electronics

MDPI

Article

Two-Dimensional Audio Compression Method Using Video Coding Schemes

Seonjae Kim [1], Dongsan Jun [1,*], Byung-Gyu Kim [2,*], Seungkwon Beack [3], Misuk Lee [3] and Taejin Lee [3]

[1] Department of Convergence IT Engineering, Kyungnam University, Changwon 51767, Korea; sjkim@kyungnam-ispl.kr
[2] Department of IT Engineering, Sookmyung Women's University, Seoul 04310, Korea
[3] Electronics and Telecommunications Research Institute (ETRI), Daejeon 34129, Korea; skbeack@etri.re.kr (S.B.); lms@etri.re.kr (M.L.); tjlee@etri.re.kr (T.L.)
* Correspondence: dsjun9643@kyungnam.ac.kr (D.J.); bg.kim@sookmyung.ac.kr (B.-G.K.)

Abstract: As video compression is one of the core technologies that enables seamless media streaming within the available network bandwidth, it is crucial to employ media codecs to support powerful coding performance and higher visual quality. Versatile Video Coding (VVC) is the latest video coding standard developed by the Joint Video Experts Team (JVET) that can compress original data hundreds of times in the image or video; the latest audio coding standard, Unified Speech and Audio Coding (USAC), achieves a compression rate of about 20 times for audio or speech data. In this paper, we propose a pre-processing method to generate a two-dimensional (2D) audio signal as an input of a VVC encoder, and investigate the applicability to 2D audio compression using the video coding scheme. To evaluate the coding performance, we measure both signal-to-noise ratio (SNR) and bits per sample (bps). The experimental result shows the possibility of researching 2D audio encoding using video coding schemes.

Keywords: audio compression; video compression; next generation audio coding; versatile video coding (VVC); mu-law encoding; linear mapping

check for updates

Citation: Kim, S.; Jun, D.; Kim, B.-G.; Beack, S.; Lee, M.; Lee, T. Two-Dimensional Audio Compression Method Using Video Coding Schemes. *Electronics* **2021**, *10*, 1094. https://doi.org/10.3390/electronics10091094

Academic Editor: Juan M. Corchado

Received: 23 March 2021
Accepted: 3 May 2021
Published: 6 May 2021

1. Introduction

As consumer demands for realistic and rich media services on low-end devices are rapidly increasing in the field of multimedia delivery and storage applications, the need for audio or video codecs with powerful coding performance is emphasized, which can achieve minimum bitrates and maintain higher perceptual quality compared with the original data. As the state-of-the-art video coding standard, Versatile Video Coding (VVC) [1] was developed by the Joint Video Experts Team (JVET) of the ITU-T Video Coding Experts Group (VCEG) and ISO/IEC Moving Picture Experts Group (MPEG). VVC can achieve a bitrate reduction of 50% with similar visual quality compared with the previous method, named High Efficiency Video Coding (HEVC) [2]. In the field of audio coding technology, Unified Speech and Audio Coding (USAC) [3] was developed by the MPEG audio group as the latest audio coding standard, which integrates speech and audio coding schemes. Whereas VVC can accomplish significant coding performance in the original image or video data, USAC can provide about 20 times compression performance of the original audio or speech data.

In this study, we conducted various experiments to verify the possibility of compressing two-dimensional (2D) audio signals using video coding schemes. In detail, we converted 1D audio signal into 2D audio signal with the proposed 2D audio conversion process as an input of a VVC encoder. We used both signal-to-noise ratio (SNR) and bits per sample (bps) for performance evaluations.

The remainder of this paper is organized as follows: In Section 2, we overview the video coding tools, which are newly adopted in VVC. In Section 3, we describe the proposed methods for converting a 1D audio signal into a 2D audio signal. Finally, the experimental results and conclusions are provided in Sections 4 and 5, respectively.

2. Overview of VVC

VVC can provide powerful coding performance compared with HEVC. One of the main differences between HEVC and VVC is the block structure. Both HEVC and VVC commonly specify coding tree unit (CTU) as the largest coding unit, which has a changeable size depending on the encoder configuration. In addition, a CTU can be split into four coding units (CUs) by a quad tree (QT) structure to adapt to a variety of block properties. In HEVC, a CU can be further partitioned into one, two, or four prediction units (PUs) according to the PU splitting type. After obtaining the residual block derived from the PU-level intra- or inter-prediction, a CU can be partitioned into multiple transform units (TUs) according to a residual quad-tree (RQT) structure similar to that of CU split.

