MITSUBISHI ELECTRIC RESEARCH LABORATORIES
http://www.merl.com

Nonlinear Equalization with Deep Learning for Multi-Purpose Visual MIMO Communications

Fujihashi, T.; Koike-Akino, T.; Watanabe, T.; Orlik, P.V.

TR2018-039 June 2018

Abstract

A major challenge of screen-camera visual multiinput multi-output (MIMO) communications is to increase the achievable throughput by reducing nonlinear channel effects including perspective distortion, ambient lights, and color mixing. To mitigate such nonlinear effects, an existing transmission method uses linear or simple nonlinear equalizations in decoding operations. However, the throughput improvement from the equalization techniques is often limited because the effects are composed of a combination of various nonlinear distortions. In addition to the above issue, the existing studies consider specific environments, such as indoor and static communications, although screen-camera communications can be used for a variety of applications including outdoor and mobile scenarios. In this study, we propose 1) deep neural network (DNN)- based decoding for screen-camera communications to increase the achievable throughput and 2) Unity 3D-based evaluation methodology to synthetically learn the DNN for being robust against many different screen-camera environments. The DNN finds the best nonlinear kernels for equalization from numerously captured images, and then decodes original bits from newly captured images based on the trained nonlinear kernels. In the Unity-based evaluation tool, we can easily capture numerous photo-realistic images in different screen-camera scenarios to learn the impact of perspective distortion, screen-to-camera distance, motion blur, and ambient lights on the throughput since Unity-based environment can freely set programmable screens, cameras, and ambient lights on a 3D space. As an initial proof of concept, we demonstrate that the proposed DNN-based decoder scheme improves the achievable throughput by up to 148% compared to existing methods by equalizing nonlinear effects.

IEEE International Conference on Communications (ICC)

Nonlinear Equalization with Deep Learning for Multi-Purpose Visual MIMO Communications

Takuya Fujihashi†, Toshiaki Koike-Akino‡, Takashi Watanabe*, Philip V. Orlik‡
† Graduate School of Science and Engineering, Ehime University, Matsuyama, Ehime
‡ Mitsubishi Electric Research Laboratories (MERL), Cambridge, MA 02139, USA
* Graduate School of Information Science and Technology, Osaka University, Suita, Osaka
Email: fujihashi@cs.ehime-u.ac.jp, koike@merl.com, watanabe@ist.osaka-u.ac.jp, porlik@merl.com

***Abstract*—A major challenge of screen-camera visual multi-input multi-output (MIMO) communications is to increase the achievable throughput by reducing nonlinear channel effects including perspective distortion, ambient lights, and color mixing. To mitigate such nonlinear effects, an existing transmission method uses linear or simple nonlinear equalizations in decoding operations. However, the throughput improvement from the equalization techniques is often limited because the effects are composed of a combination of various nonlinear distortions. In addition to the above issue, the existing studies consider specific environments, such as indoor and static communications, although screen-camera communications can be used for a variety of applications including outdoor and mobile scenarios. In this study, we propose 1) deep neural network (DNN)-based decoding for screen-camera communications to increase the achievable throughput and 2) Unity 3D-based evaluation methodology to synthetically learn the DNN for being robust against many different screen-camera environments. The DNN finds the best nonlinear kernels for equalization from numerously captured images, and then decodes original bits from newly captured images based on the trained nonlinear kernels. In the Unity-based evaluation tool, we can easily capture numerous photo-realistic images in different screen-camera scenarios to learn the impact of perspective distortion, screen-to-camera distance, motion blur, and ambient lights on the throughput since Unity-based environment can freely set programmable screens, cameras, and ambient lights on a 3D space. As an initial proof of concept, we demonstrate that the proposed DNN-based decoder scheme improves the achievable throughput by up to 148% compared to existing methods by equalizing nonlinear effects.**

I. Introduction

Visible light communications (VLC) [1], [2] have emerged as promising complementary technologies to conventional radio-frequency (RF) wireless communications. Screen-camera communications [3]–[5] are such VLC technologies, where digital data can be transmitted via image signals from a screen to a camera. For screen-camera communications, digital bits are encoded in the screen image on devices, e.g., laptop computers and smart phones. A receiver equipped with camera image sensors captures the screen to decode the information. Screen-camera communications can be used for various wireless applications, such as inter/intra vehicle communications [6], near field communications [7], [8], and augmented reality (AR) [9]. The use of screen and camera can form so-called multi-input multi-output (MIMO) systems in which optical transmissions by an array of light-emitting devices are received by an array of photo-detector elements. Although typical frame rates of screen and camera devices are relatively low in general (e.g., 50 frames per second), high-definition screen and camera can realize a massive spatial-multiplexing gain to transfer a large amount of information bits at once. In addition, visual MIMO communications often do not need to deploy dedicated equipment since recent devices including smart phones are already configured with a high-end display and camera sensor.

