PAPER

Sparse DP Quantization Algorithm

Yukihiro BANDOH[†a)], **Seishi TAKAMURA**[†], ***and*** **Atsushi SHIMIZU**[†], ***Senior Members***

SUMMARY We formulate the design of an optimal quantizer as an optimization problem that finds the quantization indices that minimize quantization error. As a solution of the optimization problem, an approach based on dynamic programming, which is called DP quantization, is proposed. It is observed that quantized signals do not always contain all kinds of signal values which can be represented with given bit-depth. This property is called amplitude sparseness. Because quantization is the amplitude discretization of signal value, amplitude sparseness is closely related to quantizer design. Signal values with zero frequency do not impact quantization error, so there is the potential to reduce the complexity of the optimal quantizer by not computing signal values that have zero frequency. However, conventional methods for DP quantization were not designed to consider amplitude sparseness, and so fail to reduce complexity. The proposed algorithm offers a reduced complexity optimal quantizer that minimizes quantization error while addressing amplitude sparseness. Experimental results show that the proposed algorithm can achieve complexity reduction over conventional DP quantization by 82.9 to 84.2% on average.

key words: *quantization, sparseness, dynamic programming*

1. Introduction

The purpose of quantization [1] is to generate codewords (quantization indices) based on a given metric. The design of the optimal quantizer leads to a kind of minimization problem, that is, how to generate the quantization indices that can minimize a distortion (quantization error) caused by a quantization process. A typical form of quantization error is summation of square error (SSE). Quantization schemes are classified into two types: conversion from continuous signal to discrete one, and conversion from fine discrete signal to coarse discrete one. This manuscript focuses on the latter type. The latter type is a kind of bit depth conversion, and is required for display adaptation [2], [3], bit-depth scalable coding [4], [5] and HDR video coding [6].

The approaches proposed to solve the above-mentioned minimization problem fall into two types: analytical optimization (which calculates the optimal solution analytically) and numerical optimization (which uses numerical computation). If the probability density function (PDF) of quantized data can be represented in particular parametric-form, for example, a uniform distribution, Gauss distribution or Laplace distribution, analytical optimization can be adopted where the codewords for symbols generated from these PDFs are analytically optimizes. However, the PDF of real quantized data generally can not be represented in the desired parametric forms. Therefore, such analytical optimization approaches generally cannot generate optimal quantizers for real data.

Consequently, numerical optimization approaches, which do not require any particular parametric form of PDF are more common. Typical of this result is the Lloyd-Max quantization algorithm (hereafter LM quantization) [7], [8], which generates quantization indices and boundaries of quantization bins iteratively, until a given convergence condition is satisfied.

However, two problems with this algorithm have been pointed out. First, LM quantization can not guarantee the optimal solution. This is because the algorithm is designed based on a necessary condition for optimal quantization. LM quantization can generate the optimal solution, only if the logarithm function of the PDF for quantized data offers convexity. For example, the above convexity condition is satisfied if the PDF follows a uniform distribution, Gauss distribution or Laplace distribution. On the other hand, when the quantized data does not satisfy the convexity condition, the most common case, LM quantization may fall into local minimum. It depends on the initial codewords as to whether LM quantization yields the optimal solution or not. Note that no specific strategy has been adopted for optimizing initial codewords. Second, the computation complexity of LM quantization can not be evaluated. This is because the algorithm is based on an iterative process and its convergence depends on the initial codewords.

In order to design optimal quantizers, adaptive quantization algorithms based on dynamic programming (henceforth, abbreviated to DP quantization) have been studied. Bruce [9] applied the principle of optimality in dynamic programming to optimizing quantizer, and showed that the complexity associated with designing an optimizing quantizer can be reduced from exponential time to polynomial time. Sharma [10] proposed a low complexity algorithm for designing a DP quantizer that minimizes the quantization error subject to convexity constraint. Wu [11] proposed an algorithm to reduce the complexity of optimal path finding in DP quantization using matrix search.

As a kind of the above-mentioned quantization, bit-depth conversion (BDC) is used in image processing. BDC transforms the bit-depth of input signal and generates signal with lower bit-depth. For example, BDC is used to adjust high bit-depth signal to legacy displays which do not support high bit-depth signal. Furthermore, BDC plays an impor-

Manuscript received May 23, 2018.
Manuscript revised October 27, 2018.

[†]The atuthors are with NTT Media Interigence Laboratories, NTT Corporation, Yokosuka-shi, 239-0847 Japan.
a) E-mail: yukihiro.bandou.pe@hco.ntt.co.jp

DOI: 10.1587/transfun.E102.A.553

Copyright © 2019 The Institute of Electronics, Information and Communication Engineers

tant role in bit-depth scalable codec [4], [5], [12], [13] as a key process for generating a layer-structured data that consists of a base layer and enhancement layers. BDC separates an input signal of the encoder into signal for the base layer and those for enhancement layers. The base layer is constructed to have backward compatibility to a decoder that does not support high bit-depth signal.

When designing optimal quantizers for image signals, it is important to note that most image signals feature amplitude sparseness of signal value, that is, pixel value. In other words, the signal values do not fully utilize the given bit-depth. For example, if an image whose bit-depth is 10 bits exhibits sparseness, it contains fewer than 1024 signal values, although the bit-depth can represent up to 1024 signal values. Some studies on image coding report that coding efficiency can be improved by considering amplitude sparseness. Lossless coding algorithms [14], [15] and a near-lossless coding algorithm [16] improve coding efficiency by utilizing fewer pixel values for images with amplitude sparseness.

The histogram of an image with sparseness has some insignificant elements, that is, their frequency is zero (hereafter called zero-frequency). Signal values that have zero-frequency do not impact the quantization error. Therefore, by appropriately suppressing the quantization process for zero-frequency signal values, we can expect to reduce the complexity while still minimizing quantization error. However, conventional DP quantization methods do not consider sparseness, suggesting that there is room for further reductions in the complexity of DP quantization. Authors proposed a basic scheme for reducing the complexity of DP quantization based on sparseness in [17], [18]. This paper enhances the basic studies [17], [18] for the following three points. Firstly, this paper presents a complete algorithm that reduces the complexity of DP quantization for images with sparseness, while still minimizing quantization error. The proposed algorithm more strictly restricts the search range of DP quantization than the basic scheme of [17], [18], from the viewpoint of the evaluation of upper bounds and lower bounds of the search range. Secondly, this paper enhances experimental results through evaluations on more kinds of image contents than those used in [17], [18]. Furthermore, these image contents have higher special resolutions, higher bit-depth and wider color-gamut than those in [17], [18]. Finally, this paper discusses the complexity of proposed algorithm based on statistical tests of numerical simulations.

This paper is organized as follows. Section 2 formulates the problem of quantizer optimization. Section 3 introduces DP quantization as the basic algorithm of our proposed method. Section 4 interprets DP quantization using a trellis transition diagram in order to facilitate the understanding of our proposed method. Section 5 provides sparse DP quantization; it extends DP quantization by utilizing input signal sparseness. As reference information, notations used in Sects. 2 to 5 are summarized in Table 1. Section 6 details the experiments done to evaluate the proposed method. Finally, Sect. 7 presents our conclusions.

2. Formulation of Quantizer Design

In this section, we formulate the design of a quantizer that translates a K-level discrete signal to a M-level equivalent ($M < K$). For this formulation, we use the histogram of the signal as the input to the quantizer. The k-th element of the histogram is $h[k]$ ($k = 0, \cdots, K-1$), which is the frequency of signal value k. For example, in the case of an 8-bit signal, the range of k is 0 to 255. The formulated quantizer is defined using two parameters Δ_m and L_m; Δ_m is the length of the m-th sub-interval of the histogram. L_m is the upper boundary of the m-th sub-interval in the histogram. In the following, L_m is simply called boundary. The boundaries are described as follows:

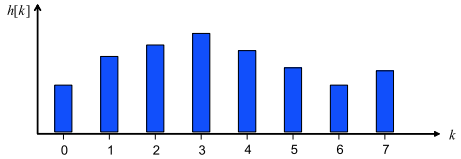
$$\begin{cases} L_m = \sum_{j=0}^{m} \Delta_j - 1 \quad (m = 0, \cdots, M-2) \\ L_{M-1} = K - 1 \end{cases} \tag{1}$$

Henceforth, the m-th interval $[L_m - (\Delta_m - 1), L_m]$ of the histogram is called the m-th bin. Since each bin is set to have at least one element, L_m ($0 \le m \le M-2$) is restricted in the following range:

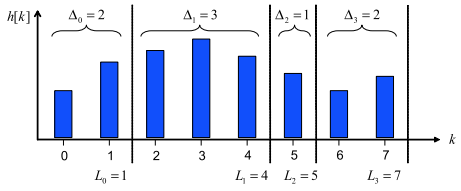
$$m \le L_m \le K - (M - m) \tag{2}$$

Figure 1 illustrates the above-mentioned parameters for the case where the histogram with eight elements ($K = 8$) is quantized to one with four bins ($M = 4$). Figure 1 (b) shows each bin contains $2(= \Delta_0)$ elements, $3(= \Delta_1)$ elements, $1(= \Delta_2)$ element, and $2(= \Delta_3)$ elements of the input histogram, respectively; and the upper boundary of each bin becomes $L_0 = \Delta_0 - 1 = 1$, $L_1 = L_0 + \Delta_1 = 4$, $L_2 = L_1 + \Delta_2 = 5$, and $L_3 = L_2 + \Delta_3 = 7$, respectively.

