IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 20 September 2023, accepted 22 November 2023, date of publication 27 November 2023,
date of current version 30 November 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3336892

RESEARCH ARTICLE

Adaptive Patch-Wise Depth Range Linear Scaling Method for MPEG Immersive Video Coding

SUNG-GYUN LIM, HYUN-HO KIM, AND YONG-HWAN KIM
Korea Electronics Technology Institute, Seongnam-si 13488, Republic of Korea

Corresponding author: Sung-Gyun Lim (rbs293@keti.re.kr)

This work was supported by the Institute of Information and Communications Technology Planning and Evaluation (IITP) Grant funded by the Korea Government through Ministry of Science and ICT (MSIT) (Development of Ultra High-Resolution Unstructured Plenoptic Video Storage/Compression/Streaming Technology for Medium to Large Space) under Grant 2020-0-00920.

ABSTRACT The Moving Picture Experts Group (MPEG) is responsible for standardizing MPEG immersive video (MIV) for immersive video coding and is involved in research and development focusing on providing six degrees of freedom through a reference software known as the test model for immersive video. To efficiently compress and transmit multiview videos with texture and depth pairings, the encoder part of the MIV codec framework reduces the pixel rate by removing redundancy between views and densely packing the remaining regions into an atlas as patches. The decoder part reconstructs multiview videos from the transmitted atlas to synthesize and render arbitrary viewports, and the depth information has a significant impact on the quality of the rendered viewport. However, the existing method of handling depth values in the MIV codec fails to adequately address the information loss that occurs during quantization or transmission. To preserve and transmit depth information more accurately, we propose a method for expanding the depth dynamic range using min–max linear scaling on a patch-by-patch basis. In addition, we efficiently encode the per-patch minimum and maximum values of depth required by the decoder to recover the original depth values and include them in the metadata. The experimental results indicate that for computer-generated sequences, the proposed method provides PSNR-based Bjøntegaard delta-rate gains of 9.1% and 3.3% in the end-to-end performance for high- and low-bitrate cases, respectively. In addition, subjective quality improvements are observed by reducing the artifacts that primarily occur at the object boundaries in the rendered viewport.

INDEX TERMS Depth scaling, immersive video, MPEG immersive video, multiview video, virtual reality, 6DoF video.

I. INTRODUCTION

As interest in virtual reality has increased, related technologies have earned and continue to attract attention [1], [2], [3], [4]. In view of this, research on videos that provide users with higher degrees of freedom (DoFs) is actively being conducted. A 3DoF video, that is, a 360-degree video, allows the user to observe all directions of a scene from a fixed position. Because a 3DoF video only supports a visual experience limited to rotational movements from the user's point of view, the sense of immersion for the user is reduced. In contrast, a 6DoF video can provide a user with a superior sense of immersion by reflecting rotational movements as well as positional changes from the user's point of view.

In general, to provide motion parallax in 6DoF videos, three-dimensional (3D) rendering technology is required to synthesize arbitrary viewpoint videos by using multiview plus depth (MVD) videos [5]. Fig. 1 shows the camera placement for the Museum [6] sequence with 24 viewpoints and examples of two viewpoints from different locations. As shown in the figure, the MVD comprises pairs of videos acquired from multiple viewpoints and their corresponding depth maps. Therefore, the volume of data that need to be transmitted for 6DoF videos is considerably larger than that

The associate editor coordinating the review of this manuscript and approving it for publication was Lei Wei.

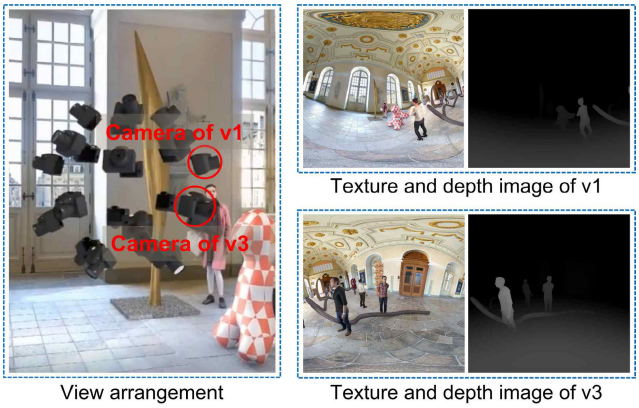

FIGURE 1. An example of a multiview camera array, and examples of input texture and depth view at specific locations (Museum sequence) [21].

for conventional 2D videos to achieve high-quality rendering, and a technology that can efficiently compress and transmit these data is essential.

The ISO/IEC Moving Picture Experts Group (MPEG) is actively working on the standardization of "Coded Representation of Immersive Media," which is referred to as MPEG-I [7]. The MPEG-I project includes a wide range of standards covering the overall structure, delivery format, audio and video compression technologies, and metadata for immersive media. In particular, part 12 of MPEG-I, called MPEG immersive video (MIV), is developing standard technologies for the efficient encoding and decoding of high-capacity 6DoF videos [8]. The MIV group published the Final Draft International Standard (FDIS) [9] in July 2021 and is currently developing its second version [10], which will introduce more expanded use cases [11]. The MIV group is working on improving the performance of the MIV codec framework by publishing a reference software called the test model for immersive video (TMIV) for the standard development, adoption, and integration of various element technologies [12].

The main concept of the MIV codec for efficiently compressing large 6DoF images involves eliminating the redundancy between multiple-input view videos [13], [14]. In TMIV, the redundant regions between the input view videos are removed, and the remaining regions are segmented into patches that are packed into a small number of atlases, which are then compressed using a conventional 2D codec. Because the input multiview videos have paired texture and depth images per viewpoint, the atlas is separately generated for texture and depth. In this case, the input depth-view videos have a bit depth of 16 bits, whereas the depth atlas has a bit depth of 10 bits. This can cause quantization errors in the depth information during the atlas generation process. In addition, coding errors in the depth atlas can cause errors in the depth information as it is decoded. Because depth information is highly important for virtual view synthesis, errors in depth information can significantly degrade the rendering quality of 6DoF videos.

One approach to solving these issues and improving the accuracy of the depth information sent to the decoder is to widen the dynamic range of the depth values. The results of various studies on depth representation methods have been adopted in the TMIV and MIV specifications because of their high performance. Furthermore, the studies have consistently verified their performance improvement effects on preserving the accuracy of depth information [15], [16], [17].