A major challenge of the screen-camera communications is to increase the transmission rate and communication distance in nonlinear channels in the presence of ambient noise. In particular, there are three issues in such links as follows. First, an encoded image on the screen is distorted due to receiver's perspective, depending on the angle of captures. When the receiver captures the encoded image on a rectangular screen from a certain angle, the captured image will become trapezoid-shaped. This phenomenon is referred to as perspective distortion [10]. Second, the luminance of encoded image is severely impaired by ambient lights such as sunlight. This impairment causes errors in the encoded information, resulting in a low transmission rate. Third, the spectrum sensitivity of red, green, and blue color channels on the camera sensor is non-orthogonal and highly nonlinear. Specifically, the output of one color channel may be degraded by the intensity of the other color channels. This is known as color mixing.

To overcome the above-mentioned issues, some approaches [11]–[15] have been proposed for screen-camera links to improve the transmission rates. For example, [16] uses orthogonal discrete multi-tone (DMT) and nonbinary coding for high-speed transmission, along with several reconstruction algorithms to reduce an effect of perspective distortion and ambient noise. In addition, an equalization based on nonlinear Volterra series is experimentally investigated to mitigate color mixing distortion.

However, there are two remaining issues in existing studies on screen-camera communications. The first issue is to realize higher transmission rate by accounting for residual nonlinear distortion in visual MIMO channels. Since the captured images are distorted by multiple nonlinear effects, a simple nonlinear model used in [16] may not lead to a significant improvement in throughput. To increase the transmission rate by decreasing such nonlinear effects, a new decoding technique to cope with

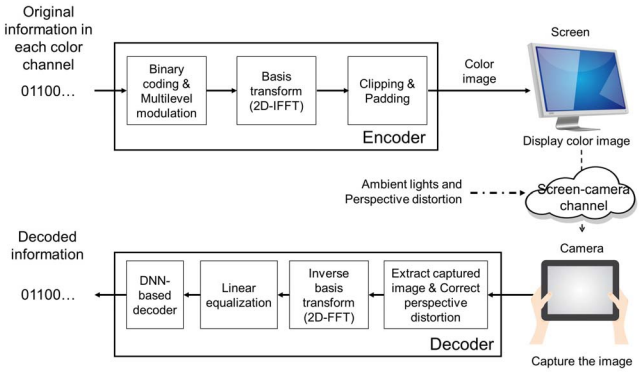


Fig. 1. Encoding and decoding operations in screen-camera visual MIMO communications with the proposed DNN-based decoder.

multiple nonlinear effects is required.

The second issue is a limited amount of available evaluation data under a specific communication environment. Although the most of studies on screen-camera communications rely on experimental measurements, the conducted experimental settings are often too limited, e.g, indoor communication and fixed position. On the other hand, screen-camera communications can be used for outdoor and mobile applications, including intelligent transportation system (ITS) and AR. To demonstrate the feasibility of screen-camera communications in various applications, a new evaluation methodology of screen-camera communications is needed.

In this paper, we propose a new decoding operation and analyzing framework to discuss the performance of screen-camera communications in many scenarios. Firstly, we propose a deep neural network [17], [18] (DNN)-based decoder for high-speed visual MIMO communications. The DNN is a multi-layer perceptron (MLP) with many hidden layers to learn nonlinear feature underlying the problem. To find the best weights of the DNN, the decoder first learns the weights from numerously captured images and the corresponding original bits as the inputs and target outputs of the DNN. Based on the learned weights, the decoder reconstructs original bits from the newly captured images. Although an effect of Volterra-based nonlinear channel equalization was discussed in [16], there was no study addressing an impact of the DNN-based approach on throughput improvement in screen-camera visual MIMO communications, to the best of authors' knowledge.

Secondly, we propose a Unity 3D-based evaluation framework of screen-camera communications. In Unity 3D [19], we can freely set the screen, cameras, and ambient lights on the 3D space. In addition, the parameters of these components, e.g., position, motion, resolution, and light intensity, can be dynamically controlled. Consequently, we can easily generate a massively large number of evaluation data to simulate the throughput of applications, that are difficult in experimental evaluations. Moreover, those synthetically captured data can be used for deep learning to make the DNN be more robust against different environments. In this paper, to demonstrate the proposed methodology, transmitted images displayed on the screen are captured by multiple cameras, which are located on different positions in the Unity 3D scene, and then the captured images are used for parameter learning in the proposed DNN-based decoder to achieve better decoding performance.

