder, respectively, while $\mathbf{b}_1$ and $\mathbf{b}_1'$ are the bias vectors for the encoder and decoder, respectively. On the other hand, $\sigma_1(\cdot)$ and $\sigma_1'(\cdot)$ are the nonlinear *activation functions* for the encoder and decoder, respectively, and are chosen to be sigmoid function in our case, i.e.,

$$\sigma_1(\upsilon) = \sigma_1'(\upsilon) = \frac{1}{1+e^{-\upsilon}}. \qquad (3)$$

To train the autoencoder-1 is to optimize all the parameters $\theta_1 = \{\mathbf{W}_1, \mathbf{W}_1', \mathbf{b}_1, \mathbf{b}_1'\}$ such that the difference between the original input vectors **x** and the reconstructed output vectors $\tilde{\mathbf{x}}$ is minimized over the whole training data set. We optimized commonly-used mean square error (MSE) objective function (also called *loss function* in machine learning literature) given as

$$\begin{aligned} E_1 &= \frac{1}{N}\sum_{i=1}^{N} E_1(i) = \frac{1}{N}\sum_{i=1}^{N}\left\|\mathbf{x}(i) - \tilde{\mathbf{x}}(i)\right\|^2 \\ &= \frac{1}{N}\sum_{i=1}^{N}\left\|\mathbf{x}(i) - \sigma_1'\left(\mathbf{W}_1'\left(\sigma_1\left(\mathbf{W}_1\mathbf{x}(i) + \mathbf{b}_1\right)\right) + \mathbf{b}_1'\right)\right\|^2 \end{aligned} \qquad (4)$$

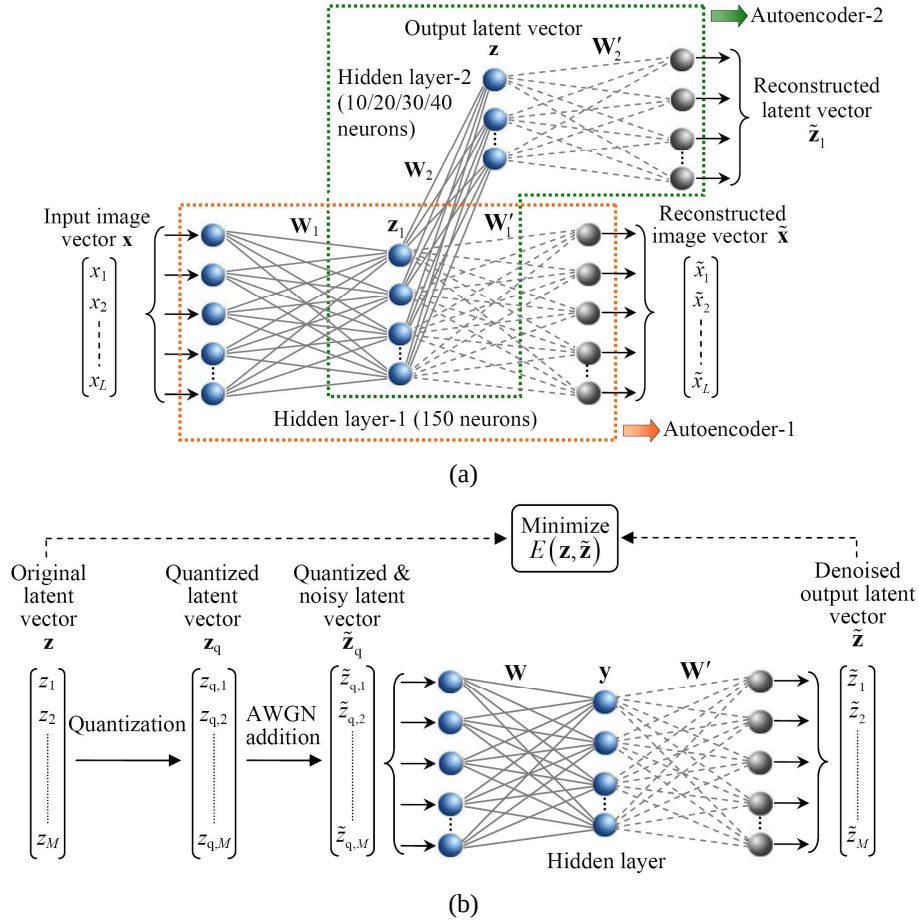


Fig. 2. Conceptual diagrams of (a) stacked AE and (b) DAE used in the proposed approach.

and used gradient descent optimization procedure during the training. Once autoencoder-1 successfully learns the mappings given by Eqs. (1) and (2), it is considered to be trained. Next, all the original input vectors $\mathbf{x}$ in the training data set are passed through the encoder of trained autoencoder-1. Since the size of hidden layer-1 (i.e., 150) is chosen to be much less than the size of vectors $\mathbf{x}$ (i.e., 784), the mapping learned by encoder of autoencoder-1 provides compressed latent representations $\mathbf{z}_1$ (of size K = 150) of the original input vectors $\mathbf{x}$ at the encoder output.

