1. Introduction

Holographic data storage (HDS) is a promising optical recorder with a large storage capacity and high data-transfer rates. Some studies on HDS report its recording density and data-transfer rate as 2.4 Tbit/in² and 1 Gbps, respectively. Our previous studies showed the usefulness and practicality of HDS by demonstrating real-time playback of 8K movies from HDS. The abovementioned studies validated the large capacity and high data-transfer rate of HDS, despite the use of a binary modulation code in its recording and reproduction.

As shown in wired/wireless communication technologies, HDS can also improve the capacity and processing speed by using a multi-level modulation code. Recent studies have introduced multi-level recording methods with amplitude, phase, and their combination to record data in HDS. However, compared with the conventional binary modulation code, the multi-level modulation code handles larger quantities of minute signals and suffers from noise in the optical path. Therefore, the resulting low robustness must be taken into consideration. Hence, it is difficult to directly improve the signal-to-noise ratio (SNR) of the reproduced data, as the system becomes more complex with the introduction of the multi-level modulation code and optical elements to handle both the phase and amplitude.

As a demodulation method, we previously reported an image recognition method using convolutional neural network (CNN) for HDS. The retrieved data from the HDS are accurately demodulated by the CNN, based on the trained network. Although the CNN demodulation method is effective for multi-level amplitude modulation and binary modulation codes, however, bit errors after the demodulation must be completely removed. Thus, the error correction code is important. Various error correction codes are introduced in the HDS, such as the Reed-Solomon, turbo code, and low-density parity-check (LDPC) code. Among them, the spatially coupled LDPC (SC-LDPC) code is one of the strongest error correction codes that approaches the Shannon limit, based on the LDPC code. We confirmed that the capability of error correction of the SC-LDPC code outperforms that of the LDPC code in the HDS.

This study presents an effective data-decoding method by combining the CNN demodulation and SC-LDPC code to enable a more powerful error correction by using the likelihood information obtained as the output from the CNN. We evaluated the characteristics of the demodulation and error correction method using the reproduced data with numerically added noise.
2. Principle of Recording and Reproduction

We developed the 10:9 modulation code\(^{18}\) to record data with multi-level amplitude modulation, as presented in Fig. 1. In the 10:9 modulation code, each 10-bit data is transformed into nine symbols. These blocks consist of 3 × 3 symbols arranged in two dimensions as data pages. Three bright symbols make up each block, representing three luminance levels \(I_1\), \(I_2\), and \(I_3\); thus, the 10:9 code is a four-level modulation, including the six dark symbols \(I_0\). Each block must include a symbol with a luminance value of \(I_3\) (255 in 8 bits) as the reference value for normalization during demodulation. As presented in Fig. 2, a spatial light modulator (SLM) displays the data pages and the modulated beam enters into the recording medium as a signal beam. A reference beam without any modulation simultaneously irradiates the same spot of the recording medium, whereupon the data is recorded as interference fringes (holograms). Photopolymer is one of the most commonly-used materials for recording media owing to its long archival lifetime and shelf-life\(^{18, 19}\). We used a photopolymer recording medium in our previous study\(^{10}\), in which we experimentally confirmed the effect of CNN demodulation; thereby, herein, we assumed an HDS system that made use of this material.

During reproduction, only the reference beam enters into the recording medium. A camera captures the reconstructed beam with the data page diffracted by the holograms. The reproduced images are decoded by a signal processor with not only the demodulation but also the error correction process, as the quality of the reproduced data pages is degraded due to the noise in the recording medium and optical path.

Here, we explain a conventional and commonly employed decoding process. During demodulation, the reproduced data pages are initially normalized using the brightest and darkest symbols in each block. The three brightest symbols in the block and others are referred to as bright and dark symbols, respectively. The brightest symbol is automatically assigned the level, \(I_3\), and the luminance levels of the remaining two bright symbols are predicted based on two arbitrary threshold values. This simple demodulation method is also referred to as the hard-decision method.