Quantizer design is based on minimizing the quantization error created by approximating all values in the m-th bin $[L_m - (\Delta_m - 1), L_m]$ in the histogram with representative value $\hat{c}(\Delta_m, L_m)$. As the quantization error of the m-th bin $[L_m - (\Delta_m - 1), L_m]$, we use the summation of square error



(a) an 8-level histogram



(b) the 4-level histogram generated by quantizing the 8-level histogram of figure 1 (a)

Fig. 1 Example of parameters for quantization.

Table 1 Notations.

Symbol	Description
K	the number of levels of input signal
M	the number of levels of quantized signal
$h[k]$	the k-th element ($k = 0, \cdots, K-1$) of the histogram of input signal, which is abbreviated "the histogram" in this table
L_m	the upper boundary of the m-th interval in the histogram ($m = 0, \cdots, M-1$)
Δ_m	the length of the m-th interval of the histogram ($m = 0, \cdots, M-1$)
$e(\Delta_m, L_m)$	quantization error of the m-th interval $[L_m - (\Delta_m - 1), L_m]$ of the histogram
$S_m[L_m]$	the minimum summation of quantization error in case that the interval $[0, L_m]$ of the histogram is divided into $m+1$ sub-intervals
$\tilde{K}$	the number of the significant elements of the histogram
$\Psi_u[m]$	the maximum significant element index of the boundary of the m-th bin ($m = 0, \cdots, M-1$)
$\Psi_l[m]$	the minimum significant element index of the boundary of the m-th bin ($m = 0, \cdots, M-1$)
$F[\tilde{k}]$	the *element index* [1] corresponding to the $\tilde{k}$-th significant element [2] of the histogram ($\tilde{k} = 0, \cdots, \tilde{K}-1$)
$G[k]$	the *significant element index* [3] corresponding to the k-th element of the histogram ($k = 0, \cdots, K-1$);
$\psi_l[m]$	the minimum element index for significant elements belonging to interval $[m, m-M+K]$ of the histogram ($m = 0, \cdots, M-1$)
$\psi_u[m-M+K]$	the maximum element index for significant elements belonging to interval $[m, m-M+K]$ of the histogram ($m = 0, \cdots, M-1$)
$\rho_u[m]$	the maximum number of consecutive insignificant elements in interval $[m-M+K, K-1]$ of the histogram ($m = 0, \cdots, M-1$)
$\rho_l[m]$	the maximum number of consecutive insignificant elements in interval $[0, m]$ of the histogram ($m = 0, \cdots, M-1$)
$\tilde{S}_m[\tilde{L}_m]$	minimum summation of quantization error in case that the interval $[0, F[\tilde{L}_m]]$ of the histogram is divided into $m+1$ sub-intervals
$E[\tilde{\Delta}_m, \tilde{L}_m]$	quantization error of the interval $[F[\tilde{L}_m - (\tilde{\Delta}_m - 1)], F[\tilde{L}_m]]$ of the histogram
$T_{m-1}[\tilde{L}_m]$	the optimal boundary of the $m-1$-th bin next to the m-th bin with boundary $\tilde{L}_m$
$\tilde{\Delta}_m^{(\tilde{L}_m)}$	$\tilde{\Delta}_m$ which minimizes the right side of Eq. (11)

[1] *Element index* is an index to identify each element of the histogram.
[2] The significant elements are listed in a sequence and each significant element in the sequence is referred by index $\tilde{k}$.
[3] *Significant element index* is an index to identify each significant element of the histogram. Note that look-up table $G[k]$ provides the inverse mapping of look-up table $F[\tilde{k}]$, and vice versa.

$e(\Delta_m, L_m)$ defined as follows:

$$e(\Delta_m, L_m) = \sum_{k=L_m-\Delta_m+1}^{L_m} \{k - \hat{c}(\Delta_m, L_m)\}^2 h[k] \tag{3}$$

where $\hat{c}(\Delta_m, L_m)$ is the integer value that is the closest to the centroid of the m-th bin $[L_m - (\Delta_m - 1), L_m]$. The centroid is defined as follows:

$$c(\Delta_m, L_m) = \frac{\sum_{k=L_m-(\Delta_m-1)}^{L_m} k h[k]}{\sum_{k=L_m-(\Delta_m-1)}^{L_m} h[k]} \tag{4}$$

Note that each bin is set so that the denominator of equation (4) does not become zero. In other words, each bin is set so as to include at least one significant element. Optimizing the quantizer means finding the parameters that minimize the following summation of quantization error

$$(\Delta_0^*, \cdots, \Delta_{M-1}^*) = \underset{\Delta_0, \cdots, \Delta_{M-1}}{\arg\min} \left\{ \sum_{m=0}^{M-1} e(\Delta_m, L_m) \right\} \tag{5}$$

3. DP Quantization

In this section, we describe DP quantization, the basic algorithm of our proposed method. This description of DP quantization will help to clarify the difference between DP quantization and our proposed method described in Sect. 5 and allow a better understanding of our proposed method.

In the optimization problem of Eq. (5), the number of combinations of M parameters $(\Delta_0, \cdots, \Delta_{M-1})$ grows exponentially. Using the brute force method to search for the optimal combination, $(\Delta_0^*, \cdots, \Delta_{M-1}^*)$, takes exponential time and is not realistic in terms of complexity.

Considering that the quantization error $e(\Delta_m, L_m)$ of the m-th bin depends on the boundary L_m of the m-th bin and the width Δ_m of the same bin, dynamic programming based approaches (DP quantization) [9]–[11] have been used to solve the optimization problem of Eq. (5).

DP quantization focuses on a recurrence relation of quantization error. We define $S_m[L_m]$ for each L_m ($m = 0, \cdots, M-1$) as the minimum summation of quantization error $\sum_{i=0}^{m} e(\Delta_i, L_i)$ where the interval $[0, L_m]$ of histogram $h[k]$ ($k = 0, \cdots, K-1$) is divided into $m+1$ bins. Since $e(\Delta_m, L_m)$ depends on L_m and Δ_m, $S_m[L_m]$ can be expressed using $S_{m-1}[L_m - \Delta_m]$ in the following recursive equation:

$$S_m[L_m] = \min_{\Delta_m} \{S_{m-1}[L_m - \Delta_m] + e(\Delta_m, L_m)\} \tag{6}$$

where $m = 1, \cdots, M-1$ and $L_m = m, \cdots, K-(M-m)$. Using Eq. (6), the computation of $S_m[L_m]$ results in the selection of the best parameter among the values of $\Delta_m = 1, \cdots, L_m - m + 1$. The range of Δ_m is described in Appendix A.

Applying Eq. (6) to Eq. (5), the minimization problem of Eq. (5) becomes as follows:

$$\min_{\Delta_{M-1}} \{S_{M-2}[L_{M-1} - \Delta_{M-1}] + e(\Delta_{M-1}, L_{M-1})\} \tag{7}$$

Furthermore, applying Eq. (6) recursively and noting $L_{M-1} = K-1$ in Eq. (1), the minimization problem of Eq. (5) is to find the optimal solution $(\Delta_0^*, \cdots, \Delta_{M-1}^*)$ from among $\frac{1}{2}M^3 - \frac{2K+5}{2}M^2 + \frac{K^2+7K+4}{2}M - K^2 - K$ candidates†. Thus,

†This number is derived in Appendix C as the number of candidate paths in a trellis diagram that is described in Sect. 4.

DP quantization can provide a polynomial time solution to the minimization problem.