Evaluation results show that the transmission rate of the proposed scheme can be improved by up to 148.4 % compared to that of existing studies at the same communication distance. The proposed framework can be more effective when we use a variety of different conditions for deep learning to resolve the impact of more serious nonlinear impairments due to motion blur, color mixing, and ambient light.

II. Screen-Camera Visual MIMO Communication

A. System Overview

The purpose of our study is to realize higher transmission rate in screen-camera communications by mitigating nonlinear effects in the screen-camera channel. Fig. 1 shows the schematic of our proposed scheme. We use a pair of screen and camera as the sender and receiver, respectively. Note that there are three major differences from RF wireless communications. First, input values for the screen, i.e., pixel luminance values, should not be complex-valued numbers. Second, the input values are two dimensional (2D) in spatial domain. Third, the pixel luminance values typically range over finite non-negative integers, i.e., $0, 1, \ldots, 255$ for 8-bit quantization.

Based on the constraints, the sender first encodes original information with binary coding, followed by 2^M-ary QAM modulation format and arranges the modulated symbols into a 2D image matrix. The modulated coefficients are then transformed to pixel luminance values by taking inverse 2D fast Fourier transform (FFT) operation, and clipped according to the luminance range. Finally, several white pixels of padding is added to the two dimensional values prior to display.

At the receiver side, pixel luminance values are captured by camera sensors and then the transmitted region is extracted from the captured values using an edge detection algorithm. The captured luminance values are filtered, transformed into frequency domain, and equalized using homography transform, 2D-FFT, and linear equalization, respectively. Each equalized coefficient is fed into the proposed pixel-wise DNN-based decoder to resolve residual nonlinear distortion in prior to decode the coefficient into original bits.

B. Sender Operations

1) Basis Transform: In order to be robust against inter-pixel interference, we use a basis transform technique based on 2D-FFT. The stream of QAM-modulated symbols are transformed to pixel luminance values using inverse 2D-FFT for each color channel. As mentioned above, the screen does not accept complex-valued luminance. To ensure transformed values are purely real, we arrange the 2D matrix to be Hermitian symmetry. Note that the output from the inverse FFT will be entirely real when the input values are Hermitian.

More specifically, we suppose the use of screen image having a resolution of $H \times W$ pixels, for transmitting HW

real values in total per color channel. In each color channel, modulated QAM symbols are arranged into a matrix of size $H \times (W/2)$ and the 1D inverse FFT is carried out for each column. The FFT coefficients are organized to be Hermitian symmetry by assigning the complex conjugate of the value at the (i, j)th frequency coefficient to the $(i, -j)$th frequency coefficient. The coefficients in each row are then fed into the inverse FFT. The resulting HW values are all real and can be sent as screen image.

2) Clipping: We consider 8 bits for quantization representation of pixel luminance for screen images. To ensure the output of FFT being within the range between 0 and 255, the pixel luminance values are shifted and scaled to have a mean of $255/2$ and a variance of $(255/2c_{\text{tail}})^2$, where c_{tail} is a clipping parameter. All values outside the range between 0 and 255 are clipped to 0 and 255, respectively. When the FFT size is large enough, the luminance values may follow a Gaussian distribution according to the central limit theorem. By adjusting the clipping parameter c_{tail}, we can control the probability of clipping events.

3) Padding: Since light emitted from an LCD screen is diffusive in nature, each photo detector of camera sensors receives multiple lights from nearby LCD pixels. As a result, the LCD pixels are blended into one camera pixel in particular for long distance and mobile devices due to blur. The FFT-based transmission is insensitive to linear inter-pixel interference induced by the blur. However, this blending effect can still cause performance degradation at edges of the screen images, i.e., background values outside the encoded pixels can interfere. To reduce the edge effect, pixels with white color are appended around the encoded values as padding. Finally, the encoded values with padding are displayed on the screen with black background. Note that the white padding is also useful for edge detection at the receiver.

C. Receiver Operations

1) Pixel Extraction and Perspective Correction: Receiver's camera first captures an image, which contains the transmitter's screen, for communications. Prior to decoding, the area of encoded pixels is extracted from the captured image. This requires the receiver to detect the four corners of the area. In our implementation, the encoded values can be extracted by detecting edges between white padding and black background.