After demodulation, error correction is performed to correct the errors in the demodulated bits. The sum-product algorithm has a low calculation cost and is widely used to decode the SC-LDPC code. The check and variable nodes exchange log likelihood ratios (LLRs) as messages in the iterative operation to correct the bit errors. In general, the SC-LDPC codes are designed for the additive white Gaussian noise (AWGN) channel as shown below:

\[
y_n = s_n + N_n, \tag{1}
\]

where, \(s_n = \{+1, -1\}\), \(y_n\), and \(N_n\) denote the binary-to-bipolar conversed value from the \(n\)-th channel input \(x_n = \{0, 1\}\), the \(n\)-th channel output, and the noise distribution with zero mean and variance \(\sigma^2\), respectively. The conditional probability distribution, \(p(y_n | x_n)\), is a Gaussian distribution, which is expressed as follows:

\[
p(y_n | x_n) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(y_n - s_n)^2}{2\sigma^2}\right), \tag{2}
\]

and the LLR for the \(n\)-th reproduced bit is defined as follows:

\[
\text{LLR}(n) = \ln \frac{p(y_n | s_n = +1)}{p(y_n | s_n = -1)}. \tag{3}
\]

The conventional LLR in the HDS system is given as follows:\(^{17}\)

\[
\text{LLR}_{\text{conv}}(n) = d^2(B_{\text{rep}}, B_{\text{rem}}(x_n = 0)) - d^2(B_{\text{rep}}, B_{\text{rem}}(x_n = 1)), \tag{4}
\]
where, $d$ denotes the Euclidean distance between two blocks; $B_{rep}$, the standardized reproduced block with bipolar conversion (the maximum is mapped to $-1$, and the minimum is mapped to $1$); and $B_{rep}$, the bipolar converted block that is remodulated from the demodulated bits. Figure 3 presents an example of the LLR calculation process. In this case,

$$
\text{LLR}_{\text{conv}}(9) = (0.33 - (-1.0))^2 - (-1.0 - (-1.0))^2 \\
+ (-0.33 - 0.90)^2 - (1.0 - 0.90)^2 \\
+ (1.0 - 0.49)^2 - (0.33 - 0.49)^2 \\
+ (-1.0 - 0.97)^2 - (1.0 - 0.97)^2 \\
+ (1.0 - (-0.17))^2 - (-0.33 - (-0.17))^2 \\
= 8.7.
$$

The conventional LLR is calculated using the pixel values of reproduced data pages in this approach. However, the noise superimposed on the data pages during the recording and reproduction processes contain various types of optical noise, such as fixed-pattern noise and lens aberrations as well as AWGN. Accordingly, the use of LLRs, based on the AWGN channel in previous studies, cannot take full advantage of the sum-product decoding.

3. Characteristics of Noise in HDS

Figure 4 presents examples of optical noise and their factors in the HDS. There is noise unique to the HDS system as well as general Gaussian noise, such as shot noise or thermal noise in the image sensor, as presented in Fig. 4(a). Aberrations of the optical lens blur bright symbols and the light is diffused and leaks onto adjacent symbols, as presented in Fig. 4(b). This phenomenon is called interpixel interference (IPI) and causes demodulation errors even in the binary modulation code, where the symbols are simply distinguished as bright or dark. The luminance level of the symbols must

![Fig. 3 Example of the calculation process of log likelihood ratios (LLR) using the reproduced pixel values.](image-url)