Next, the data set $\{\mathbf{z}_1(1), \mathbf{z}_1(2), ..., \mathbf{z}_1(N)\}$ comprising of N = 55,000 latent vectors $\mathbf{z}_1$ is used to train the autoencoder-2 in a similar fashion, where the size M of hidden layer-2 is selected to be much less than the size of latent vectors $\mathbf{z}_1$ (i.e., $M << K$) as shown in Fig. 2(a). The encoder of autoencoder-2 maps an input latent vector $\mathbf{z}_1(i) \in \mathbb{R}^K$ to a hidden representation $\mathbf{z}(i) \in \mathbb{R}^M$ while the decoder reverses this mapping such that the output vector $\tilde{\mathbf{z}}_1(i) \in \mathbb{R}^K$ is an approximation of the input latent vector $\mathbf{z}_1(i)$. In this way, the autoencoder-2 learns an even more compressed representation of the original image vectors $\mathbf{x}$. Once the training of autoencoder-2 is complete, all 55,000 latent vectors $\mathbf{z}_1$ are passed through its encoder to obtain the data set $\{\mathbf{z}(1), \mathbf{z}(2), ..., \mathbf{z}(N)\}$ containing final compressed representations of the initial input vectors $\mathbf{x}$. We consider four different sizes for hidden layer-2, i.e., 10, 20, 30 and 40 as a way to realize various compression ratios for the original images. Note that there is no fixed rule for choosing the optimum number of hidden layers in a stacked AE and typically the selection is made via

experimentation (e.g., starting with just one hidden layer and iteratively increasing the number of layers until the image reconstruction quality in terms of MSE starts to decrease). This is known as the *hyperparameter* of a stacked AE. In our case, the number of hidden layers for the stacked AE is optimized to be 2 as shown in Fig. 2(a).

To cope with quantization and communication channel noise effects, we further implement a DAE [21][22], which is a variant of basic AE with the difference that it corrupts the original inputs (e.g., by adding noise into them) before mapping them to the hidden representation. A DAE is trained to recover the original inputs from their corrupted versions. In this work, we employ a DAE at the receiver to undo the effects of quantization and channel noise on the original transmitted latent vectors **z**. The structure of a single hidden layer DAE used in our work is shown in Fig. 2(b). The set $\{\mathbf{z}(1), \mathbf{z}(2), ..., \mathbf{z}(N)\}$ of 55,000 latent vectors of the training data set (obtained using stacked AE as discussed earlier) is used for the training of DAE. For this purpose, we first quantize the latent vectors **z** to $\mathbf{z}_q$ and then Gaussian noise is added to $\mathbf{z}_q$ for generating the corrupted latent vectors $\tilde{\mathbf{z}}_q$. The vectors $\tilde{\mathbf{z}}_q$ are then mapped, as in case of a basic AE, to a hidden representation **y** from which we reconstruct the output vectors $\tilde{\mathbf{z}}$. The vectors satisfy

$$\mathbf{y}(i) = \sigma\left(\mathbf{W}\tilde{\mathbf{z}}_q(i) + \mathbf{b}\right) \qquad (5)$$

$$\tilde{\mathbf{z}}(i) = \sigma'\left(\mathbf{W}'\mathbf{y}(i) + \mathbf{b}'\right) \qquad (6)$$

where $\sigma(\cdot)$ and $\sigma'(\cdot)$ are the activation functions, **W** and $\mathbf{W}'$ are the weight matrices, and **b** and $\mathbf{b}'$ are the bias vectors. The DAE is trained to minimize the reconstruction error

$$\begin{aligned} E &= \frac{1}{N}\sum_{i=1}^{N} E(i) = \frac{1}{N}\sum_{i=1}^{N} \left\|\mathbf{z}(i) - \tilde{\mathbf{z}}(i)\right\|^2 \\ &= \frac{1}{N}\sum_{i=1}^{N} \left\|\mathbf{z}(i) - \sigma'\left(\mathbf{W}'\left(\sigma\left(\mathbf{W}\tilde{\mathbf{z}}_q(i) + \mathbf{b}\right)\right) + \mathbf{b}'\right)\right\|^2 \end{aligned} \qquad (7)$$

over the whole training data set, i.e., to have $\tilde{\mathbf{z}}$ as close as possible to **z**. The above process enables DAE to effectively learn the mapping between corrupted latent vectors $\tilde{\mathbf{z}}_q$ and uncorrupted input latent vectors **z** of the training data set.