In this study, we propose an adaptive patch-wise depth linear scaling method for a highly accurate representation of depth information. While the current TMIV is based on scaling depth values on a view-by-view basis [12], the proposed method applies min–max linear scaling on a patch-by-patch basis to expand the dynamic range of depth values, thereby reducing depth information errors and ultimately improving the rendering quality. In addition, an algorithm is designed to adaptively apply depth-value scaling on a patch-by-patch basis to achieve optimal performance in terms of the Bjøntegaard delta (BD) [18] rate through various verification experiments. The proposed method represents the minimum and maximum depth information of each patch as efficiently as possible, which is then transmitted to the decoder such that

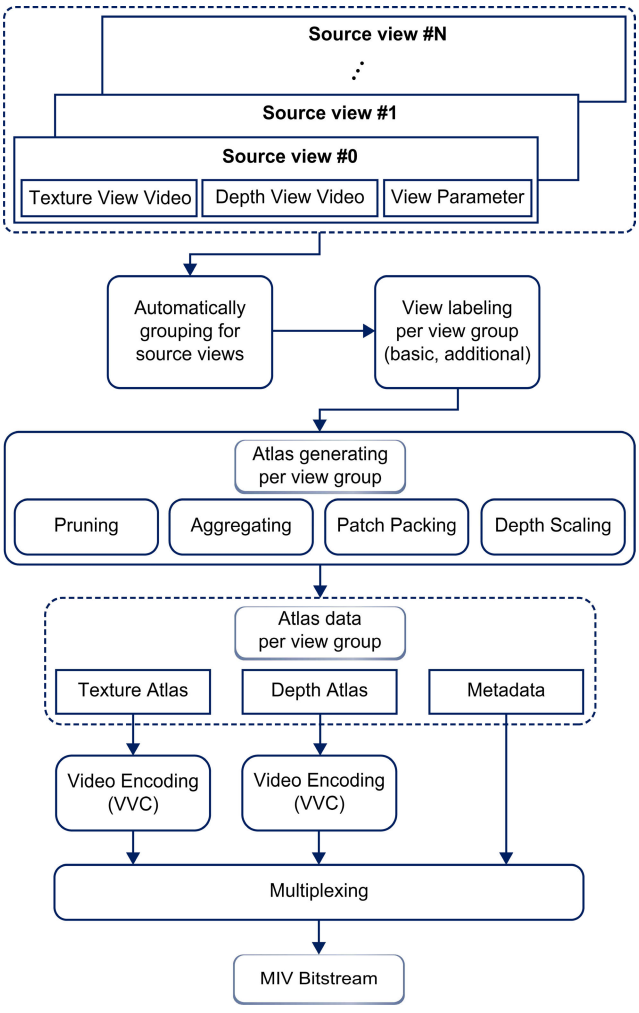

FIGURE 2. Overview of the architechture of the TMV group-based encoder.

the depth value can be inversely scaled per patch. This method provides significant compression efficiency to MPEG [19], and core experiments (CE) are being conducted for further its refinement and validation [20].

The remainder of this paper is organized as follows: Section II provides a general description of MIV encoding and related studies, followed by a detailed description of how depth values are managed in the TMIV encoder. Section III describes the algorithm used in the proposed method. Section IV presents the experimental results and performance analysis of the proposed method. Finally, Section VI summarizes the study findings and concludes the paper.

II. RELATED WORKS

This section describes the overall process of the encoder part of the MIV codec framework and presents the details of the depth range scaling algorithm, which is most closely related to the content of this study in TMIV.

A. MIV ENCODING PROCESS

The overarching structure of the TMIV encoder is illustrated in Fig. 2. Initially, the input multiview videos undergo an automated process wherein they are organized into a designated number of groups defined by the user. Subsequently, the view-labeling stage comes into play, categorizing each source view within a group as either basic or additional. Among these views, the basic views are chosen as a limited set of input views that encompass the largest portion of the entire scene. These basic views form the foundation for eliminating pixels that are replicated between different viewpoints during the pruning phase.

In the pruning phase, additional views' pixels containing information that is already present in the basic view are considered redundant and are removed. The positions of the basic views serve as projection targets for additional views. After pruning an additional view, the additional views can be pruned from both the basic and previously pruned additional views. Fig. 3 illustrates one of the source views in the Museum sequence and displays the valid (non-pruned) pixels that remain after the pruning process. The invalid (pruned) pixels are represented by the black pixels on the right-hand side of Fig. 3.

The pruning process operates at the frame level, with the valid region determined for each frame. Over time, the valid regions for all frames within an intra period are aggregated. The aggregated valid regions are subsequently grouped into distinct clusters, which are then combined to form rectangular patches. An example of this clustering process is shown in Fig. 4, which illustrates the transformation of a clustered valid region into a rectangular patch [12].

Subsequently, all patches are arranged into a series of atlases sequentially, starting with the largest patch. Throughout this arrangement process, each patch is meticulously fitted into an atlas to occupy the minimal conceivable space while still being confined within the valid region. The metadata accompanying these patches encompass various details,

FIGURE 3. Example of a source view being input as a TMIV (left), and the view being pruned (right) (view index = 0).

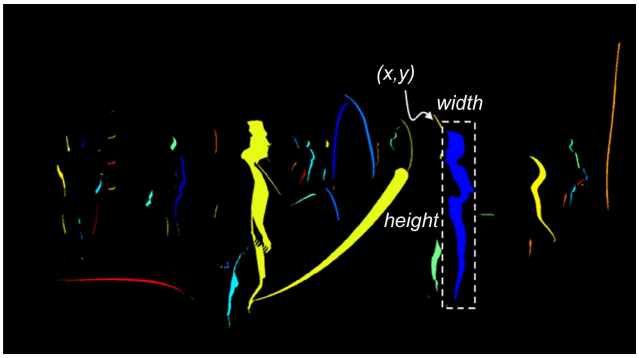

FIGURE 4. Example of an aggregated clusters and patch information in a pruned view [12].

such as the coordinates of the top-left corner, dimensions including width and height measurements, and rotation data necessary for the decoder to reconstruct the pruned views. The final step, as shown in Fig. 2, involves a combination of video bitstreams originating from the depth and texture atlases. This amalgamation, coupled with the metadata, constitutes the MIV output bitstream.