![Fig. 4 Various optical noise in recording and reproduction processes: (a) random noise, (b) interpixel interference, (c) data lost, (d) brightness unevenness, and (e) resampling noise.](image-url)
be determined in the multi-level modulation code; thus, the influence of IPI must be further taken into consideration. Figure 4(c) presents an example of fixed-pattern noise. The presence of dust on the lens projects shadows onto the bright symbol and darkens the luminance. When the effect of the fixed pattern is too large and causes the bright symbol to be missed, another pixel is incorrectly identified as the bright symbol. As presented in Fig. 4(d), the signal beam shows a nonuniform intensity profile depending on the laser specified in the optical setup, which leads to nonuniform brightness in the reproduced data. The luminance level of the dark symbols increases and vice versa. Figure 4(e) presents the resampling noise caused by image segmentation and resampling in the preprocess during reproduction, due to the relative positional deviation between the pixel of the SLM and the camera and the difference in their pixel sizes. Although accurate image interpolation processes in resampling are commonly employed to compensate for the mismatch, such as bicubic[24] and the Lanczos[25] algorithm, these processes cannot avoid small luminance errors.

CNNs trained by supervised learning can accurately demodulate the data, even in the presence of various optical noise affecting the data. Our previous work experimentally confirmed this effect[10]. The areas of data pages with the most numbers of bit errors were caused by IPI (including resampling noise) [Fig. 4(b) and (e)] and fixed-pattern noise [Fig. 4(c)]; however, the CNN reduced the bit errors by half, with CNN demodulation in these areas. Although the reproduced data pages exhibited nonuniform brightness, the CNNs could accurately demodulate across the entire page data, independent of the region [Fig. 4(d)]. Contrarily, Gaussian noise [Fig. 4(a)], which is not learnable by the CNN owing to its randomness, causes demodulation errors. To overcome this problem and take advantage of the superior properties of the SC-LDPC code for Gaussian noise, we developed an effective decoding method that combines CNN and the SC-LDPC codes.

4. Proposed Decoding System

Figure 5 presents the proposed decoding system, which consists of CNN demodulation and SC-LDPC error correction sections. First, the reproduced data from HDS is input to the CNN. Figure 6 presents the demodulation network. The $3 \times 3$ symbols of the block as well as the neighboring symbols, containing information regarding the effects of IPI, are used to enable the CNN to learn the noise characteristics more accurately; thus, the input size is $5 \times 5^{11}$. The CNN, which comprises convolutional and dense layers, previously learns the noise characteristics through a supervised learning using a dataset of the reproduced images of the data pages and the recorded original bit data as their label information. In the convolutional layer, the input information is subjected to a convolutional operation using filters (referred to as weights), and the features of the input information is extracted. The dense layer generally connects the results of the convolution to the following layer with an arbitrary number of units. Since the dense layer is also the output layer in our CNN, the number of units in this layer is equal to the number of classes to be classified. The convolutional layer utilizes the leaky rectified linear unit function[26], whereas the dense layer utilizes the softmax function[27],

$$t_i = \frac{\exp \left( s_i \right)}{\sum_{i=0}^{N-1} \exp \left( s_i \right)}$$  \hspace{1cm} (6)$$

as an activation function, where $s_i$ and $t_i$ denote the
output and input of the function, respectively, and \( N_c \) denotes the total number of classes. The softmax function accelerates parameter optimization during supervised learning by standardizing all the outputs to be positive and setting their sum to one\(^{26}\). In the 10:9 modulation code, the CNN outputs 1,024 types (\( N_c = 10 \) bits) of class probability, \( p_{cl} \), and the class with the highest probability is the predicted class by the CNN. The bit sequence corresponding to the class contains demodulated bits. The specifications of the CNN are presented in Table I\(^{28}\).

Here, we address the output class probabilities. As previously mentioned, the CNN learns various types of noise characteristics, except random Gaussian noise. Consequently, the output class probability distribution from the CNN represents pure likelihoods, which are only influenced by the untrainable random noise, unlike those calculated by Eq. (4), using reproduced pixel values.