In the proposed approach, the encoder parts of trained stacked AE are deployed at the transmitter while the decoder parts are located within receiver. On the other hand, the trained DAE (excluding the quantization and Gaussian noise addition steps) is utilized inside the receiver. In the testing phase, the latent vectors **z** for the testing data set $\{\mathbf{x}(1), \mathbf{x}(2), ..., \mathbf{x}(P)\}$ comprising of P = 10,000 image vectors **x** are first obtained using the encoder parts of trained stacked AE. Next, the elements of **z** are quantized into 2/3/4/8 bits and the corresponding binary data sequence $b(k)$ is modulated into formats such as binary phase-shift keying (BPSK). The resulting signals $s(k)$ are transmitted over an AWGN channel which adds noise $n(k)$ into the signals. The noise power is adjusted for achieving different bit-error ratios (BERs). The received noisy signals $r(k)$ are demodulated and the detected binary data sequence $\tilde{b}(k)$ is then used to generate vectors $\tilde{\mathbf{z}}_q$, which are a corrupted version of the original latent vectors **z**. Next, the pretrained DAE in the receiver is utilized to obtain estimates $\tilde{\mathbf{z}}$ of the original transmitted latent vectors **z** from the received corrupted versions $\tilde{\mathbf{z}}_q$. Finally, the decoder parts of stacked AE are used to generate the

approximations $\tilde{\mathbf{x}}$ of the original image vectors $\mathbf{x}$ from the DAE outputs $\tilde{\mathbf{z}}$. To quantify the reconstruction performance of our proposed approach, we considered two performance criteria in this work, i.e., (i) MSE between original and reconstructed images and (ii) classification accuracy (CA) for the reconstructed images. For the second criterion, we trained a *softmax* layer in a supervised manner by utilizing the latent vectors $\mathbf{z}$ and labels corresponding to the training data set as inputs and targets, respectively. The estimated latent vectors $\tilde{\mathbf{z}}$ pertaining to the testing data set are then classified using the pretrained softmax layer and the CAs are determined.

3. Results and Discussion

To investigate the effects of quantization and channel noise on data compression using AEs, we carried out extensive numerical simulations using MNIST database. For realizing various compression ratios, we considered four latent vector sizes, i.e., 10/20/30/40 and four different number of quantization bits, i.e., 2/3/4/8 to represent the latent variables. Furthermore, we assumed three distinct noise powers for the communication channel pertaining to BER = 0 (i.e., an ideal channel), BER = 4×10^{-3} and BER = 2×10^{-2}, where the last two BERs correspond to typical hard and soft decision forward error correction (FEC) limits, respectively, for communication systems. We compared systems with and without DAE and used MSE and CA as comparison metrics.