However, the extracted image is typically trapezoid and its luminance is shifted due to perspective distortion and ambient light distortion. We correct the perspective distortion by using homography operation [20]. Specifically, a trapezoid image can be transformed to a rectangle image based on four corners of images. After the homography operation, the luminance values are fed into 2D-FFT to obtain frequency-domain coefficients.

2) Equalization: In screen-camera communications, transmitted symbols are impaired by color mixing and an effective noise in frequency domain. Let $\mathbf{y}_{i,j}$ denote a 3×1 vector of received symbols at (i, j)th frequency component. Each row represents the received symbol from red, green, and blue color channels, respectively. In [16], the received symbols in screen-camera links are modeled as nonlinear systems:

$$\mathbf{y}_{i,j} = \mathbf{H}_{i,j}\phi(\mathbf{x}_{i,j}) + \mathbf{z}_{i,j}, \tag{1}$$

where $\mathbf{x}_{i,j}$ is a 3×1 vector of transmitted symbols in frequency domain, $\mathbf{H}_{i,j}$ is a $3 \times K$ channel gain matrix, $\mathbf{z}_{i,j}$ is a 3×1 additive white Gaussian noise (AWGN) vector with a noise variance of $\sigma^2_{i,j}$. Here, $\phi(\cdot)$ denotes a nonlinear kernel expansion and K is an expansion cardinality. For example, the first-order Volterra series expansion including an offset term [21], [22] is expressed as $\phi(\mathbf{x}) = [1, \mathbf{x}^{\mathrm{T}}]^{\mathrm{T}}$ with $K = 1 + 3 = 4$. Here, $[\cdot]^{\mathrm{T}}$ is a transpose.

We employ minimum mean-square error (MMSE) equalization for the Volterra series expansion of the received symbols, i.e., $\phi(\mathbf{y}_{i,j})$. Specifically, MMSE filter weights of size $3 \times K$ are obtained as follows:

$$\mathbf{G}_{i,j} = \mathbb{E}\big[\mathbf{x}_{i,j}\phi(\mathbf{y}_{i,j})^{\dagger}\big]\mathbb{E}\big[\phi(\mathbf{y}_{i,j})\phi(\mathbf{y}_{i,j})^{\dagger}\big]^{-1}, \tag{2}$$

where $\mathbb{E}[\cdot]$ and $[\cdot]^{\dagger}$ denote the expectation and Hermitian transpose, respectively. In practice, the expectation is taken place by averaging multiple measurements in the past. Note that the first-order Volterra series expansion for $\phi(\cdot)$ will reduce to a linear equalizer. Finally, the received symbols are equalized using the MMSE filter as follows:

$$\hat{\mathbf{x}}_{i,j} = \mathbf{G}_{i,j}\phi(\mathbf{y}_{i,j}). \tag{3}$$

3) DNN-based Decoding: Fig. 2 shows the proposed pixel-wise DNN-based decoder. Each DNN-based decoder is composed of an input layer, multiple hidden layers, and an output layer to decode original bits from the extracted images. To reconstruct original bits from the received coefficients, the proposed decoder solves a multi-label classification problem, which is a combination of multiple binary classification problems. More specifically, each unit in the input layer represents each received frequency coefficient of imaginary and quadratic components in each color channel, and thus the number of units in the input layer is $2{\cdot}3 = 6$. In each hidden layer, we use the adaptive-moment stochastic back-propagation algorithm, 5% dropout, and the rectified linear unit (ReLU) activation

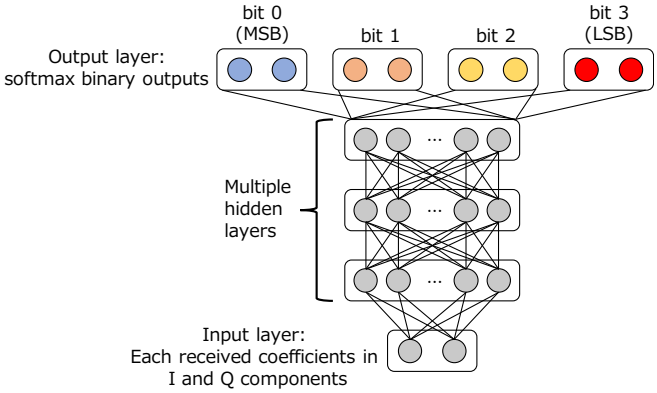


Fig. 2. Pixel-wise DNN-based decoder with multi-label classification.