B. LINEAR SCALING OF DEPTH RANGE

In the current TMIV configuration, input depth views are typically generated into depth atlases by bit-depth downscaling on the encoder side. To minimize errors in the depth information during this downscaling process, the depth dynamic range per input view is maximized, wherein min–max linear scaling is applied to the depth values per input view [15].

Fig. 5 shows how the dynamic range of the depth values changes step-by-step in the conventional MIV codec. First, the depth range of each source view with a bit depth of b-bit is linearly scaled to the full range. Optionally, a nonlinear depth range scaling method [16] can be used instead of the linear scaling method. The full-range scaled depth range is mapped to $[2T, 2^n - 1]$ when generating an atlas video of n-bit depth.

The depth atlas does not use the entire range available for depth-value representation because it also embeds occupancy information in the depth atlas and transmits it to the decoder; unoccupied (invalid) pixels have a depth value of zero and occupied (valid) pixels have their depth values mapped in

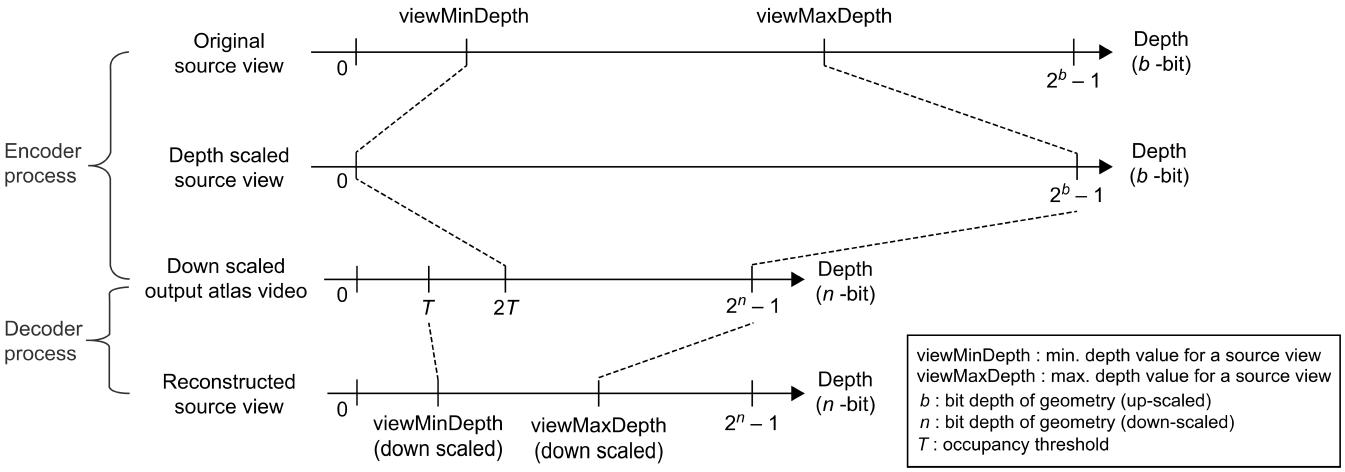

FIGURE 5. Conceptual diagram of the traditional view-by-view depth range scaling process.

the range $[2T, 2^n - 1]$. The TMIV decoder considers the encoding error in the depth atlas, determines whether pixels with depth values less than T are unoccupied and pixels with depth values greater than T are occupied, and restores the depth values in the range $[T, 2^n - 1]$ by remapping them between the minimum and maximum depth values of the view.

III. PROPOSED DEPTH RANGE SCALING METHOD

As described in the previous section, the existing TMIV maximizes the dynamic range of the depth values by applying min–max linear scaling to each input depth-perspective image. The proposed method further expands the dynamic range of the depth values by applying min–max linear scaling to the expanded depth values on a patch-by-patch basis. In other words, because the dynamic range of depth is generally narrower per patch than per view, it is more effective to apply depth range scaling per patch. When the depth value is adjusted by applying per-patch depth range scaling to the encoder, the decoder should be able to restore the original depth value. This section describes the application of per-patch adaptive depth scaling, and the encoding and transmission of the minimum and maximum depth values for each patch.

A. PATCH-WISE DEPTH RANGE SCALING

The red color in Fig. 6 indicates how the depth values are mapped per patch step-by-step when the proposed method is added to the traditional view-wise depth range scaling algorithm. To maximize the dynamic range per depth, the minimum and maximum depth values for each patch are determined, and the range is expanded f times. The depth linear mapping equation is defined as

$$d_{mapped} = f \times (d_{original} - \text{patchMinDepth}), \quad (1)$$

where $d_{original}$ is the original depth value in the scaled depth view and d_{mapped} is the mapped value with linear scaling from depth patches scaled per patch. Furthermore, f is the scale factor by which the depth dynamic range is expanded using linear depth scaling. The maximum value of f is limited to k to avoid overscaling the depth range of each patch. Expanding the range of depth values can improve the rendering quality by accurately representing the depth values; however, it can also reduce the coding performance because of the increased number of bits required to encode the depth atlases. The value of f can be expressed by (2), and the value of k to limit the scale factor f is set to 1.5 based on the experimental results for different values of k. A comprehensive breakdown of the experimental results is presented in Section IV-B.

$$f = \min\left(k, \frac{2^b - 1}{\text{patchMaxDepth} - \text{patchMinDepth}}\right) \quad (2)$$

B. ADAPTIVE APPLICATION OF DEPTH SCALING

The proposed depth scaling is designed to be adaptive based on the characteristics of the patches. In other words, the proposed algorithm is not applied to all patches but only to those patches where the application of the algorithm has a positive effect. We found that in the histogram of the original depth values, patches with a wide distribution of samples were less likely to generate depth errors owing to depth quantization. In other words, if the proposed depth scaling is applied to this type of patch, performance degradation may occur in terms of the BD-rate owing to an increase in the coding bit rate without improving the image quality. Therefore, the proposed depth-scaling method cannot be applied to patches where the minimum sample interval of the depth values within the patch is greater than $2^b/2^n$, considering that the depth values vary from b to n bits. Because the default conditions of the depth range for the TMIV are b= 16 and n= 10, the threshold for the minimum sample interval is set to 64 by default. An experiment was conducted to demonstrate the gains that can be achieved by adaptively applying the proposed algorithm; detailed descriptions of the experimental results are presented in Section IV-B.