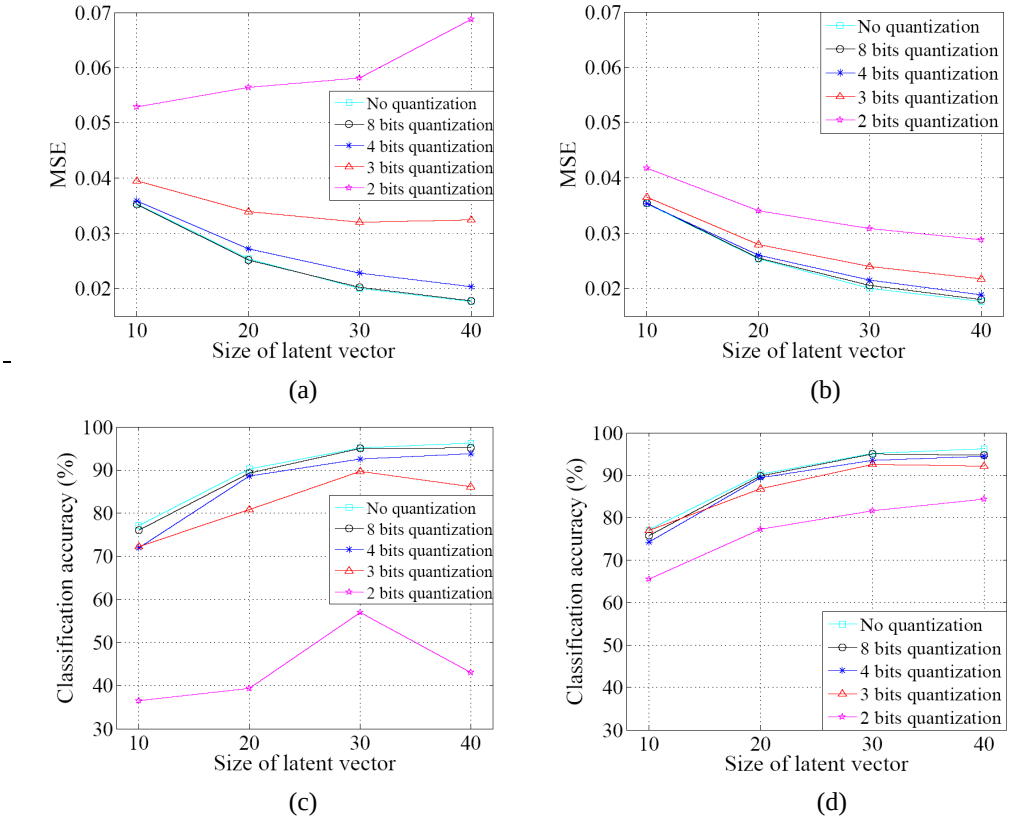


Fig. 3. MSEs for various latent vector sizes and number of quantization bits for systems (a) without denoising and (b) with denoising approaches. CAs for various latent vector sizes and number of quantization bits for systems (c) without denoising and (d) with denoising approaches. The channel considered is noiseless, i.e., BER = 0.

The simulation results (which are mean of 3 trials) for different BERs are presented in Figs. 3–5 for both autoencoding approaches discussed earlier. Fig. 3 shows MSE and CA for BER = 0. Since in this case the channel is completely noiseless, the received latent vectors $\tilde{\mathbf{z}}_q = \mathbf{z}_q$ are corrupted only by the noise introduced due to quantization of original latent variables. Fig. 3(a) and (c) depict results for the autoencoding approach involving no DAE while Fig. 3(b) and (d) show results for the approach where a pretrained DAE is first applied to denoise the received corrupted latent vectors $\tilde{\mathbf{z}}_q$ from quantization noise and the resulting estimates $\tilde{\mathbf{z}}$ of original latent vectors $\mathbf{z}$ are then used for image reconstruction and classification. It is clear from Fig. 3 that the use of DAE reduces MSE and improves CA significantly, thus showing the effectiveness of proposed approach in diminishing the adverse effects of quantization noise.

It is also evident from Fig. 3(b) that the MSE decreases with an increase in size of the latent vector. However, this decrease is not linear and tends to saturate for larger latent vector sizes. The CA results in Fig. 3(d) show similar trends. Furthermore, it can be observed from Fig. 3 that MSE increases while CA decreases only slightly as the number of quantization bits is decreased from 8 bits to 4 bits. On the other hand, both parameters change drastically as the number of quantization bits is varied from 4 bits to 2 bits. This clearly indicates that both MSE and CA scale nonlinearly with the number of quantization bits.

Figures 4 and 5 present results for BERs of 4×10^{-3} and 2×10^{-2}, respectively. Note that in contrast to BER = 0 case, the received latent vectors $\tilde{\mathbf{z}}_q$ are now corrupted by both quantization and channel noise. Once again, parts (a) and (c) of both figures show results for the approach using no DAE while parts (b) and (d) of both figures depict results for the proposed method employing a DAE first for obtaining reasonably good estimates $\tilde{\mathbf{z}}$ of the original transmitted latent vectors $\mathbf{z}$ before actual image reconstruction. A comparison with the BER = 0 case in Fig. 3 reveals that both MSE and CA deteriorate significantly with an increase in BER since the latent vectors get more corrupted. It is also evident from these plots that communication using fewer number of quantization bits, such as 2 or 3 bits, is more vulnerable to the channel noise as compared to the case where relatively larger number of quantization bits is utilized.

Moreover, parts (b) and (d) of Figs. 4 and 5 clearly illustrate that the use of DAE reduces MSE and improves CA substantially. This is attributed to the fact that the pretrained DAE is able to recover reasonably good estimates $\tilde{\mathbf{z}}$ of the original transmitted latent vectors $\mathbf{z}$ from $\tilde{\mathbf{z}}_q$. This in turn makes data recovery process more robust against channel noise and quantization effects. It can also be observed from Figs. 4 and 5 that the use of DAE is more advantageous for larger latent vector sizes. This is due to the fact that for smaller latent vector sizes, the effect of dimensionality reduction on reconstruction performance is more dominant. However, as the size of latent vector increases, the reconstruction error caused by dimensionality reduction decreases and the effect of noise on reconstruction performance prevails and this is where the benefits of using DAE become more evident.