4. Interpretation of DP Quantization as Optimal Path Search

Using a trellis transition diagram, we provide an interpretation of the optimization process of DP quantization. This interpretation will be useful in understanding the proposed algorithm described in Sect. 5. The trellis transition diagram of Fig. 2 illustrates the quantization result for the example shown in Fig. 1(b). In Fig. 2, the vertical axis and the horizontal axis represent signal values $k \in \{0, 1, \cdots, 7\}$ and quantization indices $m \in \{0, 1, 2, 3\}$, respectively. The node at (k, m) in the trellis transition diagram has a cost value that is the minimum summation of quantization error caused by approximating interval $[0, k]$ in the histogram with $m+1$ levels. For example, the node at $(4, 1)$ has the minimum summation of quantization error that is generated by using two representative values to quantize the five elements $(k = 0, \cdots, 4)$ of the histogram. Note that the node on the bottom-left corner is a dummy node introduced as the start node and does not have a cost value. Each path in the trellis transition diagram has a cost that is quantization error for a histogram interval determined by both endpoints of the path. For example, the cost of the path from node $(1, 0)$ to node $(4, 1)$ becomes the quantization error caused by quantizing the histogram interval $[2, 4]$ as the second bin. Quantization shown in Fig. 1(b) corresponds to traversed paths indicated by the bold blue line in the trellis transition diagram of Fig. 2. The horizontal displacement of each traversed path is its interval length. For example, traversed paths indicated by the bold blue line have four paths whose horizontal displacements are $2(= \Delta_0)$, $3(= \Delta_1)$, $1(= \Delta_2)$ and $2(= \Delta_3)$, respectively. Thus, the design of the optimal quantizer can be represented as the optimal path search over the trellis transition diagram.

Using the trellis transition diagram, we can interpret the reduction in complexity of optimal quantization offered by dynamic programming as follows. Let us focus on a red node in Fig. 3. The red node has four traversable paths from gray nodes. The traversable paths are characterized by horizontal displacements (1, 2, 3, and 4) corresponding to Δ_2. When we try to find the optimal path up to the red node in Fig. 3, it is not necessary to search all paths from the start node to the red node. It is enough to search paths from each gray node to the red node. This is because each gray node has an accumulated cost that is equal to the minimum summation of quantization error corresponding to the optimal path from the start node up to each gray node. Evaluating the summation of the accumulated cost stored in a gray node and the cost provided on the path from the gray node to the red node, we can identify the minimum summation of quantization error as corresponding to the optimal path that connects the start node to the red node through the gray node.

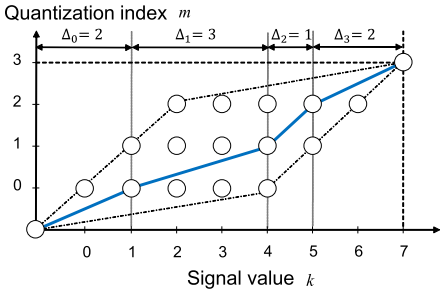


Fig. 2 Traversed path corresponding to quantization shown in Fig. 1(b).

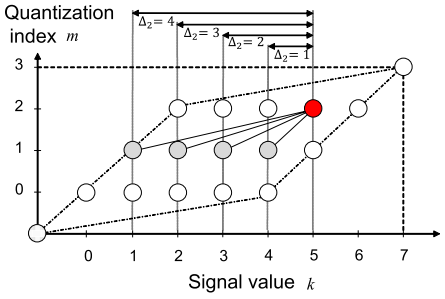


Fig. 3 Optimal path search for red node.

5. Sparse DP Quantization

5.1 Focus for Complexity Reduction

In this section, we introduce a complexity reduction algorithm for DP quantization that focuses on insignificant elements with zero-frequency. When the signal value $L_m + 1$ has zero-frequency, that is, $h[L_m + 1] = 0$, the quantization error of interval $[L_m - (\Delta_m - 1), L_m + 1]$ in the histogram is equal to that of interval $[L_m - (\Delta_m - 1), L_m]$. This is because $h[L_m + 1](= 0)$ has no effect on the quantization error. Thus, in the case of $h[L_m + 1] = 0$, we can skip the computation of $S_m[L_m + 1]$ indicated by Eq. (6). In other words, for minimization of quantization error, it is enough to consider only significant elements whose frequencies are non-zero.

In order to verify sparseness of signal values, we assessed standard images described later in Sect. 6. The sparseness of each color channel is defined as the ratio of the number of insignificant elements to the number of all elements, as follows:

$$\text{Sparseness} = \frac{\text{the number of insignificant elements}}{\text{the number of all elements}} \tag{8}$$

Table 2 confirms that all images examined have sparseness to some extent.

5.2 Restriction of Search Range Considering Sparseness

Considering sparseness, it is possible to skip some elements

Table 2 Sparseness of standard images (cells in the "Sparseness" column represent values given by Eq. (8)).

Image	G-channel Sparseness [%]	R-channel Sparseness [%]	B-channel Sparseness [%]
Image01	64.1	64.1	62.5
Image02	62.4	62.5	62.4
Image03	63.4	63.4	63.4
Image04	63.4	63.7	63.4
Image05	63.4	63.4	63.4
Image06	63.4	63.4	63.4
Image07	63.5	63.7	66.4
Image08	64.0	64.4	63.4
Image09	63.5	63.7	65.3
Image10	65.4	65.5	67.5
Image11	56.6	57.1	56.4
Image12	57.7	56.3	59.4
Image13	69.7	69.4	67.1
Image14	57.9	57.8	56.8
Image15	64.9	65	62.7
Image16	62.1	61.3	60.4
Image17	58.2	60.9	56.8
Image18	58.4	60.6	56.4
Image19	63.5	62.1	59.4
Image20	57.4	58.2	56.6
Image21	57.9	56.5	60
Image22	57.3	59.7	56.4
Image23	57.2	56.8	56.5
Image24	57.7	58.3	56.3
Image25	57.3	59.3	56.5
Image26	58.1	56.5	56.3

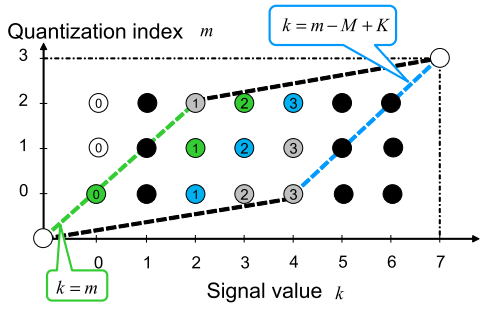


Fig. 4 Insignificant nodes (black circles) and significant nodes (circles with numeral) in the case of $K = 8$ and $M = 4$.

of a histogram in DP quantization. Before explaining detailed algorithm, we provide its basic idea using a toy example. In preparation for description of the proposed quantization algorithm, we define some symbols and terminologies below. Index k identifies an element of histogram $h[k]$ ($k = 0, \cdots, K-1$). The index is called *element index*. $\tilde{K}$ represents the number of significant elements of the histogram. Those significant elements are listed in a sequence and each significant element in the sequence is referred by index $\tilde{k}$ ($\tilde{k} = 0, \cdots, \tilde{K}-1$). The index is called *significant element index*.

Figure 4 provides an example of a trellis transition diagram whose elements with $k = 1, 5, 6$ are insignificant for the case of $M = 4$ and $K = 8$. The nodes in the trellis transition diagram are classified into two types, significant node and insignificant node. Significant node is located in the position where the abscissa is a significant element. Insignificant node is located in the position where the abscissa is an insignificant element. In this figure, significant nodes and insignificant nodes are represented by circles with numeral and black circles, respectively. The numeral in each circle represents the significant element index.

As described in Sect. 2, the boundary of each bin is in the range defined by inequality (2). In Fig. 4, the blue broken line and the green broken line correspond to superior end and inferior end defined by inequality (2). Thus, the optimal path is restricted to candidates that pass through nodes located between the green broken line and the blue broken line.

If there are any insignificant elements in a quantized histogram, insignificant elements can be excluded from the candidates for the boundary of each bin. From the viewpoint of searching the optimal path in the trellis transition diagram, it is not necessary to search paths via insignificant nodes. Thus, the search range for the optimal path is restricted to candidates that pass through significant nodes within a parallelogram area surrounded by broken-lines.

Furthermore, the leftmost nodes within the search range can be additionally restricted according to the source nodes of the transition. Let us consider the leftmost node within the search range in each row in the trellis transition diagram. In the followings, the node located at the position of $k = \kappa$ and $m = \mu$ is referred as "node (κ, μ)". In the case of the 0-th row ($m = 0$), the node $(0, 0)$ on the green broken line is a significant node. This node is identified as the leftmost node in this row. In the case of the first row ($m = 1$), the node $(1, 1)$ on the green broken line is an insignificant node. In the right side of the green broken line, the closest significant node to the node $(1, 1)$ is node $(2, 1)$ indicated by a green circle with the numeral of one. In the case of the second row ($m = 2$), although node $(2, 2)$ on the green broken line is a significant node, this node is excluded from candidates for searching the optimal path. This is because the transition source to the node $(2, 2)$ is only node $(1, 1)$, which is excluded from candidates for searching the optimal path as stated in the case of $m = 1$. Thus, as the leftmost node in this row, node $(3, 2)$ is identified.