Accordingly, this study adopts the output class probabilities from the CNN for the calculation of LLR in the sum-product decoding. However, the exponential function in the softmax function sets numerous probability components to nearly zero values. Figure 7 displays exemplary outputs of the CNN demodulation when the demodulated bit sequence is “0000001010” (class number is 10). In Fig. 7(a), information on the class probabilities other than class number 10 is almost completely lost, i.e., it discards information regarding demodulation by the softmax function. Therefore, we used a probability distribution, \( p_{fc} \), presented in Fig. 7(b), which is the output of the fully connected calculation (input to the softmax function). We normalized the probability distribution, \( p_{fc} \), from –1 to 1 with bipolar conversion instead of the exponential function. Thereafter, we defined a CNN-based conditional probability distribution, \( p_{CNN} \), as follows:

\[
p_{CNN} = \exp(p_{bpfC})
\]

where, \( p_{bpfC} \) denotes the bipolar converted probability distribution, \( p_{fc} \). Therefore, the LLR of the \( n \)-th bit, \( LLR_{CNN}(n) \), is expressed in our proposed system as

\[
LLR_{CNN}(n) = p_{bpfC}(x_n = 0) - p_{bpfC}(x_n = 1),
\]

by combining Eqs. (3) and (7). The demodulated data are decoded in the sum-product algorithm using the proposed LLR values. The parity check matrix used in the SC-LDPC decoding is created using base matrix \( B \), shown in Fig. 8. Each element of “1” in \( B \) was replaced by an \( M \times M \) permutation matrix, where \( M \) is a lifting number. Table II presents the specifications of the parity check matrix. We set the code rate to 0.65 to ensure high robustness against noise, assuming the introduction of recording density enhancement methods such as angle\(^{29}\) and shift
multiplexing\textsuperscript{30}). Another study of practical use also set the code rate of the error correction code to 0.66\textsuperscript{13}).

5. Results

5.1 Demodulation

We prepared several datasets with varying amounts of noise, as mentioned in Section 3, to validate the decoding characteristics. The conditions of the CNN are presented in Table III. Figure 9 shows the results of the comparison of the demodulation using hard-decision decoding and CNN. The residual bit errors after the demodulation depend on the SNR of the data pages, which is defined by\textsuperscript{31)

\begin{equation}
\text{SNR [dB]} = 10 \log_{10} \left( \frac{\sum_{i=0}^{u-1} \sum_{j=0}^{v-1} O_{ij}^2}{\sum_{i=0}^{u-1} \sum_{j=0}^{v-1} (O_{ij} - R_{ij})^2} \right),
\end{equation}

where, $O_{ij}$ and $R_{ij}$ denote the pixel values of the original (ground truth) and reproduced data pages, respectively, and $u$ and $v$ denote the width and height of the data page in pixels, respectively. When the SNR is too low (SNR < 8.1 dB), the CNN demodulation resulted in numerous bit errors, as the signal components of bright pixels were completely buried in the noise and the CNN failed in learning the noise characteristics. With higher SNR, the CNN became more efficient in demodulating the data. The residual bit error was about one-tenth of that obtained by the conventional hard-decision method.

As shown in Figure 6, the demodulation network is very shallow for the fast demodulation. We evaluated the demodulation speed and using a NVIDIA Quadro GP100 graphics processing unit (GPU), the bit rate became 106 kbps. This was the result when each block was demodulated sequentially. As the blocks are arranged two-dimensionally in one data page, multiple blocks could be demodulated in parallel by means of multiple CNNSs. For instance, in a previous report that used a high-definition size SLM\textsuperscript{10}, the data area totaled 1,740 × 1,044 pixels, excluding the markers for the data region’s detection, and there were 201,840 blocks. If we apply the present result to this, the demodulation speed of up to 21.5 Gbps can be expected on the basis of the parallel CNN demodulation process.