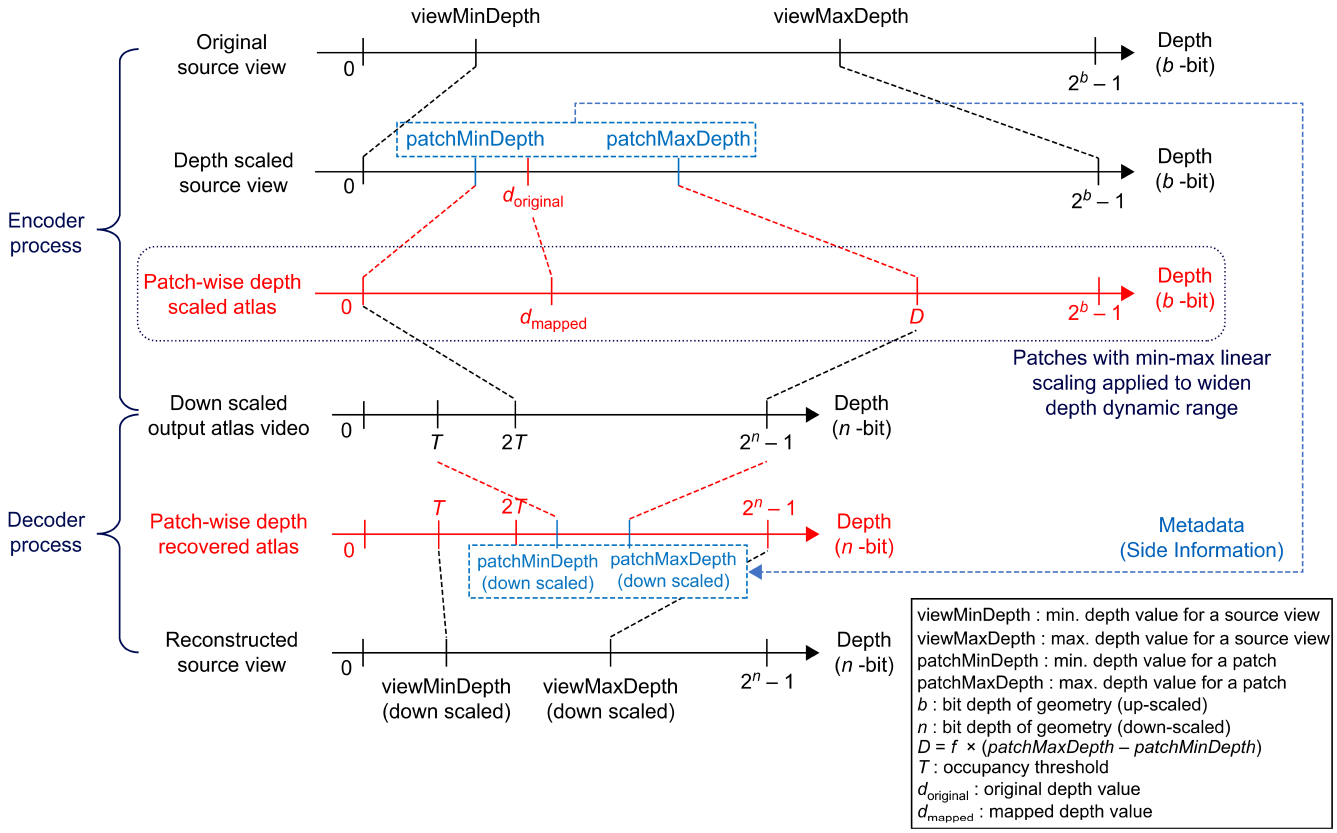

FIGURE 6. Conceptual diagram of the proposed depth scaling process with per-patch depth scaling.

In addition, the proposed depth range scaling algorithm adapts to the quality of the original depth information per sequence. The proposed algorithm aims to minimize errors in depth information by expanding the dynamic range for depth-value representation. However, if the original depth information is inaccurate, the improvement in the rendering quality is less, even if the depth values are transmitted without errors. In such cases, the proposed method may result in performance loss in terms of the BD-rate because the increase in the rendered viewport peak signal-to-noise ratio (PSNR) is smaller than the increase in the depth-atlas encoding bitrate. Therefore, generally, the proposed algorithm should be applied only to computer-generated (CG) sequences, where depth information is highly reliable, and not natural content (NC) sequences with low depth quality [22].

C. SIGNALING MIN/MAX DEPTH VALUE PER PATCH

After the TMIV encoder applies patch-wise depth scaling and encodes the atlases, the TMIV decoder must restore the original depth values by applying patch-wise inverse depth scaling. Therefore, the minimum and maximum original depth values of all patches obtained using the proposed method are included in the metadata and sent to the decoder. However, transmitting the minimum and maximum depth values of multiple patches requires several bits. Therefore, it is important to develop an efficient method to encode the depth information of each patch. This section introduces a method to embed the min/max depth information of each patch in the metadata with the least number of bits.

To minimize the number of bits required to encode the depth min/max values per patch, it is necessary to reduce the absolute values of the transmitted data. Instead of transmitting the per-patch depth min/max values as they are, we used two differential coding schemes to reduce the number of bits. The first is intra-differential coding, which transmits a range of depth values instead of the maximum depth within a single patch, as shown in Fig. 7 (blue font). The second is inter-differential coding, which transmits the difference in depth information between patches at adjacent indices in the patch list, as shown in Fig. 7 (in red). By combining these two methods, the difference values are transmitted instead of the depth minima and maxima for each patch, and the decoder recovers the original values through backward calculation. An experiment was conducted to determine the extent to which the amount of transmitted metadata can be reduced by differential coding; detailed descriptions of the results are presented in Section IV-B.