Similarly, the rightmost nodes within the search range can be additionally restricted according to the destination nodes of the transition. Let us consider the rightmost node within the search range in each row. In the case of $m = 1$, node $(5, 1)$ on the blue broken line is an insignificant node. In the left side of the blue broken line, the closest significant node to the node $(5, 1)$ is node $(4, 1)$. But, this node is excluded from candidates for searching the optimal path as well as insignificant nodes. This is because there are no possible paths to transit from the node $(4, 1)$. The destination to transit from the node $(4, 1)$ is only node $(5, 2)$ and $(6, 2)$, which are insignificant nodes. Thus, as the rightmost node in this row, we select node $(3, 1)$ indicated by a blue circle with the numeral of two. In the case of $m = 0$, as with the case of $m = 1$, although node $(4, 0)$ on the blue broken line and node $(3, 0)$ are significant nodes, these nodes are

excluded from candidates for searching the optimal path. This is because there are no possible paths to transit from the node $(4, 0)$ and $(3, 0)$. The destination to transit from the node $(4, 0)$ is only node $(5, 1)$ which is an insignificant node, and those from the node $(3, 0)$ are node $(5, 1)$ and $(4, 1)$ which are excluded from candidates for searching the optimal path as stated in the case of $m = 1$. Thus, as the rightmost node in this row, we select node $(2, 0)$ indicated by a blue circle with the numeral of one.

The above-illustrated restriction of the nodes in the trellis diagram are formulated as the range specification of the index of each bin. The leftmost node within the search range in the m-th row is identified as the minimum significant element index of the boundary of m-th bin, which is referred as the lower limit of the m-th bin, hereinafter. For the lower limit of the m-th bin, table $\Psi_l[m]$ $(m = 0, \cdots, M-1)$ is prepared. $\Psi_l[m]$ $(m = 0, \cdots, M-1)$ is generated as follows:

$$\Psi_l[m] = \begin{cases} G[\max(\psi_l[m], m + \rho_l[m])] \\ \qquad\qquad (m = 0, \cdots, M-2) \\ \tilde{K} - 1 \qquad (m = M-1) \end{cases} \tag{9}$$

where, table $G[k]$ lists the significant element index corresponding to the k-th element. max() returns the greater of the given values. $\psi_l[m]$ is the minimum element index for significant elements belonging to interval $[m, m - M + K]$. In the example of Fig. 4, $\psi_l[m]$ indicates the closest significant node to the green broken line in the right side of the green broken line. In other words, $\psi_l[m]$ indicates the significant element which is the closest to the inferior end in the range for the boundary L_m of the m-th bin. $\rho_l[m]$ is the maximum number of consecutive insignificant elements in interval $[0, m]$, and is generated according to the process shown in Fig. 5. In the example of Fig. 4, the element index $m + \rho_l[m]$ in the above equation can be interpreted as follows. $\rho_l[m]$ is the maximum number of consecutive insignificant elements in the left side of the green broken line and on itself, that is, $\rho_l[0] = 0$, $\rho_l[1] = \rho_l[2] = 1$. m corresponds to the element index of the node on the green broken line. Thus, the element index $m+\rho_l[m]$ indicates the leftmost node among candidates whose transition source are limited to significant nodes.

The rightmost node within the search range in the m-th row is identified as the maximum significant element index of the boundary of m-th bin, which is referred as the upper limit of the m-th bin, hereinafter. For the upper limit of the m-th bin, table $\Psi_u[m]$ $(m = 0, \cdots, M-1)$ is prepared. $\Psi_u[m]$ $(m = 0, \cdots, M-1)$ is generated as follows:

$$\begin{aligned} \Psi_u[m] = G[\min(&\psi_u[m - M + K], \\ &m - M + K - \rho_u[m])] \\ &(m = 0, \cdots, M-1) \end{aligned} \tag{10}$$

where min() returns the smaller of the given values. $\psi_u[m - M + K]$ is the maximum element index for significant elements belonging to interval $[m, m-M+K]$ of the histogram.

```
1.  if h[0] ≠ 0
2.      c ← 0
3.      ρ_l[0] ← 0
4.  else
5.      c ← 1
6.      ρ_l[0] ← 1
7.  for m = 1, ··· , M − 1
8.      if h[m] ≠ 0
9.          c ← 0
10.     else
11.         c + +
12.     ρ_l[m] ← max(ρ_l[m − 1], c)
```

Fig. 5 Generation algorithm for look-up-table $\rho_l[m]$.

```
1.  if h[K − 1] ≠ 0
2.      c ← 0
3.      ρ_u[M − 1] ← 0
4.  else
5.      c ← 1
6.      ρ_u[M − 1] ← 1
7.  for m = M − 2, ··· , 0
8.      if h[m − M + K] ≠ 0
9.          c ← 0
10.     else
11.         c + +
12.     ρ_u[m] ← max(ρ_u[m + 1], c)
```

Fig. 6 Generation algorithm for look-up-table $\rho_u[m]$.

In the example of Fig. 4, $\psi_u[m - M + K]$ corresponds to the closest significant node to the blue broken line in the left side of the blue broken line. In other words, $\psi_u[m - M + K]$ indicates the significant element which is the closest to the superior end in the range for the boundary L_m of the m-th bin. $\rho_u[m]$ is the maximum number of consecutive insignificant elements in interval $[m - M + K, K - 1]$, and is generated according to the process shown in Fig. 6. In the example of Fig. 4, the element index $m - M + K - \rho_u[m]$ in the above equation can be interpreted as follows. $\rho_u[m]$ is the maximum number of consecutive insignificant elements in the right side of the blue broken line and on itself, that is, $\rho_u[0] = \rho_u[1] = 2$, $\rho_u[2] = 1$. $m - M + K$ corresponds to the element index of the node on the blue broken line. Thus, the element index $m - M + K - \rho_u[m]$ indicates the rightmost node among candidates whose transition destinations are limited to significant nodes.

5.3 Optimal Quantizer Design Considering Sparseness

In this subsection, we describe our algorithm of sparse DP quantization that reduces the complexity by skipping computation for insignificant elements while retaining optimality in terms of minimizing quantization error. In the following, we use table $F[\tilde{k}]$ that lists the element index corresponding to the $\tilde{k}$-th significant element.

Let us consider dividing histogram interval $[0, F[\tilde{L}_m]]$, whose boundary is the $\tilde{L}_m$-th significant element, into $m + 1$ bins. The sub-interval $[F[\tilde{L}_i - (\tilde{\Delta}_i - 1)], F[\tilde{L}_i]]$ (where,

1. Load histogram $h[k]$ $(k = 0, \cdots, K-1)$ of the signal values
2. Load the number of bin M
3. Generate look-up-tables $G[k]$ $(k = 0, \cdots, K-1)$, $F[\tilde{k}]$ $(\tilde{k} = 0, \cdots, \tilde{K}-1)$, $\Psi_l[m]$, $\Psi_u[m]$ $(m = 0, \cdots, M-1)$
4. Generate look-up-table $E[\tilde{i}_e - \tilde{i}_s + 1, \tilde{i}_e]$ $(\tilde{i}_s \le \tilde{i}_e, \tilde{i}_e = 0, \cdots, \tilde{K}-1, \tilde{i}_s = 0, \cdots, \tilde{K}-1)$ for quantization error of each interval in histogram $h[k]$
5. for $j = 0, \cdots, G[K-M-1]$
6. $\quad S_0[j] \leftarrow E[0, j]$
7. for $m = 1, \cdots, M-1$
8. $\quad$ for $\tilde{L}_m = \Psi_l[m], \cdots, \Psi_u[m]$
9. $\quad\quad \tilde{S}_m[\tilde{L}_m] \leftarrow \min_{\tilde{\Delta}_m = 1, \cdots, \tilde{L}_m - \Psi_l[m-1]} \left[\tilde{S}_{m-1}[\tilde{L}_m - \tilde{\Delta}_m] + E[\tilde{L}_m - (\tilde{\Delta}_m - 1), \tilde{L}_m]\right]$
10. $\quad\quad \tilde{\Delta}_m^{(\tilde{L}_m)} \leftarrow \arg\min_{\tilde{\Delta}_m = 1, \cdots, \tilde{L}_m - \Psi_l[m-1]} \left[\tilde{S}_{m-1}[\tilde{L}_m - \tilde{\Delta}_m] + E[\tilde{L}_m - (\tilde{\Delta}_m - 1), \tilde{L}_m]\right]$
11. $\quad\quad T_{m-1}[\tilde{L}_m] \leftarrow \tilde{L}_m - \tilde{\Delta}_m^{(\tilde{L}_m)}$
12. $\tilde{L}_{M-1}^* \leftarrow \tilde{K} - 1$
13. for $m = M-1, \cdots, 1$
14. $\quad \tilde{L}_{m-1}^* \leftarrow T_{m-1}[\tilde{L}_m^*]$
15. $\quad \Delta_m^* \leftarrow F[\tilde{L}_m^*] - F[\tilde{L}_{m-1}^*]$
16. $\Delta_0^* \leftarrow F[\tilde{L}_0^*] + 1$

Fig. 7 Sparse DP quantization algorithm.