VVC substitutes the concepts of multiple partition unit types (CU, PU, and TU) with a QT-based multi-type tree (QTMTT) block structure, where the MTT is classified into binary tree (BT) split and ternary tree (TT) split to support more flexibility for CU partition shapes. This means that a QT can be further split by the MTT structure after a CTU is first partitioned by a QT structure. As depicted in Figure 1, VVC specifies four MTT split types: vertical binary split (SPLIT_BT_VER), horizontal binary split (SPLIT_BT_HOR), vertical ternary split (SPLIT_TT_VER), and horizontal ternary split (SPLIT_TT_HOR), as well as QT split. In VVC, a QT or MTT node is considered a CU for prediction and transform processes without any further partitioning schemes. Note that CU, PU, and TU have the same block size in the VVC block structure. In other words, a CU in VVC can have either a square or rectangular shape, whereas a CU in HEVC always has a square shape.

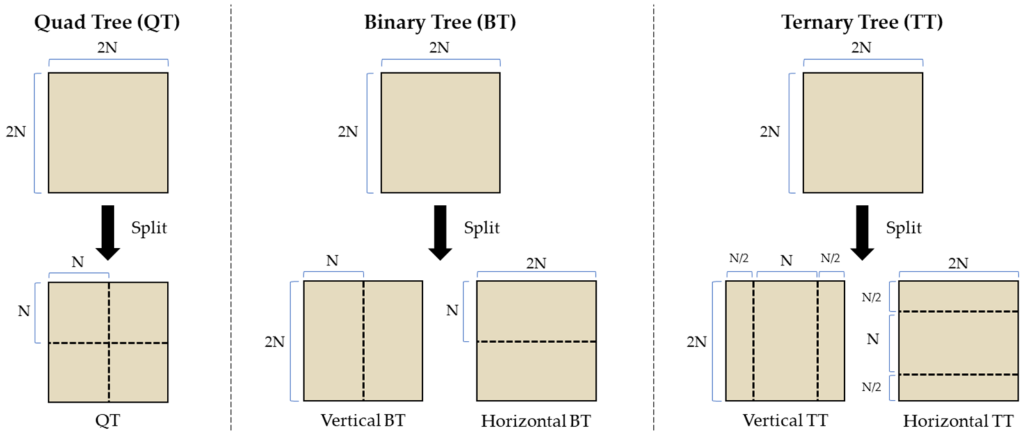


Figure 1. VVC block structure (QTMTT).

Table 1 shows newly adopted coding tools between HEVC and VVC. In general, intra-prediction generates a predicted block from the reconstructed neighboring pixels of the current block. As shown in Figure 2, VVC can provide 67 intra-prediction modes, where modes 0 and 1 are planar and DC mode, respectively, and the others are in angular prediction mode to represent edge direction. According to [4], the VVC test model (VTM) [5] achieves 25% higher compression performance than the HEVC test model (HM) [6] under the all-intra (AI) configuration recommended by JVET Common Test Conditions (CTC) [7]. This improvement was mainly realized by the newly adopted coding tools, such as position dependent intra-prediction combination (PDPC) [8], cross component linear model intra-prediction (CCLM) [9], wide angle intra-prediction (WAIP) [10], and matrix-based intra-prediction (MIP) [11]; the computational complexity of encoder substantially increased approximately 26 times [4].

Table 1. The list of video coding tools. Comparison between HEVC and VVC.

Category	HEVC	VVC
Intra prediction	35 Intra-prediction modes	67 intra-prediction modes Wide angle intra-prediction (WAIP) Position-dependent intra-prediction combination (PDPC) Matrix-based intra-prediction (MIP) Multi reference line prediction (MRL) Intra sub-block partitioning (ISP) Cross-component linear model (CCLM)
Skip/Merge	Regular skip/merge	Regular skip/merge Sub-block-based temporal motion vector predictors (SbTMVP) Geometric partitioning mode (GPM) Merge with motion vector difference (MMVD) Decoder-side motion vector refinement (DMVR) Bi-directional optical flow (BDOF) Combined inter- and intra-prediction (CIIP)
Inter prediction	Advanced motion vector prediction (AMVP)	Advanced motion vector prediction (AMVP) Symmetric motion vector difference (SMVD) Affine inter-prediction Adaptive motion vector resolution (AMVR) Bi-prediction with CU-level weight (BCW)

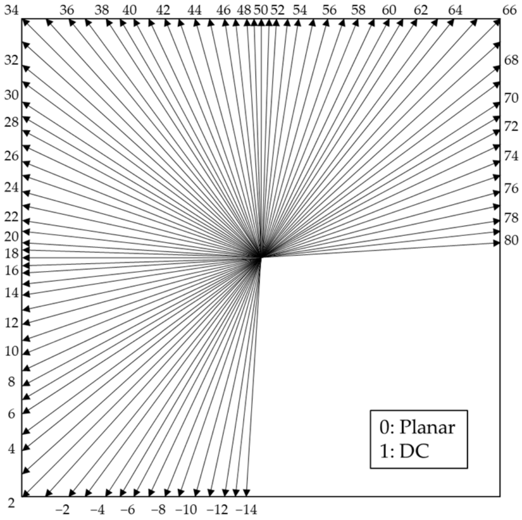


Figure 2. Intra-prediction modes in VVC.