Furthermore, if one considers total number of bits used for representing the whole latent vector, there exists an optimal combination of latent vector size and number of quantization bits per latent variable that minimizes MSE and maximizes CA. For example, if the number of transmission bits allowed for a given MNIST image is fixed to be 80 then this can be potentially realized by using a latent vector of size 10 with 8 quantization bits, a latent vector of size 20 with 4 quantization bits, and a latent vector of size 40 with 2 quantization bits. Comparing parts (b) and (d) of Figs. 3–5, it can be noted that a latent vector of size 20 with 4 quantization bits yields minimum MSE and maximum CA for all three BERs considered in this work. It is therefore an optimum choice for this particular compression scenario. We note that an in-depth study of the trade-offs between latent vector size and number of quantization bits per latent variable is interesting as well as important and this will be explored in the future.

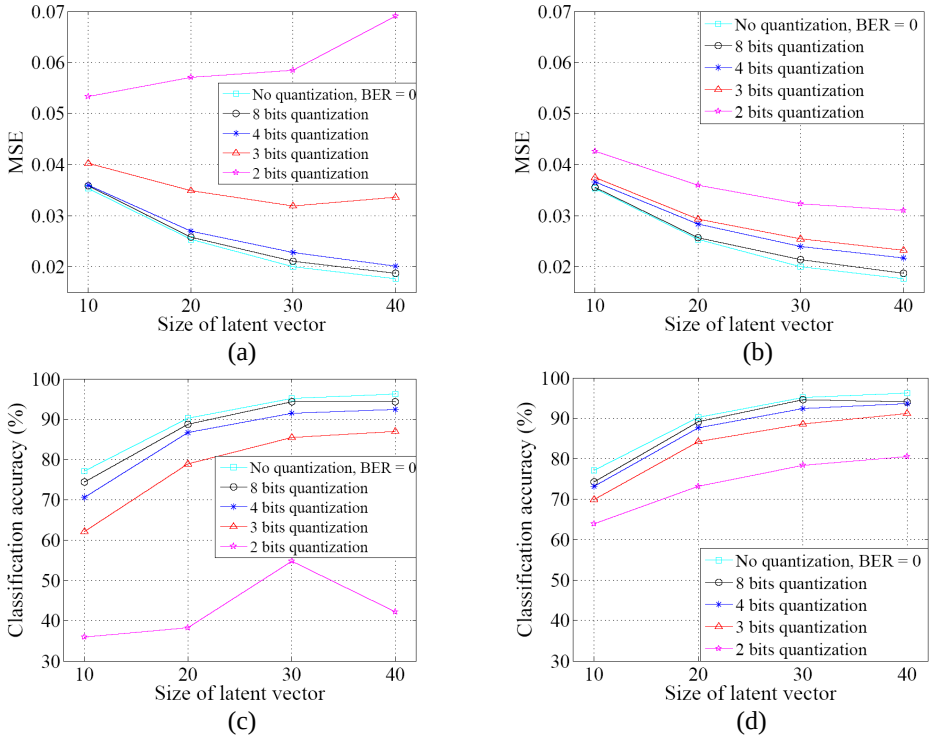


Fig. 4. MSEs for various latent vector sizes and number of quantization bits for systems (a) without denoising and (b) with denoising approaches. CAs for various latent vector sizes and number of quantization bits for systems (c) without denoising and (d) with denoising approaches. The BER is 4×10^{-3}.