$i = 0, \cdots, m$) of the interval $[0, F[\tilde{L}_m]]$ is the i-th bin. $\tilde{\Delta}_i$ is the number of significant elements in the i-th bin, and $\tilde{L}_i$ is the significant element index of the boundary of the i-th bin, in other words, $\tilde{L}_i$ is the maximum significant index among significant elements in the i-th bin. We compute quantization error $e(F[\tilde{L}_i] - F[\tilde{L}_i - (\tilde{\Delta}_i - 1)] + 1, F[\tilde{L}_i])$ caused by using the centroid value to approximate all values in the i-th bin. This error is stored in look-up table $E[\tilde{\Delta}_i, \tilde{L}_i]$ and refer to the entries as needed, in order to eliminate duplicate computations for quantization error. We define a look-up table to store the minimum summation of quantization error $\sum_{i=0}^{m} E[\tilde{\Delta}_i, \tilde{L}_i]$ as $\tilde{S}_m[\tilde{L}_m]$. Note that $\tilde{S}_m[\tilde{L}_m]$ is equal to $S_m[F[\tilde{L}_m]]$.

Since $E[\tilde{\Delta}_m, \tilde{L}_m]$ depends on the significant element index $\tilde{L}_m$ of the boundary of the m-th bin and the number of significant elements $\tilde{\Delta}_m$ in the m-th bin, the value stored in $\tilde{S}_m[\tilde{L}_m]$ is computed using $\tilde{S}_{m-1}[\tilde{L}_m - \tilde{\Delta}_m]$ as follows:

$$\tilde{S}_m[\tilde{L}_m] = \min_{\tilde{\Delta}_m} \left\{\tilde{S}_{m-1}[\tilde{L}_m - \tilde{\Delta}_m] + E[\tilde{\Delta}_m, \tilde{L}_m]\right\} \tag{11}$$

where $m = 1, \cdots, M-1$. Using recursive equation (11), the computation of $\tilde{S}_m[\tilde{L}_m]$ results in the selection of the optimal parameter $\tilde{\Delta}_m$ among $1, \cdots, \tilde{L}_m - \Psi_l[m-1]$. The range of $\tilde{\Delta}_m$ can be found in Appendix B. Considering that the upper limit and the lower limit of significant indices in the m-th bin are defined as $\Psi_u[m]$ and $\Psi_l[m]$ respectively, $\tilde{L}_m$ can be taken from $\Psi_l[m]$ to $\Psi_u[m]$. The value stored in $\tilde{S}_m[\tilde{L}_m]$ is used in computing $\tilde{S}_{m+1}[\tilde{L}_{m+1}]$. In the case of $m = 0$, $\tilde{S}_0[\tilde{L}_0]$ represents the quantization error caused by using the centroid to approximate histogram interval $[0, F[\tilde{L}_0]]$, and we obtain:

$$\tilde{S}_0[\tilde{L}_0] = E[0, F[\tilde{L}_0]]$$

The optimal boundary of the $m-1$-th bin, which is next to the m-th bin with boundary $\tilde{L}_m$ $(= \Psi_l[m], \cdots, \Psi_u[m])$, is stored in table $T_{m-1}[\tilde{L}_m]$ as follows:

$$T_{m-1}[\tilde{L}_m] = \tilde{L}_m - \tilde{\Delta}_m^{(\tilde{L}_m)}$$

where $\tilde{\Delta}_m^{(\tilde{L}_m)}$ denote $\tilde{\Delta}_m$ which minimizes the right side of Eq. (11). Casting the above processes in pseudo-code yields instructions 4 to 11 in Fig. 7. Instruction 3 in Fig. 7 generates the look-up tables $\Psi_l[]$, $\Psi_u[]$ described in Sect. 5.2 and $G[], F[]$. Instruction 4 generates a look-up-table that stores the quantization error of every interval in the histogram. The stored value in $E[\tilde{i}_e - \tilde{i}_s + 1, \tilde{i}_e]$ $(\tilde{i}_s \le \tilde{i}_e, \tilde{i}_e = 0, \cdots, \tilde{K}-1, \tilde{i}_s = 0, \cdots, \tilde{K}-1)$ is the quantization error of the interval $[F[\tilde{i}_s], F[\tilde{i}_e]]$ in the histogram. $\tilde{i}_s$ and $\tilde{i}_e$ are significant element indices that identify an interval in the histogram. The minimization problem of Eq. (5) is rewritten as the following recursive formulation:

$$\min_{\tilde{\Delta}_{M-1}} \left\{\tilde{S}_{M-2}[\tilde{L}_{M-1} - \tilde{\Delta}_{M-1}] + E[\tilde{\Delta}_{M-1}, \tilde{L}_{M-1}]\right\}$$

In the final step at the instruction 10 in Fig. 7, we obtain $\tilde{\Delta}_{M-1}^{(\tilde{L}_{M-1})}$ as follows:

$$\tilde{\Delta}_{M-1}^{(\tilde{L}_{M-1})} = \arg\min_{\tilde{\Delta}_{M-1}} \left\{\tilde{S}_{M-2}[\tilde{L}_{M-1} - \tilde{\Delta}_{M-1}] + E[\tilde{\Delta}_{M-1}, \tilde{L}_{M-1}]\right\}$$

The optimal parameters $(\Delta_0^*, \cdots, \Delta_{M-1}^*)$ are obtained from the following process, the back-track process. Since the possible value of $\tilde{L}_{M-1}$ is limited to $\tilde{K}-1$, as the optimal value of $\tilde{L}_{M-1}$, we have $\tilde{L}_{M-1}^* = \tilde{K} - 1$. By using $\tilde{L}_{M-1}^* = \tilde{K} - 1$ and referring to table $T_{M-2}[]$, we obtain $\tilde{L}_{M-2}^* = T_{M-2}[\tilde{L}_{M-1}^*]$. Similarly, we obtain $\tilde{L}_{M-3}^* = T_{M-3}[\tilde{L}_{M-2}^*], \cdots, \tilde{L}_0^* = T_0[\tilde{L}_1^*]$ as the significant element indices of the boundary of each bin. By accessing $F[]$ with these obtained significant element indices $\tilde{L}_{M-1}^*$, $\tilde{L}_{M-2}^*, \cdots, \tilde{L}_0^*$, we get the element indices of the boundary of each bin. As a result, the intervals of each bin are derived as follows: $\Delta_{M-1}^* = F[\tilde{L}_{M-1}^*] - F[\tilde{L}_{M-2}^*]$, $\Delta_{M-2}^* = F[\tilde{L}_{M-2}^*] - F[\tilde{L}_{M-3}^*], \cdots, \Delta_1^* = F[\tilde{L}_1^*] - F[\tilde{L}_0^*]$, $\Delta_0^* = F[\tilde{L}_0^*] + 1$. Casting the above-mentioned processes in pseudo-code yields instructions 12 to 16 in Fig. 7.

6. Experiments

We performed the following experiments in order to investigate the effectiveness of our quantization algorithm from the viewpoint of complexity. As the input signal of each quantization algorithm, we used the sequences in *ITE/ARIB Ultra-high definition/wide-color-gamut standard test sequences - Series A, Series B* [19], [20]†. The sequences employ the progressive scan format with resolution of 3840×2160 pixels/frame in the RGB4:4:4 color format defined as ITU-R Recommendation BT.2020. The sequences are provided as still serial number files in uncompressed DPX format [21]. Pixel values of RGB components in the DPX file are treated as 16-bit integers. Since the actual pixel value only has 12 bit depth, it is stored in the higher 12 bits of the 16-bit integer and the remaining 4 bits are set to 0. In other words, these signals were sampled at 12 bit scale, so $K = 4096$. By extracting the higher 12 bits of each color component for every pixel, we obtained the evaluation data. Each color channel signals of the 61th frame†† of each sequence were used in the following evaluation experiments. Given the existence of legacy displays, it is often necessary to convert high bit depth signals into low bit depth signals that have just ten or eight bits/channel. Accordingly, we set $M = 1024, 256$ as the number of bins. Additionally, we also investigated the cases of $M = 512, 128$ for considering the characteristic of the proposed algorithm due to change in the number of bins. These experiments were performed on a computer with CPU:Intel core i7 (2.8 GHz) and memory: 8 GB.