5.2 LLR Distribution

Figure 10 (a) and (b) present the comparison examples of the likelihood distributions calculated using Eqs. (4) and (8), respectively. In the AWGN channel, the ideal LLR must consist of two Gaussian curves. We obtained orange and red dotted lines corresponding to Gaussian fit via nonlinear least-squares fitting. The overlapping region between the two distributions indicates bit errors. When the LLR is obtained using the conventional calculation method, there are numerous spikes in the distributions and deviations from the Gaussian fit. This is because the pixel values used in the calculation of

\begin{table}[h]
\centering
\begin{tabular}{|c|c|}
\hline
Epochs in training & 300 \\
Number of blocks for training & 665,520 \\
Number of blocks for test & 201,840 \\
\hline
\end{tabular}
\caption{Experimental conditions in supervised training.}
\end{table}

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{fig9}
\caption{Demodulation results.}
\end{figure}

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{fig10}
\caption{Comparison of the LLR calculated using (a) reproduced pixel values and (b) bipolar-conversed probability distribution.}
\end{figure}
LLR include various optical noise effects. Contrarily, the LLR calculated using the CNN-based conditional probability distribution has a clean, smooth shape and is well fitted to the Gaussian curves. It can be inferred that the CNN learned the noise information accurately, except for AWGN and the CNN probability distribution contained only Gaussian noise.

5.3 Error Correction Performance

The SC-LDPC decoding corrects the residual bit errors after demodulation. Figure 11 presents a comparison example of the decoding data with an SNR of 13.2 dB using the conventional and proposed LLR distributions by the sum-product algorithm. The difference in the initial bit error rates in Fig. 11 (a) and (b), even though the same reproduced data is decoded, is because the demodulation methods vary for the hard-decision method and CNN. When the LLR calculated using the conventional method is employed, the residual bit errors are not corrected, despite an increase in the number of iterations. We evaluated the relationship between SNR and decodability. We set the maximum number of iterations to 10 in the sum-product algorithm to prevent the decoding speed from becoming a bottleneck in the overall HDS system. As a result, the maximum permissible bit error rate after the demodulation was $1.8 \times 10^{-2}$ when the conventional LLR was employed. As presented in Fig. 9, the SNR required for an error-free decoding in the HDS system is 17.9 dB. Similarly, we evaluated the CNN-demodulated data using the proposed LLR. We were able to correct all errors, as long as the bit error rate after the CNN demodulation was less than $2.3 \times 10^{-2}$, which meant that the required SNR of the data was 8.23 dB. Thus, the combination of CNN demodulation and SC-LDPC decoding based on the output likelihood distribution from the CNN improved the SNR required for the HDS system to 9.67 dB.

6. Conclusions

We developed an efficient decoding system for HDS by combining CNN demodulation and SC-LDPC decoding. The CNN uses supervised learning to learn how to accurately demodulate the recorded signals from noise-superimposed data. Based on the learning results, when reproduced data is input, the CNN provides the output probability distributions, whose bit sequence is plausible for the reproduced data. As the trained CNN cannot learn the characteristics of Gaussian noise owing to its randomness, only Gaussian noise is superimposed on the output probability distributions. We evaluated the demodulation capability using data with added noise, which yielded bit errors reduced to about one-tenth.

The sum-product algorithm for the SC-LDPC decoding significantly corrects residual errors after demodulation. In particular, the LLR used in the iterative decoding process is simply defined when the channel is AWGN. We designed and defined the LLR using the output probability distribution from the CNN. Due to the elimination of various noise components, except Gaussian noise, from the LLR, more powerful error correction is provided in the decoding. We confirm that the trained CNN provided output probabilities where the complex noise components in the HDS were clearly eliminated. The required SNR for the optical system improved by approximately 10 dB using the CNN-based LLR.

Fig. 11 Decoding performance using (a) conventional LLR and (b) proposed CNN-based LLR.