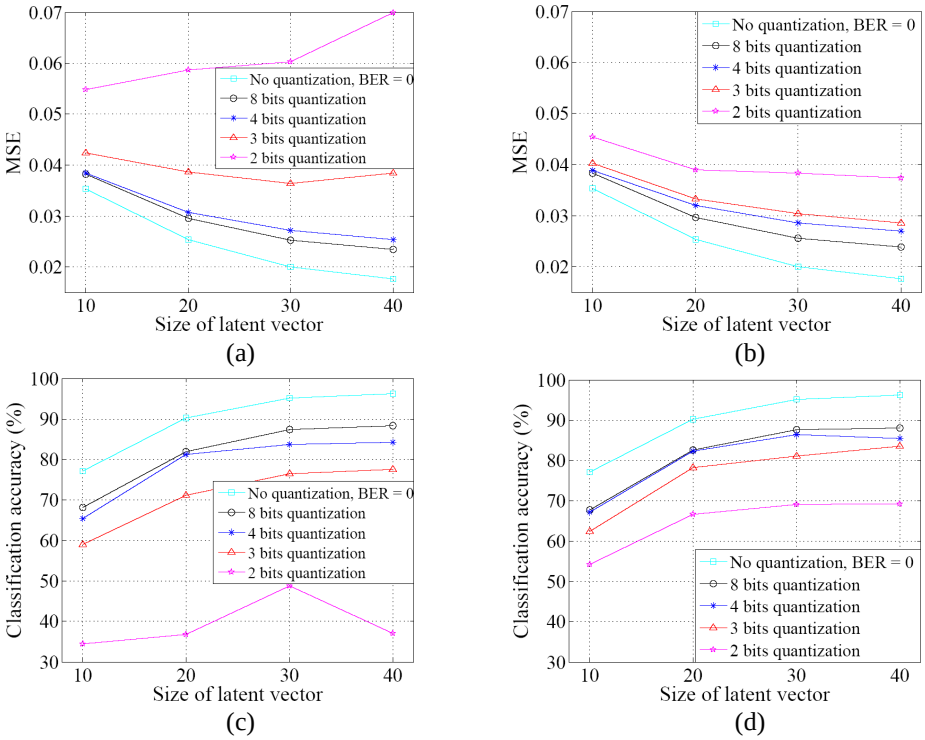
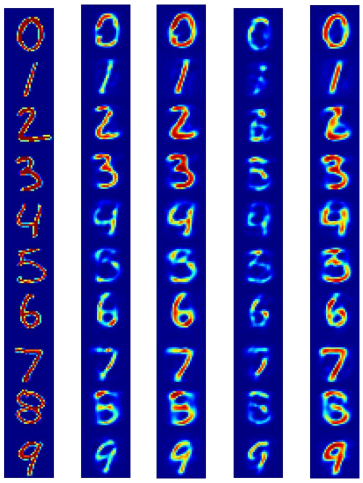


Fig. 5. MSEs for various latent vector sizes and number of quantization bits for systems (a) without denoising and (b) with denoising approaches. CAs for various latent vector sizes and number of quantization bits for systems (c) without denoising and (d) with denoising approaches. The BER is 2×10^{-2}.

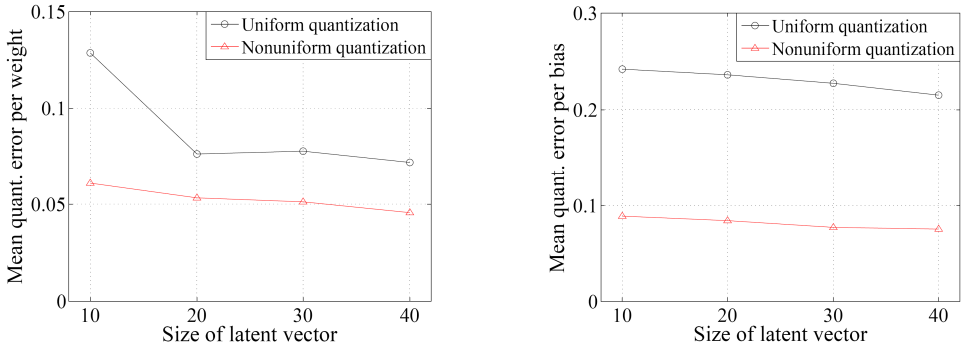
We also visually analyzed the effects of quantization and channel noise on the image reconstruction performance. Fig. 6 shows few original and reconstructed MNIST images for different latent vector sizes, number of quantization bits and BERs for systems without and with DAE. It is clear from the figure that without denoising the received corrupted latent vectors, the reconstructed images are severely distorted especially when fewer number of quantization bits is utilized. On the other hand, the use of DAE significantly improves the reconstruction quality, as expected, thus demonstrating the advantage of proposed autoencoding approach.