In order to evaluate the complexity reduction achieved by sparse DP quantization, we compared sparse DP quantization (abbreviated to SDP-Q) with DP quantization (abbreviated to DP-Q) described in Sect. 3, in terms of processing time. The results are shown as bar graphs in Fig. 8, where processing time is the average of 100 trials. Additionally, we evaluated the complexity reduction attained by SDP-Q using the following metric:

$$\text{complexity reduction ratio} = \frac{\text{processing time of DP-Q} - \text{processing time of SDP-Q}}{\text{processing time of DP-Q}} \quad (12)$$

The line graphs in Fig. 8 show the complexity reduction ratio for each image. From Table 2 and Fig. 8, we can confirm that the complexity reduction ratio improves as sparseness increases. In order to elucidate the overall performance for all images, Table 3 shows average DP-Q and SDP-Q processing times for all images at every M value. From this table, we can confirm that sparse DP quantization can, relative to DP quantization, reduce complexity by 82.9 to 84.2% on average. Additionally, Table 4 shows the breakdown of the processing time of SDP-Q; look-up table (LUT) generation processes corresponding to instruction 3 and 4 in Fig. 7, and search processes corresponding to the other instructions in Fig. 7. It is observed that "search processes" has strong effect in the total processing time.

In order to evaluate computing time reduced by SDP-Q with different M values, we applied analysis of variance (ANOVA) to complexity reduction ratio with M. Table 5 shows the results of ANOVA test for complexity reduction ratio. In this case, the critical value at the 5% significance level is found 2.6955 from the F-distribution table. It was observed that F-ratio values for every color channel in Table 5 were less than the critical value. Thus, we could not reject the null hypothesis that four kinds of M values produce the same expected values of complexity reduction ratio. That is, we could not obtain statistical evidence that there was significant difference among the expected values of complexity reduction ratio with four kinds of M values.

Next, we analyze the complexity of the search processes and the LUT generation processes. The search processes conduct the following P1. As a dominant factor in the generation of look-up tables, let us focus on that of look-up table $E[]$ which conducts the following P2.

(P1) optimal path search based on DP recursive equation
(P2) construction of look-up table $E[]$ to store quantization error for each quantized bin

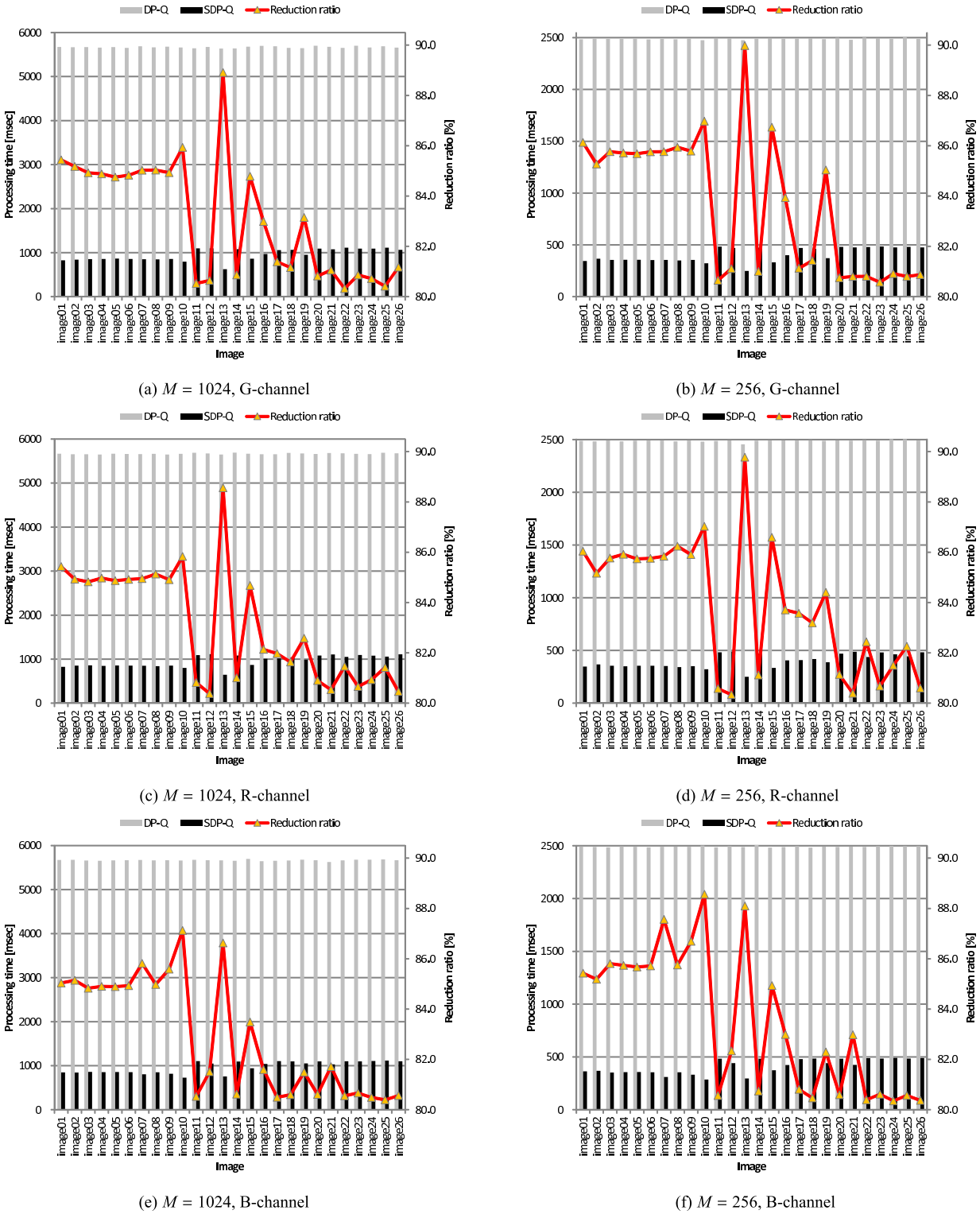
The complexity of "P1" is related to the number of feasible paths within the search range. In the following, feasible path within the search range is abbreviated to candidate path. The number of candidate paths for DP-Q is derived from K and M as follows:

$$\Omega_{\text{path}}(M, K) = \frac{1}{2}M^3 - \frac{2K+5}{2}M^2 + \frac{K^2+7K+4}{2}M - K^2 - K \quad (13)$$

The derivation of the above equation can be found in Appendix C. The figures in column "DP-Q" in Table 6 were generated based on the above equation. By contrast, the number of candidate paths for SDP-Q could not be derived like DP-Q due to the restriction of search range described in Sect. 5.2. So, we counted up candidate paths for each image for every M, and computed the average number of all images for every M value. These average numbers are shown as the figures in column "SDP-Q" in Table 6. The figures in column "reduction ratio" in Table 6 are computed by applying the same concept as Eq. (12) to the candidate paths of DP-Q and SDP-Q. We evaluated the number of candidate paths reduced by SDP-Q through applying ANOVA to its reduction ratio with different M values. Table 7 shows the results of the above-mentioned ANOVA test. It was observed that F-ratio values for every color channel in Table 7 were less than the critical value (2.6955) at the 5% significance level. Thus, we could not reject the null hypothesis that four kinds of M values produce the same expected values of the above-mentioned reduction ratio.

†Only Japanese manuals are available now. The Web site of the ITE says that English version is being made at present.

††The 61th frame is the head frame that contains captured scenes. The first 60 frames capture caption telop only.



(a) $M = 1024$, G-channel

(b) $M = 256$, G-channel

(c) $M = 1024$, R-channel

(d) $M = 256$, R-channel

(e) $M = 1024$, B-channel

(f) $M = 256$, B-channel

Fig. 8 Processing time of DP-Q and SDP-Q (bar graphs labeled "DP-Q" are processing times for DP quantization, bar graphs labeled "SDP-Q" are processing times for SDP quantization, and "Reduction ratio" are values defined in Eq. (12)).

The complexity of "P2" is related to the number of intervals in the input histogram. In the following, the quantized bin is abbreviated to candidate interval. The number of candidate intervals for DP-Q is derived from K and M as follows:

$$\Omega_{\text{interval}}(M, K) = (-M^2 + M + K^2 + K)/2 \qquad (14)$$

The derivation of the above equation can be found in Ap-

Table 3 Average processing time of DP-Q and SDP-Q of all images in every M (cells in the "Reduction ratio" column represent values defined in Eq. (12)).

(a) G-channel

M	DP-Q [msec]	SDP-Q [msec]	reduction ratio [%]
1024	5667.5	957.6	83.1
512	4000.0	662.6	83.4
256	2490.0	406.8	83.7
128	1546.1	248.2	83.9

(b) R-channel

M	DP-Q [msec]	SDP-Q [msec]	reduction ratio [%]
1024	5666.0	957.0	83.1
512	3996.2	657.9	83.5
256	2488.8	400.5	83.9
128	1546.6	244.1	84.2

(c) B-channel

M	DP-Q [msec]	SDP-Q [msec]	reduction ratio [%]
1024	5665.7	969.3	82.9
512	4001.3	670.2	83.2
256	2491.6	410.9	83.5
128	1548.7	252.1	83.7

Table 4 The breakdown of processing time of SDPQ ("LUT generation processes consist of instruction 3 and 4 in Fig. 7, and "search processes" consist of the other instructions in Fig. 7).