IV. EXPERIMENTAL RESULTS

This section describes the results of the experiments conducted to evaluate the performance of the proposed patch-wise depth range scaling method and their comparison

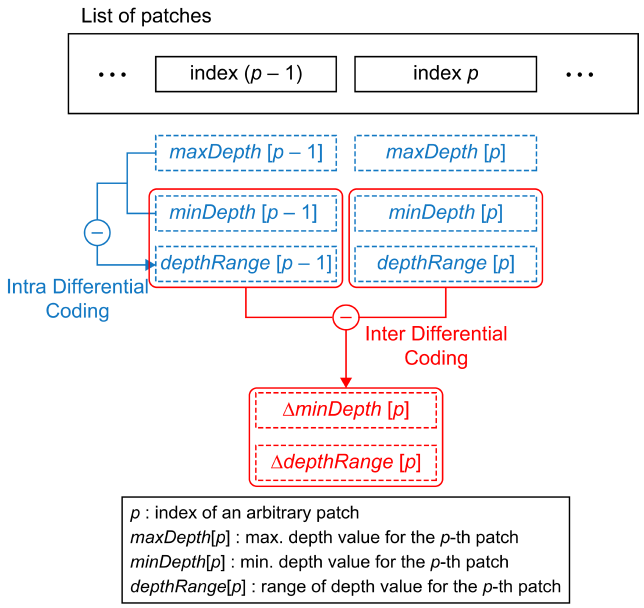

FIGURE 7. Conceptual diagram of differential coding for transmitting minimum and maximum per-patch depth values.

TABLE 1. Test-sequence configurations in MIV CTCs.

Name	Projection type	Views	Geometry bit-depth	View Resolution
ClassroomVideo	Full-ERP	15	16	4096 × 2048
Museum	Half-ERP	24	16	2048 × 2048
Fan	Perspective	15	16	1920 × 1080
Kitchen	Perspective	25	**10**	1920 × 1080
Chess	Half-ERP	10	16	2048 × 2048
Group	Perspective	21	16	1920 × 1080
Hijack	Half-ERP	10	16	4096 × 2048
ChessPieces	Half-ERP	10	16	2048 × 2048
Cadillac	Perspective	15	16	1920 × 1080

TABLE 2. Quantization parameters for texture atlas coding in MIV CTCs [21].

Name	QP1	QP2	QP3	QP4	QP5
ClassroomVideo	27	29	33	41	49
Museum	33	41	46	49	51
Fan	31	34	41	46	51
Kitchen	21	27	31	36	41
Chess	17	24	28	34	40
Group	24	30	35	38	42
Hijack	14	17	22	29	35
ChessPieces	17	22	28	36	43
Cadillac	24	30	37	44	51

TABLE 3. Varying the maximum scale factor k as measured by the High-BR BD-Rate of WS-PSNR.

Sequence	High-BR BD rate WS-PSNR (Y)		
	$k = 1.25$	$k = 1.5$	$k = 1.75$
ClassroomVideo	1.1%	2.4%	3.7%
Museum	-3.2%	-3.2%	-3.0%
Fan	-0.9%	-1.2%	-1.6%
Kitchen	0.0%	0.0%	0.0%
Chess	-18.9%	-21.1%	-20.8%
Group	-36.1%	-42.5%	-43.0%
Hijack	-12.8%	-15.1%	-14.0%
ChessPieces	-7.6%	-8.8%	-7.8%
Cadillac	-1.4%	-1.2%	-1.9%
Average	**-8.9%**	**-10.1%**	**-9.8%**

(TMIV 14.0, 17 FRAMES)

with the depth range scaling currently employed in TMIV (described in Section II-B). The proposed method, which is based on the TMIV framework, has been not only conceptualized but also put into practice. Its performance has been rigorously assessed within the context of the common test conditions (CTCs) established for MIV [23]. The following subsections present the specifics of the test conditions and the results obtained from the experiments using the proposed method.

A. EXPERIMENTAL CONDITIONS

The MIV CTC was established to enable the application of uniform evaluation conditions for a fair comparison of all the technologies proposed for the MIV group. The standard covers several aspects, including the definition of different types of test sequences, delineation of anchor generation methods, and establishment of guidelines and templates for the presentation of experimental results obtained from contributed technologies.

Table 1 lists the sequences used in the experiments presented in this paper. The test sequences are CG sequences from those specified in the MIV CTCs, each consisting of a different camera array with various resolutions in perspective and an equirectangular projection (ERP) format. Using these sequences as inputs, the MIV encoder produces textures and depth atlases, which are then subjected to compression. The compression is achieved using a random-access configuration of versatile video coding (VVC) [24], a standard codec. The compression process is facilitated using two open-source software implementations: VVenC [25] for encoding and VVdeC [26] for decoding.

TABLE 4. Encoding performance W/Wo adaptive algorithm considering patch depth sample interval as measured by the high-BR BD-rate of WS-PSNR.

Sequence	High-BR BD-rate WS-PSNR (Y)	
	Fixed depth scaling	Adaptive depth scaling
ClassroomVideo	2.4%	2.4%
Museum	-3.2%	-3.2%
Fan	-1.2%	-1.2%
Kitchen	**20.4%**	**0.0%**
Chess	-21.1%	-21.1%
Group	-42.5%	-42.5%
Hijack	**-14.6%**	**-15.1%**
ChessPieces	-8.8%	-8.8%
Cadillac	-1.2%	-1.2%
Average	**-7.8%**	**-10.1%**

(k = 1.5, TMIV 14.0, 17 FRAMES)

TABLE 5. Generated metadata W/Wo differential coding for per-patch depth information transmission.

Sequence	Amount of generated metadata (kbps)		Reduction ratio (%)
	Without differential coding	With differential coding	
ClassroomVideo	326.34	281.31	-13.8%
Museum	214.79	183.15	-14.7%
Fan	236.32	206.26	-12.7%
Kitchen	276.75	276.75	-0.0%
Chess	188.29	163.14	-13.4%
Group	300.95	262.25	-12.9%
Hijack	198.23	171.97	-13.2%
ChessPieces	364.32	314.19	-13.8%
Cadillac	301.81	266.41	-11.7%
Average	**267.53**	**236.16**	**-11.8%**

(k = 1.5, TMIV 14.0, 17 FRAMES)

The texture atlas is compressed using the quantization parameters (QPs) that vary by sequence, as listed in Table 2. The QPs for depth atlas encoding are calculated as follows:

$$\mathrm{QP}_d = max(1, \lfloor -14.2 + 0.8 \times \mathrm{QP}_t \rfloor), \quad (3)$$

where QP_d is the QP value of the depth atlas and QP_t is the QP value of the texture atlas. The QP conditions are organized according to five target bitrates, and to account for different bandwidth conditions, the performance is evaluated by distinguishing two bitrate (BR) modes: high-BR for the four lowest QPs and low-BR for the four highest QPs.