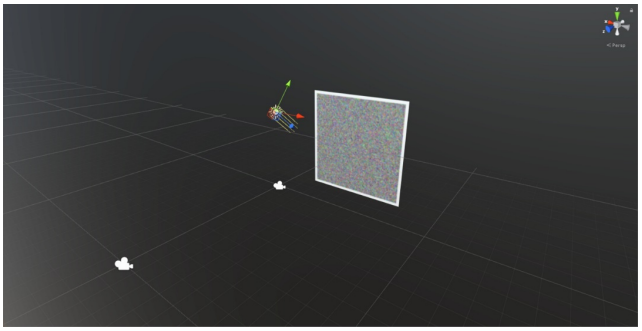
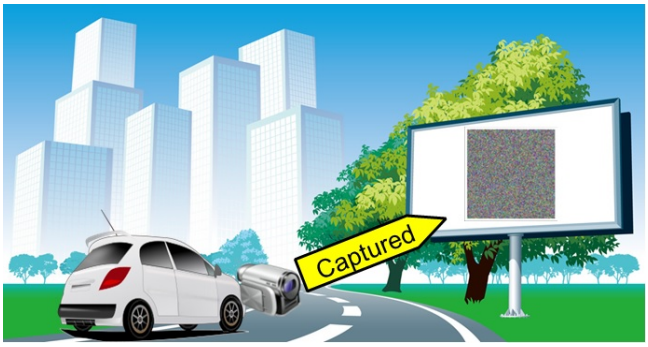
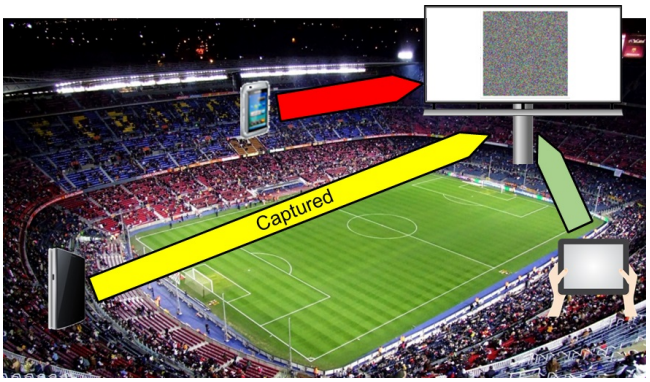


Fig. 3. Unity 3D platform for screen-camera communications.



(a) ITS systems (modified image of freedesignfile.com CC BY 2.0)



(b) Broadcasting in crowded environment

Fig. 4. Use-cases in screen-camera communications.

to find nonlinear relationship between the inputs and outputs. Finally, the output layer with $2 \cdot 3M$ units can classify M bits in each color channel from the hidden layer outputs. Each two-node tuple in the output layer calculates the softmax probability to represent the likelihood of one bit.

To learn the best weights for multi-label classification during multiple iterations, the proposed decoder first calculates the softmax cross-entropy loss for each binary classification, and then sums up the cross entropy loss over $3M$ bits. Based on the cross entropy across $3M$ bits, the proposed decoder updates the weights of the hidden layers to minimize the cross entropy for the subsequent iterations. The DNN has recently shown a great success in the media signal processing community including natural language, computer visions, etc. In order to achieve a significant gain, we usually require a massively huge amount of training data, which are usually not available in screen-camera communications experiments. To overcome this issue, we next introduce a new framework, which may facilitate multi-purpose VLC applications.

III. Unity 3D-based Screen-Camera Evaluation

A. Photo-Realistic Data Acquisition Framework

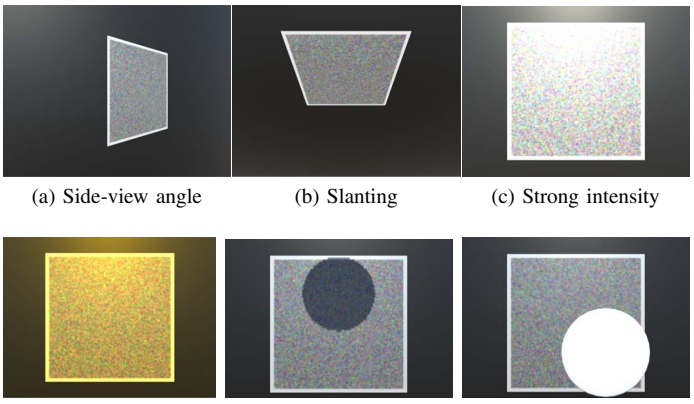
In our study, we create an image acquisition environment of screen-camera communications on Unity 3D. In Unity 3D, components of screen-camera communications, e.g., screens, cameras, and ambient lights, can be freely set on 3D space as shown in Fig. 3. The parameters of these components such as position, screen size, camera's resolution, angle, and ambient light intensity, can be dynamically controlled, depending on a use case that wants to evaluate the throughput performance.