(a) G-channel

M	LUT generation processes	search processes
1024	43.6	914.1
512	71.5	591.1
256	77.4	329.4
128	77.8	170.4

(b) R-channel

M	LUT generation processes	search processes
1024	42.3	914.7
512	70.4	587.5
256	76.0	324.5
128	76.8	167.3

(c) B-channel

M	LUT generation processes	search processes
1024	46.0	923.3
512	73.5	596.7
256	79.5	331.4
128	79.8	172.3

pendix D. The figures in column "DP-Q" in Table 8 were generated based on the above equation. By contrast, candidate intervals for SDP-Q were counted for each image for every M. The average number of candidate intervals for every M value was computed. These average numbers are shown as the figures in column "SDP-Q" in Table 8.

We evaluated the number of candidate intervals reduced by SDP-Q through applying ANOVA to its reduction ratio with different M values. Table 9 shows the results of the above-mentioned ANOVA test. It was observed that F-ratio values for every color channel in Table 9 exceed the critical value (2.6955) at the 5% significance level. Thus, the ANOVA test suggested that there was significant differences among the expected values of the above-mentioned

Table 5 ANOVA test results for complexity reduction ratio of processing time (df, SS and MS mean Degrees of Freedom, Sums of Squares, Mean Squares, respectively).

(a) G-channel

Source of Variation	df	SS	MS	F-ratio
Between Treatments	3	10.14	3.38	0.48
Error	100	700.39	7.00	–
Total	103	710.53	–	–

(b) R-channel

Source of Variation	df	SS	MS	F-ratio
Between Treatments	3	17.95	5.98	0.96
Error	100	625.55	6.26	–
Total	103	643.50	–	–

(c) B-channel

Source of Variation	df	SS	MS	F-ratio
Between Treatments	3	10.01	3.34	0.46
Error	100	726.58	7.27	–
Total	103	736.59	–	–

Table 6 Average number of candidate paths in every M.

(a) G-channel

M	DP-Q	SDP-Q	reduction ratio [%]
1024	4827117568	808879659	83.2
512	3278238720	532953038	83.7
256	1874162176	298821085	84.1
128	992694528	155958701	84.8

(b) R-channel

M	DP-Q	SDP-Q	reduction ratio [%]
1024	4827117568	808782460	83.2
512	3278238720	528395833	83.9
256	1874162176	293897544	84.3
128	992694528	153101171	84.6

(c) B-channel

M	DP-Q	SDP-Q	reduction ratio [%]
1024	4827117568	818392732	83.0
512	3278238720	538483520	83.6
256	1874162176	301716348	83.9
128	992694528	157991606	84.1

Table 7 ANOVA test results for reduction ratio of the number of candidate paths (df, SS and MS mean Degrees of Freedom, Sums of Squares, Mean Squares, respectively).

(a) G-channel

Source of Variation	df	SS	MS	F-ratio
Between Treatments	3	15.97	5.32	0.76
Error	100	696.01	6.96	–
Total	103	711.98	–	–

(b) R-channel

Source of Variation	df	SS	MS	F-ratio
Between Treatments	3	26.48	8.83	1.40
Error	100	628.40	6.28	–
Total	103	654.88	–	–

(c) B-channel

Source of Variation	df	SS	MS	F-ratio
Between Treatments	3	16.19	5.40	0.77
Error	100	702.54	7.03	–
Total	103	718.73	–	–

reduction ratio with four kinds of M values.

Let us consider how M value affects the reduction ratio of the number of candidate intervals. Although we cannot derive the number of candidate intervals for SDP-

Table 8 Average number of candidate intervals in every M.

(a) G-channel

M	DP-Q	SDP-Q	reduction ratio [%]
1024	7866880	757177	90.4
512	8259840	1150137	86.1
256	8358016	1248313	85.1
128	8382528	1272825	84.8

(b) R-channel

M	DP-Q	SDP-Q	reduction ratio [%]
1024	7866880	743271	90.6
512	8259840	1136231	86.2
256	8358016	1234407	85.2
128	8382528	1258919	85.0

(c) B-channel

M	DP-Q	SDP-Q	reduction ratio [%]
1024	7866880	792665	89.9
512	8259840	1185625	85.6
256	8358016	1283801	84.6
128	8382528	1308313	84.4

Table 9 ANOVA test results for reduction ratio of the number of candidate intervals (df, SS and MS mean Degrees of Freedom, Sums of Squares, Mean Squares, respectively).

(a) G-channel

Source of Variation	df	SS	MS	F-ratio
Between Treatments	3	521.74	173.91	22.75
Error	100	764.31	7.64	–
Total	103	1286.05	–	–

(b) R-channel

Source of Variation	df	SS	MS	F-ratio
Between Treatments	3	523.78	174.59	24.03
Error	100	726.47	7.26	–
Total	103	1250.25	–	–

(c) B-channel

Source of Variation	df	SS	MS	F-ratio
Between Treatments	3	516.54	172.18	18.50
Error	100	930.58	9.31	–
Total	103	1447.12	–	–

Q analytically, the number can be roughly estimated as $\Omega_{\text{interval}}(M, \tilde{K})$. Note that the estimation $\Omega_{\text{interval}}(M, \tilde{K})$ does not take account of the restriction of search range described in Sect. 5.2. Using the above estimation, the reduction ratio of the number of candidate intervals is approximated as follows:

$$\frac{\Omega_{\text{interval}}(M, K) - \Omega_{\text{interval}}(M, \tilde{K})}{\Omega_{\text{interval}}(M, K)} = \frac{K^2 + K - \tilde{K}^2 - \tilde{K}}{-M^2 + M + K^2 + K} \tag{15}$$

Here, the numerator of the above equation is independent from M. The denominator of the above equation monotonically decreases as M increases, in the range of $1 \le M$. Thus, it is expected that Eq. (15) monotonically increases in the range of $1 \le M$. This expectation agrees with the observed results that are shown in column "reduction ratio" in Table 8.

7. Conclusions

This paper studied the complexity reduction possible with DP quantization which focuses on the sparseness of signal values. The proposed method, which is called sparse DP quantization, keeps the optimality of DP quantization in terms of minimizing quantization error. Specifically, sparse DP quantization can reduce the complexity of DP quantization without increasing quantization error. Experiments showed that sparse DP quantization can achieve 82.9 to 84.2% complexity reduction, on average, compared to DP quantization.

Sparse DP quantization can be used as a complementary approach to conventional methods [10], [11] for complexity reduction of DP quantization, since the conventional methods take approaches that are independent of signal value sparseness. Therefore, by combining sparse DP quantization and conventional methods, the complexity of DP quantization can be reduced further.

Let us mention future works on the family of DP quantization technologies from the following two aspects. Firstly, an important future work is an extension for HDR image format. There are growing expectations for HDR imaging in many areas. Many HDR imaging applications use floating-point data. By contrast, the family of DP quantization technologies are designed on the premise that inputs are discrete signal formatted as integer data, as described in this paper. So, it is beyond the scope of this paper to apply the family of DP quantization technologies to floating-point data such as HDR image format at this time. But, we would like to discuss such extension in a future paper. Secondly, another important future work is an extensions for video sequences. Although sparse DP quantization significantly reduced the computational complexity over the existing DP quantization, it is still costly operation to design optimal quantizer for every frame in video sequences. So, it is worth to extend sparse DP quantization from the view point of complexity reduction based on temporal correlation among a video sequence. We would like to study such extension as a future work.

References

[1] R.M. Gray and D.L. Neuhoff, "Quantization," IEEE Trans. Inf. Theory, vol.44, no.6, pp.2325–2383, Oct. 1998.
[2] E. Reinhard, G. Ward, S. Pattanaik, P. Debevec, W. Heidrich, and K. Myszkowski, High Dynamic Range Imaging: Acquisition, Display, and Image-Based Lighting, Morgan Kaufmann Publisher, 2010.
[3] E. François, D. Rusanovskyy, P. Yin, P. Topiwala, G. Sullivan, and M. Naccari, "Signalling, backward compatibility and display adaptation for HDR/WCG video coding," draft 1. ISO/IEC JTC1/SC29/WG11/N16508, Oct. 2016.
[4] J. Boyce, Y. Ye, J. Chen, and A. Ramasubramonian, "Overview of SHVC: Scalable extensions of the high efficiency video coding standard," IEEE Trans. Circuits Syst. Video Technol., vol.26, no.1, pp.20–34, 2016.
[5] ISO/IEC 18477-2:2016: Information technology: Scalable compression and coding of continuous-tone still images – Part 2: Coding