(a) (b) (c) (d) (e)

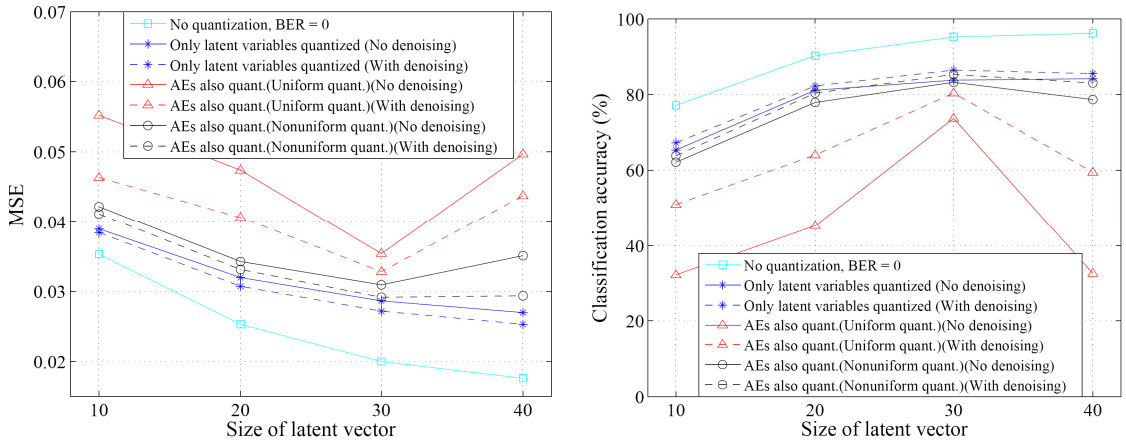
Fig. 6. (a) Original images ; reconstructed images (b) without denoising and (c) with denoising using latent vector size = 40, 3 bits quantization for latent variables and BER = 2×10^{-2}; reconstructed images (d) without denoising and (e) with denoising using latent vector size = 30, 2 bits quantization for latent variables and BER = 4×10^{-3}.

Finally, we investigated the effect of quantization of AEs' parameters on image recovery performance in addition to the latent variables quantization considered earlier. For this purpose, we analyzed uniform as well as nonuniform quantization (i.e., Lloyd-Max quantization) of AEs' weights and biases. Fig. 7 shows mean quantization errors per weight/bias using 4 bits for both quantization approaches. It is evident from the figure that for the same number of quantization bits used, nonuniform quantization introduces comparatively much less errors into the weights/biases. This is due to the fact that AEs' weights and biases are not uniformly distributed and hence nonuniform quantization would be an optimum choice in this case. Fig. 8 shows MSE and CA results when both latent variables and AEs' parameters are quantized using 4 bits. The results for the case where only latent variables are quantized are also shown as a reference. It is clear from the figure that both MSE and CA deteriorate significantly when AEs' parameters are uniformly quantized, as expected. On the other hand, nonuniform quantization of AEs' parameters reduces MSE and improves CA substantially as evident from the figure. In fact, the MSE and CA results for nonuniform quantization are quite close to the case where no quantization of AEs' parameters is performed at all. From these results, we can conclude that nonuniform quantization can substantially decrease the storage requirements of AEs' parameters while still offering quite similar image reconstruction performance, thus highlighting the critical advantage of this approach in practical resource-limited devices.



(a) (b)

Fig. 7. Mean quantization error per (a) weight and (b) bias using uniform and nonuniform quantization approaches. The number of quantization bits used for each weight/bias is 4.



(a) (b)

Fig. 8. Effect of quantization of latent variables and AEs' parameters on (a) MSE and (b) CA. The number of quantization bits used for each latent variable/AE parameter is 4 and the BER considered is 2×10^{-2}.

4. Conclusion

In this paper, we studied the effects of quantization and channel noise on systems in which latent representations of MNIST images, obtained using AEs, are quantized and transmitted over noisy communication channels. The results show that the corruption of latent vectors severely degrades image reconstruction and classification performance. We demonstrated that the use of an additional DAE at the receiver can mitigate deleterious quantization and channel noise effects on latent vectors and thus significantly improve image reconstruction quality and CA. We also explored various trade-offs between data recovery performance and the number of latent variables/quantization bits used. Much more remains to be investigated to better understand the most appropriate way to derive and transmit latent representations of images and other data over practical communication channels.

In the future, we will explore the use of more advanced AE architectures such as variational autoencoder (VAE) or convolutional autoencoder (CAE) in combination with DAE and compare their data recovery performance and computational complexity with the method proposed in this paper. Moreover, we will evaluate the generalization capabilities of these approaches by training them using images from one domain and later applying them to compress images from a different domain.

Acknowledgments

The authors would like to express their appreciation to David Ha at Google Brain for his technical advice and constructive comments. This work was supported by Hong Kong Government general research fund (GRF) under project number PolyU152757/16E as well as by National Natural Science Foundation China under project numbers 61435006 and 61401020.

References

[1] M. A. Alsheikh, S. Lin, D. Niyato, and H.-P. Tan, "Machine learning in wireless sensor networks: Algorithms, strategies, and applications," *IEEE Communication Surveys and Tutorials*, vol. 16, no. 4, pp. 1996–2018, 4th Quart. 2014.