Although this paper still focuses on a particular environment to evaluate the screen-camera communications, we can simulate the achievable throughput of many use cases in principle by reproducing the environment on the 3D space in Unity 3D. Fig. 4 shows examples of the use-cases in screen-camera communications. In the ITS scenarios, screen-camera communications can be used for vehicle-to-roadside and vehicle-to-vehicle communications to receive information such as safety alerts and digital advertisement around the area. For these cases, we should demonstrate effects of vehicle's speed, the size of each displayed image, the captured hour and weather on the achievable throughput.

In view of entertainment, we can use the visual MIMO communications for crowded environments, e.g., sport events, to broadcast digital information for many people. For example, a large screen displays visual information about professional players, and when the people capture the image on their smart phones, the image tells the player information in more details. In this case, effects of ambient lights and communication distance should be evaluated to broadcast much information to many people.

B. Deep Learning with Large-Scale Synthetic Data

The use of Unity 3D can generate a massively large number of data sets with photo-realistic images across a variety of use-case scenarios. Compared to experimental measurements, this methodology may produce much more data by several magnitudes of order. Those big data are specifically important for deep learning to make the proposed DNN decoder more robust against different nonlinear distortion regardless of any specific use-case applications. For example, because the performance of visual MIMO communications highly depends on the ambient light, the DNN shall be trained over different scenes having different sunlight color, direction, reflection, shadows, intensity, etc. Those different conditions can be easily treated by using the Unity 3D tool as shown in Fig. 5.



(a) Side-view angle (b) Slanting (c) Strong intensity

(d) Orange sunlight (e) Shadows (f) Occlusion

Fig. 5. Photo-realistic images in different scenes.

IV. Performance Evaluation

A. Analysis Setting

Image Acquisition Setting: For image acquisition in screen-camera communications, we create an environment on Unity 3D. We set one screen on the 3D space and two cameras at the front of the screen. The distance between the screen and each camera is 10 m to 30 m, respectively. The resolution of each camera is 352×288 pixels. In addition, one directional light is set on the above of the screen.

Metric: We evaluate the achievable throughput of the screen-camera communications system in terms of bits per image. The throughput is defined as follows.

$$R = B \cdot \mathcal{I}(X;Y), \tag{4}$$

where R denotes throughput (bits/image), B is the total number of transmitted bits in one encoded image, and $\mathcal{I}(X;Y)$ is mutual information between a transmitted image X and received image Y. Here, the mutual information for binary coding is calculated directly from the DNN outputs as follows:

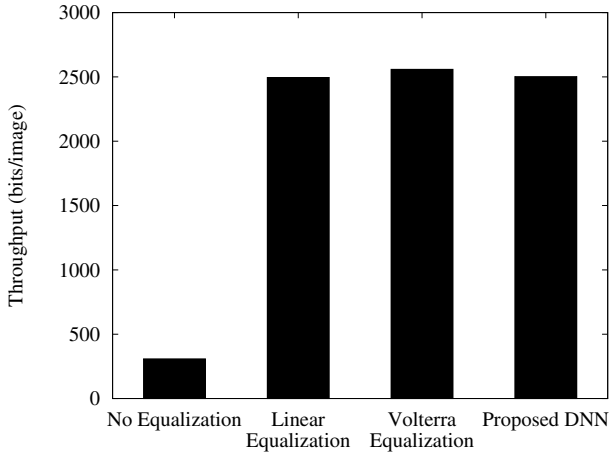
$$\mathcal{I}(X;Y) = 1 - \mathbb{E}\Big[\log_2\left(1+\exp(-L)\right)\Big|b=0\Big], \tag{5}$$

where L is the log-likelihood ratio (LLR) which is the difference of two DNN output binary nodes in softmax classifiers.

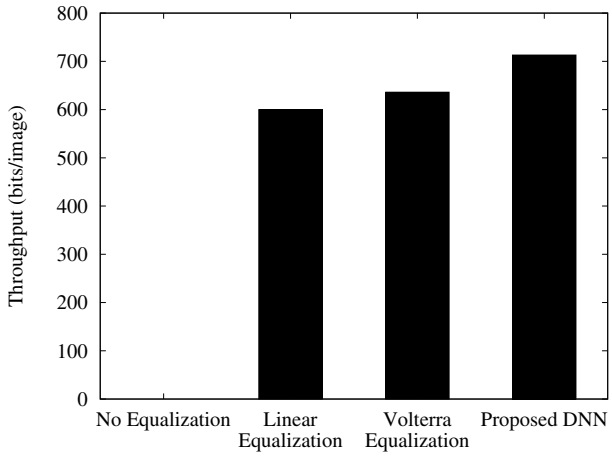
In our evaluation, 8000 images at a resolution of 100×100 pixels with 16QAM format are captured. Here, 7910 images are used for training and 90 images for test data.