References

1) D. Psaltis and G.W. Burr: “Holographic data storage”, Cover Feature, 9 (1998)
2) L. Hesselink, S.S. Orlov and M.C. Bashaw: “Holographic data storage systems”, Proc. IEEE, 92, pp.1231-1280 (2004)
3) L. Dhar, K. Curtis and T. Facke: “Coming of age”, Nature Photonics, 2, pp.403-405 (2008)
4) T. Hoshizawa, K. Shimada, K. Fujita and Y. Tada: “Practical angular-multiplexing holographic data storage system with 2 terabyte capacity and 1 gigabit transfer rate”, Jpn. J. Appl. Phys., 55, pp.09SA06-01-09SA06-09 (2016)
5) N. Kinoshita, Y. Katano, T. Muroi and N. Saito: “Demonstration of 8K SHV Playback from Holographic Data Storage”, in International Symposium on Optical Memory (2015)
6) Y. Katano, T. Muroi, N. Kinoshita and N. Ishii: “Prototype holographic data storage drive with wavefront compensation for playback of 8K video data”, IEEE Trans. Consum. Electron., 63, pp.243-250 (2017)
7) S.Y. Kim and J. Lee: “A Simple 2/3 Modulation Code for Multi-Level Holographic Data Storage”, Jpn. J. Appl. Phys., 52, pp.09LE04-01-09LE04-03 (2013)
8) J. Liu, H. Horimai, Y. Huang and X. Tan: “Phase modulated high density collinear holographic data storage system with phase-retrieval reference beam locking and orthogonal reference encoding”, Opt. Express, 26, pp.R3828-R3838 (2018)
9) T. Nobukawa and T. Nomura: “Multilevel recording of complex amplitude data pages in a holographic data storage system using digital holography”, Opt. Express, 24, pp.21001-21011 (2016)
10) Y. Katano, T. Muroi, N. Kinoshita, N. Ishii and N. Hayashi: “Data demodulation using convolutional neural networks for holographic data storage”, Jpn. J. Appl. Phys., 57, pp.09SC01-01-09SC01-05 (2018)
11) Y. Katano, T. Muroi, N. Kinoshita and N. Ishii: “Demodulation of Multi-Level Data using Convolutional Neural Network in Holographic Data Storage”, in 2018 Digital Image Computing: Techniques and Applications (DICTA) (2018)
12) N. Mark Allen and C. Wu-Chun: “Likelihood-based error correction for holographic storage systems”, in Proc.SPIE (1999)
13) Y. Nakamura and T. Hoshizawa: “Two high-density recording methods with run-length limited turbo code for holographic data storage system”, Jpn. J. Appl. Phys., 55, pp.09SA01-01-09SA01-10 (2016)
14) H. Hayashi and K. Kimura: “Low-Density Parity-Check Coding for Holographic Data Storage”, Jpn. J. Appl. Phys., 44, pp.3495-3498 (2005)
15) S. Kudekar, T.J. Richardson and R.L. Urbanke: “Threshold Saturation via Spatial Coupling: Why Convolutional LDPC Ensembles Perform So Well over the BEC”, IEEE Trans. Inf. Theory, 57, pp.803-834 (2011)
16) R. Gallager: “Low-density-parity-check codes”, IEEE Trans. Inf. Theory, 8, pp.21-28 (1962)
17) N. Ishii, Y. Katano, T. Muroi and N. Kinoshita: “Spatially coupled low-density-parity-check error correction for holographic data storage”, Jpn. J. Appl. Phys., 56, pp.09NA03-01-09NA03-04 (2017)
18) W. Patrick, I. Ben, S. Melinda, Q. Sean, B. David, S. Songvit, T. Tim, C. Michael, A. Fred, M. David, M. Sam, H. Adrian, W. William and D. Lisa: “Photopolymer media for holographic storage at ~ 405 nm”, in Proc.SPIE (2004)
19) A. Fred, R.A. Mark and C.U. Adam: “High dynamic range holographic data storage media”, in Proc.SPIE (2015)
20) L.D. Ramamoorthy and B.V. K. Vijaya Kumar: “Two-Dimensional Equalization and Error Correction Using Low Density Parity Check Codes for Holographic Data Storage”, Jpn. J. Appl. Phys., 45, pp.1305-1310 (2006)
21) N. Kinoshita, T. Muroi, N. Ishii, K. Kamijo and N. Shimidzu: “Classification and Evaluation of Noises in Holographic Memory System”, Jpn. J. Appl. Phys., 49, pp.08KD12-01-08KD12-02 (2010)
22) M.P. Bernal, G.W. Burr, H. Coufal, R.K. Grygier, J.A. Hoffnagle, C.M. Jefferson, E. Oesterschulze, R.M. Shelby, G.T. Sincerbox and M. Quintanilla: “Effects of multilevel phase masks on interferplex cross talk in digital holographic storage”, Appl. Opt., 36, pp.3107-3115 (1997)
23) J.F. Heaneue, K. Gurkan and L. Hesselink: “Signal detection for page-accessoptical memories with intersymbol interference”, Appl. Opt., 35, pp.2431-2438 (1996)
24) R. Keys: “Cubic convolution interpolation for digital image processing”, IEEE Trans. Acoust., 29, pp.1153-1160 (1981)
25) W. Burger and M.J. Burge: Principles of Digital Image Processing: Core Algorithms (Springer Publishing Company, Incorporated, 2009), pp.223-232
26) A.L. Maas, A.Y. Hannun and A.Y. Ng: “Rectifier Nonlinearities Improve Neural Network Acoustic Models”, in Proceedings of the 30th International Conference on Machine Learning (2013)
27) S. Lawrence, C.L. Giles, A.C. Tsoi and A.D. Back: “Face recognition: A convolutional neural-network approach”, IEEE Trans Neural Netw, 8, pp.98-113 (1997)
28) D.P. Kingma and J.L. Ba: “Adam: A Method for Stochastic Optimization”, in 3rd International Conference for Learning Representations, Adam: A Method for Stochastic Optimization (2014)
29) X. An and D. Psaltis: “Experimental characterization of an angle-multiplexed holographic memory”, Opt. Lett., 20 (1995)
30) Z. Ushiyama, H. Kurata, Y. Tsukamoto, S. Yoshida and M. Yamamoto: “Shift-Peristrophic Multiplexing for High Density Holographic Data Storage”, Applied Sciences, 4 (2014)
31) H.C. Gonzalez and R.E. Woods: Digital Image Processing, Third Edition (Pearson, 2007), pp.374-376