[2] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1798–1828, Aug. 2013.

[3] Y. Bengio, "Learning deep architectures for AI," *Foundations and Trends in Machine Learning*, vol. 2, no. 1, pp. 1–127, Nov. 2009.

[4] Y. Lv, Y. Duan, W. Kang, Z. Li, and F.-Y. Wang, "Traffic flow prediction with big data: A deep learning approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 2, pp. 865–873, Apr. 2015.

[5] F. N. Khan, C. Lu, and A. P. T. Lau, "Joint modulation format/bit-rate classification and signal-to-noise ratio estimation in multipath fading channels using deep machine learning," *IET Electronics Letters*, vol. 52, no. 14, pp. 1272–1274, Jul. 2016.

[6] F. N. Khan, K. Zhong, W. H. Al-Arashi, C. Yu, C. Lu, and A. P. T. Lau, "Modulation format identification in coherent receivers using deep machine learning," *IEEE Photonics Technology Letters*, vol. 28, no. 17, pp. 1886–1889, Sep. 2016.

[7] M.-J. Kang, and J.-W. Kang, "Intrusion detection system using deep neural network for in-vehicle network security," *Plos One*, vol. 11, no. 6, Jun. 2016.

[8] Q. Feng, Y. Zhang, C. Li, Z. Dou, and J. Wang, "Anomaly detection of spectrum in wireless communication via deep auto-encoders," *Journal of Supercomputing*, pp. 1–18, Mar. 2017.

[9] X. Li, Y. Zhang, I. Marsic, A. Sarcevic, and R. S. Burd, "Deep learning for RFID-based activity recognition," in *Proc. 14th ACM Conference on Embedded Network Sensor Systems*, Stanford, USA, 2016, pp. 164–175.

[10] D. Zordan, B. Martinez, I. Villajosana, and M. Rossi, "On the performance of lossy compression schemes for energy constrained sensor networking," *ACM Transactions on Sensor Networks*, vol. 11, no. 1, pp. 15:1–15:34, Jul. 2014.

[11] B. A. Rajoub, "An efficient coding algorithm for the compression of ECG signals using the wavelet transform," *IEEE Transactions on Biomedical Engineering*, vol. 40, no. 4, pp. 355–362, Apr. 2002.

[12] G. Valenzise, A. Purica, V. Hulusic, and M. Cagnazzo, "Quality assessment of deep-learning-based image compression," in *Proc. IEEE 20th International Workshop on Multimedia Signal Processing*, Vancouver, Canada, 2018.

[13] O. Rippel, and L. Bourdev, "Real-time adaptive image compression," in *Proc. 34th International Conference on Machine Learning*, Sydney, Australia, 2017, pp. 2922–2930.

[14] G. E. Hinton, and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, Jul. 2006.

[15] D. D. Testa, and M. Rossi, "Lightweight lossy compression of biometric patterns via denoising autoencoders," *IEEE Signal Processing Letters*, vol. 22, no. 12, pp. 2304–2308, Dec. 2015.

[16] M. A. Alsheikh, P. K. Poh, S. Lin, H.-P. Tan, and D. Niyato, "Efficient data compression with error bound guarantee in wireless sensor networks," in *Proc. 17th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, Montreal, Canada, 2014, pp. 307–311.

[17] J. Wang, H. He, and D. V. Prokhorov, "A folded neural network autoencoder for dimensionality reduction," *Procedia Computer Science*, vol. 13, pp. 120–127, Oct. 2012.

[18] S. Rao, and S. Yadav, "A novel and efficient technique of communication in IOTs using autoencoders," *International Journal of Computer Science and Engineering*, vol. 4, no. 2, pp. 39–44,

Mar. 2015.
[19] Y.-D. Zhang, Y. Zhang, X.-X. Hou, H. Chen, and S.-H. Wang, “Seven-layer deep neural network based on sparse autoencoder for voxelwise detection of cerebral microbleed,” *Multimedia Tools and Applications*, vol. 77, no. 9, pp. 10521–10538, 2018.
[20] W. Jia, M. Yang, and S.-H. Wang, “Three-category classification of magnetic resonance hearing loss images based on deep autoencoder,” *Journal of Medical Systems*, vol. 41, no. 10, Paper 165, Oct. 2017.
[21] S. Marsland, *Machine Learning: An Algorithmic Perspective*, 2nd ed. Boca Raton, USA: CRC Press, 2015.
[22] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, “Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion,” *Journal of Machine Learning Research*, vol. 11, no. 11, pp. 3371–3408, Dec. 2010.