DNN Parameters: We implemented the proposed pixel-wise DNN-based decoder using Chainer [23]. We consider two hidden layers, each of which is composed of 600 units. Detail analysis of the optimal number of units will be left as future work. To find the best weights of the hidden layers, we update the weights based on the total cross-entropy loss across $2 \cdot 3M$ units over 20-epoch iterations.

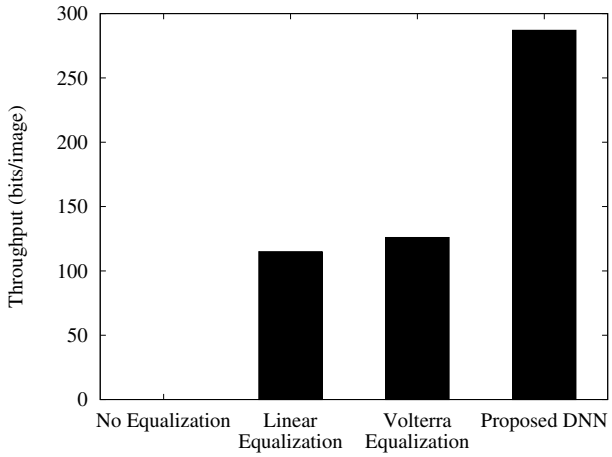
Reference Schemes: We consider four reference schemes under different decoding operations for comparison: without equalization, linear equalization, nonlinear equalization with the 2nd order Volterra series, and proposed DNN-based decoding.



(a) Communication distance of 10 m



(b) Communication distance of 20 m



(c) Communication distance of 30 m

Fig. 6. Achievable throughput at different communication distances.

B. Effect of Communications Distance

In this section, we evaluate an effect of communication distances between the screen and camera on throughput. Figs. 6(a), (b), and (c) show achievable throughput in screen-camera links at a communication distance of 10, 20, and

30 m, respectively. From the results, the proposed DNN-based decoding can enhance the achievable throughput compared to the conventional linear and nonlinear equalizations, in particular at longer distances. For example, the improvement by the proposed DNN-based decoder from the linear and nonlinear equalizations is approximately 15.9 % and 12.1 %, respectively, at a distance of 20 m. On the other hand, the achievable throughput will be nearly zero if we do not employ any equalization technique. It suggests that the equalization techniques play an important role in increasing throughput.

At a short-range communication, the improvement of the throughput in the proposed DNN-based decoder is marginal. This may be simply because there are less-dominant nonlinear distortion at such short-distance communications, and the achieved throughput by linear equalization is already high enough. At a long distance range of 30 m communications, we found that the improvement of throughput by the proposed DNN-based decoder is much more significant than 20 m screen-camera communications. Specifically, the proposed DNN-based decoder improves the throughput by 148.4 % compared to the linear equalization and 127.1 % compared to the nonlinear equalization. Considering these results, the proposed DNN-based decoder may be more advantageous in even more highly distorted environments.

V. Conclusion

This paper proposed a DNN-based decoder for nonlinear visual MIMO channels to improve achievable throughput. The DNN can find nonlinear relationship underlying the captured images to recover the original bits. To improve the decoding performance of the proposed DNN-based decoder by using many captured images, we created an image acquisition framework on Unity 3D and obtained thousands of the captured images for evaluations. From the evaluations, we demonstrated that the proposed DNN-based decoder achieves 2-times higher throughput than conventional schemes at long-range communications. The proposed framework will facilitate multi-purpose visual MIMO communications by synthetically creating million data sets for even deeper learning to realize environment-insensitive decoder.

Acknowledgment

T. Fujihashi's work was partly supported by JSPS KAKENHI Grant Number 17K12672.

References

[1] H. Burchardt, N. Serafimovski, D. Tsonev, S. Videv, and H. Haas, "VLC: Beyond point-to-point communication," *IEEE Communications Magazine*, vol. 52, no. 7, pp. 98–105, 2014.

[2] E. Curry, D. Borah, and J. M. Hinojo, "Optimal symbol set design for generalized spatial modulations in MIMO VLC systems," in *IEEE GLOBECOM*, 2016, pp. 1–7.