To objectively measure the rendering quality, views with the same positions as the input views were synthesized, and the PSNR was measured for all synthesized views relative to the original views. Two types of metrics were used for objective quality measurement: weighted spherical PSNR (WS-PSNR) [27] and immersive video PSNR (IV-PSNR) [28]. The coding performance of each metric was determined by calculating the BD-rate. In addition, the subjective quality assessment includes the evaluation of viewport videos synthesized along a user-defined virtual trajectory, known as a pose trace. All the experiments were conducted on an Intel Xeon Gold 6242R 3.10 GHz processor with Visual C++ 16.0 compiler under the Windows 11 operating system.

TABLE 6. Final encoding performance of immersive videos using the proposed method as evaluated by the BD-Rate.

Sequence	High-BR BD rate WS -PSNR (Y)	Low-BR BD rate WS -PSNR (Y)	High-BR BD rate IV-PSNR	Low-BR BD rate IV-PSNR
ClassroomVideo	4.1%	6.2%	4.3%	6.0%
Museum	-3.3%	-0.3%	1.1%	2.1%
Fan	0.0%	1.6%	1.9%	3.4%
Kitchen	0.0%	0.0%	0.0%	0.0%
Chess	-18.8%	-6.9%	-2.5%	0.6%
Group	-39.8%	-23.4%	-19.9%	-11.2%
Hijack	-15.8%	-5.4%	-4.3%	-0.4%
ChessPieces	-7.5%	-0.7%	0.7%	3.3%
Cadillac	-0.6%	-0.9%	0.4%	0.7%
Average	**-9.1%**	**-3.3%**	**-2.0%**	**0.5%**

(k = 1.5, TMIV 14.0, 97 FRAMES)

TABLE 7. Encoding and decoding time increase for TMIV with the proposed method.

Sequence	TMIV Encoding Time Increment Ratio	TMIV Decoding Time Increment Ratio
ClassroomVideo	102.3%	101.2%
Museum	103.1%	101.0%
Fan	101.1%	101.3%
Kitchen	102.4%	101.3%
Chess	105.0%	103.2%
Group	101.1%	100.1%
Hijack	101.0%	101.0%
ChessPieces	102.3%	101.0%
Cadillac	102.6%	101.2%
Average	**102.3%**	**101.3%**

(k = 1.5, TMIV 14.0, 97 FRAMES)

B. PERFORMANCE EVALUATIONS

First, we evaluate the performance of the proposed depth range scaling method by varying the value of k to limit the scaling factor to {1.25, 1.5, 1.75}. In this experiment, the proposed method was integrated into TMIV 14.0 [29], and its evaluation was performed according to the MIV CTCs. As shown in Table 3, the comprehensive coding performance over three different k values {1.25, 1.5, 1.75} resulted in

FIGURE 8. Examples of the rendered anchor viewports (Left) and proposed viewports (Right) in the Group sequence (QP4).

average BD-rate improvements of 8.9%, 10.1%, and 9.8%, respectively, for the luma component, specifically in the context of a high-BR WS-PSNR. Based on the experimental results, the process of identifying an appropriate k value led to improved coding efficiency. In particular, the k value of 1.5 emerges as the optimal choice, resulting in favorable end-to-end coding performance.

Table 4 presents the coding efficiencies with and without the adaptive algorithm, considering the sample interval of the depth values in the patches. Applying the proposed depth-scaling algorithm to all patches without excluding patches with a sparse depth-value distribution resulted in an average gain of 7.8%. However, not adaptively applying the proposed algorithm to patches with a minimum sample interval of 64 or more depth values resulted in an average gain of 10.1%. In particular, in the Kitchen sequence [30], applying depth scaling to all patches resulted in a severe loss of 20.4% in the coding performance. As shown in Table 1 in Section IV-A, only the bit depth of the input geometry of the Kitchen sequence is 10 bits, which is upscaled to 16 bits within the TMIV encoder before being processed in the same manner as the other sequences. Therefore, all patches in the Kitchen sequence have a sparse distribution of depth values and cannot be expected to benefit from the proposed method. Therefore, by not applying depth scaling to patches having a sparse distribution of depth values, we can avoid the performance degradation pertaining to the Kitchen sequence, with very small performance gains in other sequences.

Table 5 compares the amount of metadata generated with and without differential coding in the transmission of per-patch depth information. Using differential coding to reduce the number of bits required to encode the per-patch

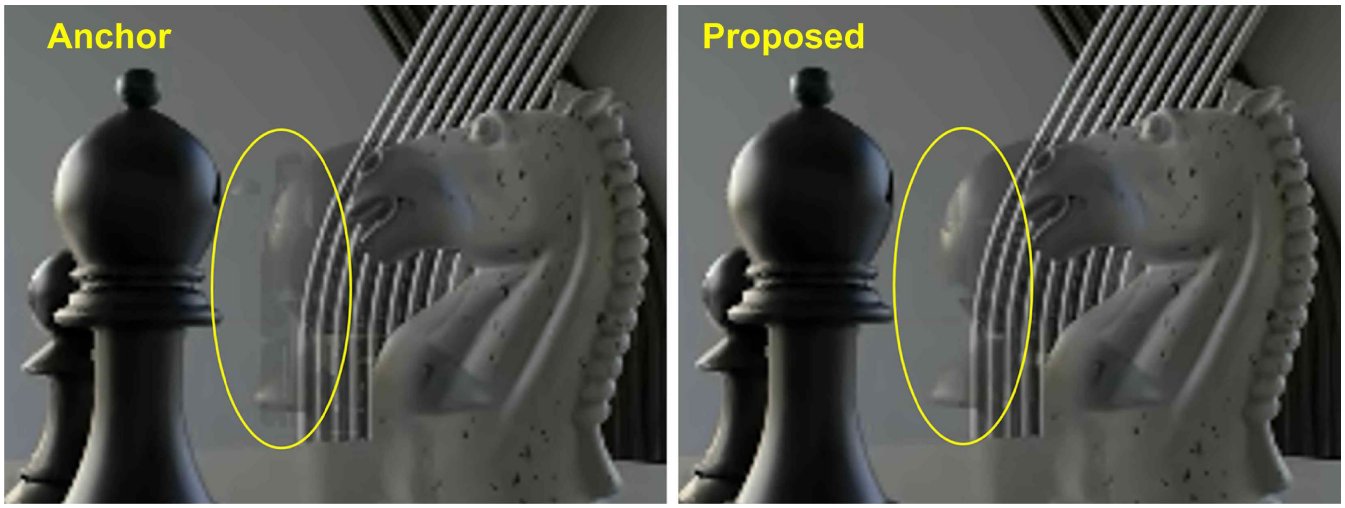

FIGURE 9. An example of the rendered anchor viewport (Left) and proposed viewport (Right) in the Chess sequence (QP4).