Yuutarō Katano received his B. Eng. and M. Eng. degrees from Waseda University, Tokyo, Japan, in 2009 and 2011, respectively. In 2011, he joined Japan Broadcasting Corporation (NHK), Tokyo, Japan. Since 2014, he has been with NHK Science & Technology Research Laboratories and has been engaged in research on holographic data storage and computational photography.

Teruyoshi Nobukawa received his B. Eng., M. Eng., and Ph. D. degrees from Waseda University, Japan, in 2012, 2014, and 2017, respectively. He worked as a Post-Doctoral Fellow at Japan Broadcasting Corporation (NHK), Tokyo, Japan before joining NHK in 2017, and has been with NHK Science & Technology Research Laboratories since then. He has been engaged in research on holographic data storage and computational photography.

Tetsuhiko Muroi received his B. Eng., M. Eng., and Ph. D. degrees from Keio University, Tokyo, Japan, in 1995 and 1997. He received his Ph. D. degree in engineering from Keio University, Japan, in 2011. In 1997, he joined Japan Broadcasting Corporation (NHK). Since 2000, he has been with NHK Science & Technology Research Laboratories (STRL), Tokyo, Japan, and has been engaged in research on optical disc, holographic data storage, and computational photography. He is currently a senior manager in STRL’s Planning & Coordination Division.

Nobuhiro Kinoshita received his B. Eng. and M. Eng. degrees from Doshisha University, Kyoto, Japan, in 1995 and 1997. He received his Ph. D. degree in engineering from Keio University, Kobe, Japan, in 2011. In 1997, he joined Japan Broadcasting Corporation (NHK). Since 2000, he has been with NHK Science & Technology Research Laboratories (STRL), Tokyo, Japan, and has been engaged in research on optical disc, holographic data storage, and computational photography at NHK Science & Technology Research Laboratories, Tokyo, Japan.