of high dynamic range images, 2016.
[6] ISO/IEC PDTR 23008-14: Information technology — High efficiency coding and media delivery in heterogeneous environments — Part 14: Conversion and Coding Practices for HDR/WCG Y'CbCr 4:2:0 Video with PQ Transfer Characteristics, 2016.
[7] S.P. Lloyd, "Least squares quantization in PCM," IEEE Trans. Inf. Theory, vol.IT-28, no.2, pp.129–136, March 1982.
[8] J. Max, "Quantizing for minimum distortion," IRE Trans. Inf. Theory, vol.IT-7, no.1, pp.7–12, March 1960.
[9] J.D. Bruce, Optimum quantizer, Ph.D. thesis, M.I.T., May 1964.
[10] D. Sharma, "Design of absolutely optimal quantizers for a wide class of distortion measures," IEEE Trans. Inf. Theory, vol.24, no.6, pp.693–702, Nov. 1978.
[11] X. Wu, "Optimal quantization by matrix searching," J. Algorithms, vol.12, no.4, pp.663–673, Dec. 1991.
[12] T. Richter, A. Artusi, and T. Ebrahimi, "JPEG XT: A new family of JPEG backward-compatible standards," IEEE Multimedia Mag., vol.23, no.3, pp.80–88, 2016.
[13] ITU-T and ISO/IEC JTC 1. High efficiency video coding, ITU-T Rec.H.265 and ISO/IEC 23008-2(HEVC), 2016.
[14] P. Ferreira and A.J. Pinho, "Why does histgram packing improve lossless compression rates?," IEEE Signal Process. Lett., vol.9, no.8, pp.259–261, 2002.
[15] M. Aguzzi and M. Albanesi, "A novel approach to sparse histogram image lossless compression using JPEG 2000," Electronic Letters on Computer Vision and Image Analysis, vol.5, no.4, pp.24–46, 2006.
[16] E. Nasr-Esfahani, S. Samavi, N. Karimi, and S. Shiran, "Near lossless image compression by local packing of histogram," Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing, pp.1197–1200, 2008.
[17] Y. Bandoh, S. Takamura, and A. Shimizu, "Complexity reduction for dynamic programming based quantization considering sparseness of signal value," Proc. Forum on Information Technology, 2015 (in Japanese).
[18] Y. Bandoh, S. Takamura, and A. Shimizu, "Complexity reduction algorithm for optimum quantizer design based on amplitude sparseness," Proc. IEEE Int. Conf. Acoust., Speech, Signal Process., IVMSP-P10.6, 2018.
[19] http://www.ite.or.jp/contents/chart/uhdtv_a/manual.pdf (in Japanese).
[20] http://www.ite.or.jp/contents/chart/uhdtv_b/UHDTV_Bseries_manual.pdf (in Japanese).
[21] SMPTE ST 268:2014, file format for digital moving-picture exchange (DPX). Nov. 2014.

Appendix A: Range of Δ_m

Let us consider the range of Δ_m. From Eq. (2), we have

$$m-1 \le L_{m-1} \le K-(M-m+1)$$

From the above equation and $L_{m-1} = L_m - \Delta_m$, we get the range of Δ_m in the following:

$$L_m - K + (M-m+1) \le \Delta_m \le L_m - m + 1 \qquad (A\cdot 1)$$

Furthermore, considering $\Delta_m \ge 1$, we obtain

$$1 \le \Delta_m \le L_m - m + 1 \qquad (A\cdot 2)$$

Appendix B: Range of $\tilde{\Delta}_m$

Let us consider the range of $\tilde{\Delta}_m$. From $\tilde{L}_{m-1} = \tilde{L}_m - \tilde{\Delta}_m$, the range of $\tilde{L}_m - \tilde{\Delta}_m$ becomes

$$\Psi_l[m-1] \le \tilde{L}_m - \tilde{\Delta}_m \le \Psi_u[m-1]$$

Using the above relationship, the range of $\tilde{\Delta}_m$ becomes

$$\tilde{L}_m - \Psi_u[m-1] \le \tilde{\Delta}_m \le L_m - \Psi_l[m-1] \qquad (A\cdot 3)$$

Furthermore, considering $\tilde{\Delta}_m \ge 1$, we obtain

$$1 \le \tilde{\Delta}_m \le \tilde{L}_m - \Psi_l[m-1] \qquad (A\cdot 4)$$

Appendix C: Derivation of Eq. (13)

The paths in a trellis transition diagram are divided into three classes as follows:
(i) nodes in a range with $m = 0$
(ii) nodes in a range with $m = 1, \cdots, M-2$
(iii) nodes in a range with $m = M-1$

Firstly, let us consider the class (i). In this class, there are $K-M+1$ kinds of nodes, and every node has single path. So, we find that there are $K-M+1$ kinds of paths.

Secondly, let us consider the class (ii). In this class, a node $(m+\ell, m)$ has $\ell+1$ kinds of paths. In a group of nodes which are located at $(m+\ell, m)$ where $\ell = 0, 1, \cdots, K-M$, $m = 1, \cdots, M-2$, there are following number of paths:

$$\sum_{m=1}^{M-2}\sum_{\ell=0}^{K-M}(\ell+1) = (M-2)\sum_{\ell=0}^{K-M}(\ell+1)$$

Thirdly, let us consider the class (iii). In this class, there is single node, and the node has $K-M+1$ kinds of paths.

From the above results, we obtain the following:

$$\begin{aligned}&\Omega_{\text{path}}(M,K)\\ &= (K-M+1) + (M-2)\sum_{\ell=0}^{K-M}(\ell+1) + (K-M+1)\\ &= \frac{1}{2}M^3 - \frac{2K+5}{2}M^2 + \frac{K^2+7K+4}{2}M - K^2 - K\end{aligned}$$

Appendix D: Derivation of Eq. (14)

The lower boundary of the interval are divided into two classes as follow:
(i) lower boundaries in a range with $k = 0, 1, \cdots, M-1$
(ii) lower boundaries in a range with $k = M, M+1, \cdots, K-1$

Firstly, let us consider the class (i). In this class, the allowable upper bound is $K-M+k$. There are $K-M+1$ kinds of intervals for every k, respectively. For example, when the lower boundary is $k = 0$, there are $K-M+1$ kinds of intervals; $[0,0]$, $[0,1]$, $\cdots$, $[0, K-M]$. So, we find the sum of the number of intervals as follows:

$$(K-M+1)M \qquad (A\cdot 5)$$

Secondly, let us consider the class (ii). In this class, the allowable upper bound is $K-1$. When the lower boundary

is k, there are $K-k$ kinds of intervals; $[k,k]$, $[k,k+1]$, $\cdots$, $[k,K-1]$. So, we find the sum of the number of intervals as follows:

$$\sum_{k=M}^{K-1}(K-k) \qquad (\mathrm{A}\cdot 6)$$

From Eqs. (A· 5) and (A· 6), we obtain the following:

$$\begin{aligned}&\Omega_{\mathrm{interval}}(M,K)\\ &= (K-M+1)M+\sum_{k=M}^{K-1}(K-k)\\ &= (-M^2+M+K^2+K)/2\end{aligned}$$

Yukihiro Bandoh received the B.E., M.E., and Ph.D. degrees from Kyushu University, Japan, in 1996, 1998 and 2002, respectively. He granted JSPS Research Fellowship for Young Scientists from 2000 to 2002. In 2002, he joined Nippon Telegraph and Telephone (NTT) Corporation, where he has been engaged in research on efficient video coding for high realistic communication. He is currently a senior research engineer, distinguished technical member at NTT Media intelligence Laboratories. Dr. Bandoh is a senior member of IEEE, IEICE, and IPSJ.

Seishi Takamura received the B.E., M.E. and Ph.D. degrees from Department of Electronic Engineering, Faculty of Engineering, the University of Tokyo in 1991, 1993 and 1996 respectively. In 1996 he joined Nippon Telegraph and Telephone (NTT) Corporation, where he has been engaged in research on efficient video coding and ultra-high quality video coding. He has fulfilled various duties in the research and academic community in current and prior roles including Associate Editor of IEEE Transactions on Circuits and Systems for Video Technology (2006–2014), Executive Committee Member of the Institute of Electrical and Electronics Engineers (IEEE) Tokyo Section. He has also served as Chair of ISO/IEC JTC 1/SC 29 Japan National Body, Japan Head of Delegates of ISO/IEC JTC 1/SC 29, and as an International Steering Committee Member of the Picture Coding Symposium. From 2005 to 2006, he was a Visiting Scientist at Stanford University, California, USA. He is currently a Senior Research Engineer, Supervisor, Senior Distinguished Engineer at NTT Media Intelligence Laboratories. Dr. Takamura is a senior member of IEEE, IEICE, and IPSJ and a member of MENSA, APSIPA, SID and ITE.

Atsushi Shimizu received the B.E. and M.E. degrees in electronic engineering from Nihon University in 1990 and 1992 respectively. He joined Nippon Telegraph and Telephone (NTT) Corporation in 1992 and has been engaged in video compression algorithm and software development. He is currently a Senior Research Engineer, Supervisor, and the head of visual media coding group in NTT Media intelligence Laboratories.