FIGURE 10. An example of the rendered anchor viewport (Left) and proposed viewport (Right) in the Hijack sequence (QP4).

depth value information, the total amount of metadata can be reduced by approximately 11.8%.

Table 6 demonstrates the end-to-end coding performance of the immersive video with the patch-wise depth range scaling proposed in this study compared to that of TMIV with the view-wise depth range scaling method described in Section II-B. In the experiments, the proposed method was implemented in TMIV 14.0 with the maximum scale factor k set to 1.5. The differential coding was applied to encode the minimum and maximum depth values per patch. In the overall assessment, significant reductions of 9.1% and 2.0% in the BD-rate were observed for the high-BR BD-rate of the luma component in WS-PSNR and IV-PSNR, respectively. These reductions were compared with those of the conventional view-wise depth-scaling method. In particular, the Group [31] sequence exhibited the highest performance improvement of 39.8%. In addition, the proposed method tends to perform better in the high-BR range than in the low-BR range. The ClassroomVideo [32] sequence was the only sequence in which the performance of the proposed method degraded noticeably. This is because of a slight improvement in the quality of the rendered viewport but a larger bitrate increase in the depth atlas.

Table 7 shows that the proposed depth scaling slightly increases the complexity of TMIV by increasing the encoding and decoding times by 2.3% and 1.3%, respectively.

TMIV consists of an immersive video pre and postprocessor and a conventional 2D codec for encoding the atlas generated by the preprocessor. Incorporating the proposed method into the pre and postprocessing stages of TMIV resulted in a small, practically negligible increase in complexity. Therefore, applying the proposed depth-scaling technique to TMIV does not add significant complexity.

Figs. 8, 9, and 10 show a subjective quality comparison of viewports rendered with TMIV anchors using traditional depth scaling methods and the proposed method on the Group [31], Chess [33], and Hijack [7] sequences. The severe artifacts observed in the anchor viewport in these figures are significantly reduced (yellow circles) in the viewport using the proposed method. In particular, we can observe an improvement in artifacts at the object boundaries, where errors in depth information are sensitive to compositing. As illustrated by the examples in the viewport, the proposed method visibly improves the visual quality of certain sequences and brings about significant improvements in encoding performance, as measured by the BD-rate.

V. CONCLUSION

In this study, we proposed patch-wise depth range scaling to improve the performance of immersive video coding by preserving depth information more accurately. In the proposed method, min–max linear scaling is applied to the depth value on a per-patch basis to expand the depth dynamic range. We limit the scaling factor with which the dynamic range of depth is expanded per patch to a maximum of 1.5 times to prevent excessive scaling. In addition, if the histogram distribution of the depth values within a patch has a large gap between samples, we consider the application of depth range scaling to be ineffective and do not apply the proposed algorithm to that patch. In addition, sequences with low reliability of the original depth information (e.g., NC sequences) are excluded because it is difficult to expect a performance improvement using the proposed method. Finally, we combine intra- and inter-differential coding to encode and transmit the per-patch depth minima and maxima to the decoder to minimize bit consumption.

Experimental results indicated that the proposed method provides a significant coding advantage throughout the process, as indicated by the improvement in BD-rate for immersive videos. A noticeable subjective quality improvement is observed with an average coding gain of 9.1%. Moreover, the proposed method has little effect on the complexity of both the TMIV encoder and decoder. However, the depth information in different sequences is diverse, and the performance differs significantly, which is a drawback. Further research on adaptive algorithms is required to solve this problem.

REFERENCES

[1] J. Chakareski, "UAV-IoT for next generation virtual reality," *IEEE Trans. Image Process.*, vol. 28, no. 12, pp. 5977–5990, Dec. 2019, doi: 10.1109/TIP.2019.2921869.

[2] Z. Lai, Y. C. Hu, Y. Cui, L. Sun, N. Dai, and H.-S. Lee, "Furion: Engineering high-quality immersive virtual reality on today's mobile devices," *IEEE Trans. Mobile Comput.*, vol. 19, no. 7, pp. 1586–1602, Jul. 2020, doi: 10.1109/TMC.2019.2913364.

[3] M. Domanski, O. Stankiewicz, K. Wegner, and T. Grajek, "Immersive visual media—MPEG-I: 360 video, virtual navigation and beyond," in *Proc. Int. Conf. Syst., Signals Image Process. (IWSSIP)*, Poznan, Poland, May 2017, pp. 1–9, doi: 10.1109/IWSSIP.2017.7965623.

[4] F. Isgro, E. Trucco, P. Kauff, and O. Schreer, "Three-dimensional image processing in the future of immersive media," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 14, no. 3, pp. 288–303, Mar. 2004, doi: 10.1109/TCSVT.2004.823389.

[5] P. Merkle, A. Smolic, K. Müller, and T. Wiegand, "Multi-view video plus depth representation and coding," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2007, p. 201, doi: 10.1109/ICIP.2007.4378926.

[6] *Call for Proposals on 3Dof+ Visual*, document ISO/IEC JTC1/SC29/WG11 MPEG/N18709, Marrakesh, Morocco, Jan. 2019.

[7] R. Doré, G. Briand, and T. Tapie. *Technicolor 3DoF+ Test Materials*, document ISO/IEC JTC1/SC29/WG11/M42349, Apr. 2018.

[8] R. Koenen and M. Champel. *Requirements MPEG-I Phase 1b*, document ISO/IEC JTC1/SC29/WG11 MPEG/N7331, Gwanju, (South) Korea, Jan. 2018.

[9] *Text of ISO/IEC DIS 23090-12 MPEG Immersive Video*, document ISO/IEC JTC1/SC29/WG4/N0049, Apr. 2021.

[10] *Preliminary WD5 of ISO/IEC 23090-12 MPEG immersive Video*, document ISO/IEC JTC1/SC29/WG4/N0305, Jan. 2023.

[11] *Draft Use Case and Requirements for MIV—Edition-2*, document ISO/IEC JTC1/SC29/WG2/N0157, Jan. 2022.

[12] *Test Model 15 for MPEG Immersive Video*, document ISO/IEC JTC1/SC29/WG4/N0271, Nov. 2022.