[3] A. Wang, Z. Li, C. Peng, G. Fang, and B. Zeng, "InFrame++: Achieve simultaneous screen-human viewing and hidden screen-camera communication," in *ACM MobiSys*, 2015, pp. 181–195.

[4] V. Nguyen, Y. Tang, A. Ashok, M. Gruteser, K. Dana, W. Hu, and E. Wengrowski, "High-rate flicker-free screen-camera communication with spatially adaptive embedding," in *IEEE INFOCOM*, 2016, pp. 1–9.

[5] M. Izz, Z. Li, H. Liu, Y. Chen, and F. Li, "Uber-in-light: Unobtrusive visible light communication leveraging complementary color channel," in *IEEE INFOCOM*, 2016, pp. 1–9.

[6] T. Yamazato, I. Takai, H. Okasa, T. Fujii, T. Yendo, S. Arai, M. Andoh, T. Harada, K. Yasutomi, K. Kagawa, and S. Kawahito, "Image-sensor-based visible light communication for automotive applications," *IEEE Communication Magazine*, vol. 52, no. 7, pp. 88–97, 2014.

[7] A. Wang, S. Ma, C. Hu, J. Huai, C. Peng, and G. Shen, "Enhancing reliability to boost the throughput over screen-camera links," in *ACM Annual International Conference on Mobile Computing and Networking*, 2014, pp. 41–52.

[8] W. Hu, H. Gu, and Q. Pu, "Lightsync: Unsynchronized visual communication over screen-camera links," in *ACM Annual International Conference on Mobile Computing and Networking*, 2013, pp. 15–26.

[9] T. W. Kan, C. H. Teng, and W. S. Chou, "Applying QR code in augmented reality applications," in *ACM International Conference on Virtual Reality Continuum and its Applications in Industry*, 2009, pp. 253–257.

[10] A. Ashok, S. Jain, M. Gruteser, N. Mandayam, W. Yuan, and K. Dana, "Capacity of pervasive camera based communication under perspective distortions," in *IEEE International Conference on Pervasive Computing and Communications*, 2014, pp. 114–120.

[11] S. Perli, N. Ahmad, and D. Katabi, "Pixnet: Interference-free wireless links using LCD-camera pairs," in *ACM Annual International Conference on Mobile Computing and Networking*, 2010, pp. 137–148.

[12] W. Huang, C. Gong, P. Tian, and Z. Xu, "Experimental demonstration of high-order modulation for optical camera communication," in *IEEE Symposium on Signal Processing for Optical Wireless Communications*, 2015, pp. 1027–1031.

[13] T. Hao, R. Zhou, and G. Xing, "COBRA: Color barcode streaming for smartphone systems," in *ACM International Conference on Mobile Systems, Applications, and Services*, 2012, pp. 85–98.

[14] W. Hu, J. Mao, Z. Huang, Y. Xue, J. She, K. Bian, and G. Shen, "Strata: Layered coding for scalable visual communication," in *ACM Annual International Conference on Mobile Computing and Networking*, 2014, pp. 79–90.

[15] W. Du, J. C. Liando, and M. Li, "SoftLight: Adaptive visible light communication over screen-camera links," in *IEEE Annual conference on Computer Communications*, 2016, pp. 1–9.

[16] T. Fujihashi, T. Koike-Akino, P. Orlik, and T. Watanabe, "Experimental throughput analysis in screen-camera visual MIMO communications," in *IEEE GLOBECOM*, 2016, pp. 1–6.

[17] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Networks*, vol. 61, pp. 85–117, 2015.

[18] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, http://www.deeplearningbook.org.

[19] *Unity 3D Game Engine*, 2017. [Online]. Available: https://unity3d.com/

[20] E. Vincent and R. Laganiere, "Detecting planar homographies in an image pair," in *International Symposium on Image and Signal Processing and Analysis*, 2001, pp. 1–6.

[21] M. Schetzen, *The Volterra and Wiener Theories of Nonlinear Systems*. Kriger, 1989.

[22] N. Mallouki, B. Nsiri, S. Mhatli, M. Ghanbarisabagah, W. Hakimi, M. Ammar, and E. Giacoumidis, "Analysis of full Volterra nonlinear equalizer for downlink LTE system," in *Wireless Telecommunications Symposium*, 2015, pp. 1–6.

[23] S. Tokui, *Chainer: A powerful, flexible and intuitive framework of neural networks*. [Online]. Available: https://chainer.org/