[13] J. M. Boyce, R. Doré, A. Dziembowski, J. Fleureau, J. Jung, B. Kroon, B. Salahieh, V. K. M. Vadakital, and L. Yu, "MPEG immersive video coding standard," *Proc. IEEE*, vol. 109, no. 9, pp. 1521–1536, Sep. 2021, doi: 10.1109/JPROC.2021.3062590.

[14] V. K. M. Vadakital, A. Dziembowski, G. Lafruit, F. Thudor, G. Lee, and P. R. Alface, "The MPEG immersive video standard—Current status and future outlook," *IEEE MultimediaMag.*, vol. 29, no. 3, pp. 101–111, Jul. 2022, doi: 10.1109/MMUL.2022.3175654.

[15] A. Dziembowski, D. Mieloch, M. Domanski, G. Lee, and J. Jeong. *Immersive Video CE1.2: Geometry Scaling*, document ISO/IEC JTC1/SC29/WG11 MPEG/M54176, Jun. 2020.

[16] S.-G. Lim, D. Park, J.-G. Kim, J. Jeong, K.-J. Oh, and G. Lee. *[MIV] CE3.2: Piecewise Linear Scaling of Geometry Atlas*, document ISO/IEC JTC1/SC29/WG4/M57419, Jul. 2021.

[17] S.-G. Lim, D.-H. Kim, J.-G. Kim, K.-J. Oh, J. Jeong, and G. Lee. *Wider Depth Dynamic Range Using Occupancy Map Correction*, document ISO/IEC JTC1/SC29/WG4/M60192, Jul. 2022.

[18] G. Bjøntegaard. *Calculation of Average PSNR Differences Between RD-Curves*, document VCEG-M33, Mar. 2001.

[19] S.-G. Lim, H.-H. Kim, and Y.-H Kim. *Adaptive Patch-wise Depth Range Linear Scaling*, document ISO/IEC JTC1/SC29/WG4 MPEG/M62701, Apr. 2023.

[20] *Description of MPEG Immersive Video 2nd Edition Core Experiments 2*, document ISO/IEC JTC1/SC29/WG4/N0345, Apr. 2023.

[21] D. Park, S.-G. Lim, K.-J. Oh, G. Lee, and J.-G. Kim, "Nonlinear depth quantization using piecewise linear scaling for immersive video coding," *IEEE Access*, vol. 10, pp. 4483–4494, 2022, doi: 10.1109/ACCESS.2022.3140537.

[22] A. Schenkel, D. Bonatto, S. Fachada, H.-L. Guillaume, and G. Lafruit, "Natural scenes datasets for exploration in 6DOF navigation," in *Proc. Int. Conf. 3D Immersion (ICD)*, Brussels, Belgium, Dec. 2018, pp. 1–8, doi: 10.1109/IC3D.2018.8657865.

[23] J. Jung and B. Kroon. *Common Test Conditions for MPEG Immersive Video*, document ISO/IEC JTC1/SC29/WG4/N0232, Jul. 2022.

[24] *Versatile Video Coding*, Standard ISO/IEC 23090-3, ISO/IEC JTC1/SC29, Jul. 2020.

[25] *Fraunhofer HHI VVenC Software Repository*. Accessed: May 2021. [Online]. Available: https://github.com/fraunhoferhhi/vvenc

[26] *Fraunhofer HHI VVdeC Software Repository*. Accessed: May 2021. [Online]. Available: https://github.com/fraunhoferhhi/vvdec

[27] *WS-PSNR Software Manual*, document ISO/IEC JTC1/SC29/WG11/N18069, Oct. 2018.

[28] *Software Manual of IV-PSNR for Immersive Video*, document ISO/IEC JTC1/SC29/WG11 MPEG/N18709, Jul. 2019.

[29] B. Salahieh, J. Jung, and A. Dziembowski. *Test Model 14 for MPEG Immersive Video*, document ISO/IEC JTC1/SC29/WG4/N0242, Jul. 2022.
[30] P. Boissonade and J. Jung. *[MPEG-I Visual] Proposition of New Sequences for Windowed-6DoF Experiments on Compression, Synthesis, and Depth Estimation*, document ISO/IEC JTC1/SC29/WG11 MPEG/M43318, Jul. 2018.
[31] R. Doré, G. Briand, and F. Thudor. *Group Content for MIV*, document ISO/IEC JTC1/SC29/WG11/M54731, Jul. 2020.
[32] B. Kroon, *3DoF+ Test Sequence ClassroomVideo*, document ISO/ IEC JTC1/SC29/WG11/M42415, Apr. 2018.
[33] L. Ilola, V. Vadakital, K. Roimela, and F. Kerānen, *New Test Content for Immersive Video—Nokia Chess*, document ISO/IEC JTC1/SC29/WG11/M50787, Oct. 2019.

SUNG-GYUN LIM received the B.S. and M.S. degrees in electronics and information engineering from Korea Aerospace University, Goyang-si, Gyeonggi-do, Republic of Korea, in 2020 and 2022, respectively.

He joined the Korea Electronics Technology Institute (KETI), Seongnam-si, Republic of Korea, in 2022. He has been involved in several projects focused on immersive video processing and video coding standardization. His current research interests include the development of immersive video processing and video coding techniques.

HYUN-HO KIM received the B.S. and M.S. degrees in electronics and information engineering from Korea Aerospace University, Goyang-si, Gyeonggi-do, Republic of Korea, in 2018 and 2020, respectively.

He joined the Korea Electronics Technology Institute (KETI), Seongnam-si, Republic of Korea, in 2020. He has been involved in several projects focused on immersive video coding standardization. His current research interests include the development of immersive video coding, particularly depth-map processing techniques.

YONG-HWAN KIM was born in Jeju, South Korea, in 1972. He received the B.S. and M.S. degrees in electrical engineering and the Ph.D. degree in image engineering from Chung-Ang University, Seoul, South Korea, in 1996, 1998, and 2008, respectively.

From 1999 to 2001, he was with SungJin C&C, Seoul, where he optimized MPEG-1/2 video CODEC for DVR. Since 2001, he has been with the Korea Electronics Technology Institute (KETI), Seongnam-si, South Korea, where he is currently a Chief Researcher with the Intelligent Image Processing Research Center. His current research interests include V-PCC, VVC, AV1, MIV video coding, and their optimized implementations.