Inter-prediction fetches a predicted block from previously decoded reference frames using motion estimation (ME) and motion compensation (MC) processes. The motion parameter can be transmitted to the decoder in either an explicit or implicit manner. When a CU is coded as SKIP or MERGE mode, the encoder does not transmit any coding parameters, such as significant residual coefficients, motion vector differences, or a reference picture index. In cases of SKIP and MERGE modes, the motion information for the current CU is derived from the neighboring CUs including spatial and temporal candidates. VVC adopted several new coding tools for SKIP/MERGE and inter-prediction, such as affine inter prediction [12], geometric partitioning mode (GPM) [13], merge with motion vector difference (MMVD) [14], decoder-side motion vector refinement (DMVR) [15], combined inter- and intra-prediction (CIIP) [16], and bi-prediction with CU-level weight (BCW) [17]. VVC also adopted adaptive motion vector resolution (AMVR) [18], symmetric motion vector difference (SMVD) [19], and advanced motion vector prediction (AMVP) to save the bitrate of the motion parameters. Although these tools can significantly improve the coding performance, the computational complexity of a VVC encoder is up to nine times higher under the random-access (RA) configuration compared with that of HEVC [4].

Per the JVET CTC [7], VVC supports three encoding configurations [20]: all-intra (AI), low-delay (LD), and random-access (RA). In the AI configuration, each picture is encoded as an intra picture and does not use temporal reference pictures. Conversely, in case of the LD configuration, only the first frame in a video sequence is encoded as an intra frame; subsequent frames are encoded using the inter-prediction tools to exploit temporal reference pictures as well as intra-prediction tools. In the RA coding mode, an intra frame is encoded in an approximately one second interval in accordance with the intra period, and other frames are encoded as hierarchical B frames along the group of pictures (GOP) size. This means that a delay may occur in RA configuration because the display order is different from the encoding order.

Although the aforementioned VVC coding tools provide significant compression performance compared with HEVC, their computational complexity [21] of the encoder is high. Therefore, many researchers have studied various fast encoding algorithms for block structure, intra-prediction, and inter-prediction of VVC while maintaining the visual quality. In terms of block structure, Fan et al. proposed a fast QTMT partition algorithm based on variance and a Sobel operator to choose only one partition from five QTMT partitions in the process of intra-prediction [22]. Jin et al. proposed a fast QTBT partition method through a convolutional neural network (CNN) [23]. To reduce the computational complexity of intra-prediction, Yang et al. proposed a fast intra-mode decision method with gradient descent search [24] and Dong et al. proposed an adaptive mode pruning method by removing the non-promising intra-modes [25]. In terms of inter-prediction, Park and Kang proposed a fast affine motion estimation method using statistical characteristics of parent and current block modes [26].

3. VVC-Based 2D Audio Compression Method

3.1. Overall Frameworks of 2D Audio Coding

In general, the original 1D audio signal is generated from pulse code modulation (PCM), which involves sampling, quantization, and binarization processes. In this study, we used a 48 kHz sampled audio signal, and each sample was already mapped to signed 16 bits binary number by PCM coding. As the inputs of VVC specify a video sequence, which is an unsigned 2D signal with a max 12 bits integer number [7], we converted the original 1D audio signal into an unsigned 2D audio signal as the pre-processing before VVC encoding.

As shown in Figure 3, the overall architectures for 2D audio compression can be divided into the encoder and decoder side, where the encoder side consists of the 2D audio conversion process and the VVC encoder. In the 2D audio conversion process, signed 1D audio signals with a bit-depth of 16 bits are mapped to unsigned 1D audio signals with the bit-depth of 8 or 10 bits and then unsigned 2D audio signals are generated by 1D-to-2D packing as an input of the VVC encoder. After receiving a compressed bitstream, the decoder side conducts VVC decoding and inverse 2D audio conversion process to reconstruct the 1D audio signal with a bit-depth of 16 bits. Note that it may strongly influence the coding performance of the VVC depending on how the 1D signal is converted to a 2D signal.

3.2. Proposed 2D Audio Conversion Process

To generate unsigned 1D audio signals, we exploited three approaches as follows:

1. Non-linear mapping (NLM) to generate unsigned 8 bits per sample;
2. Linear mapping (LM) to generate unsigned 10 bits per sample;
3. Adaptive linear mapping (ALM) to generate unsigned 10 bits per sample.

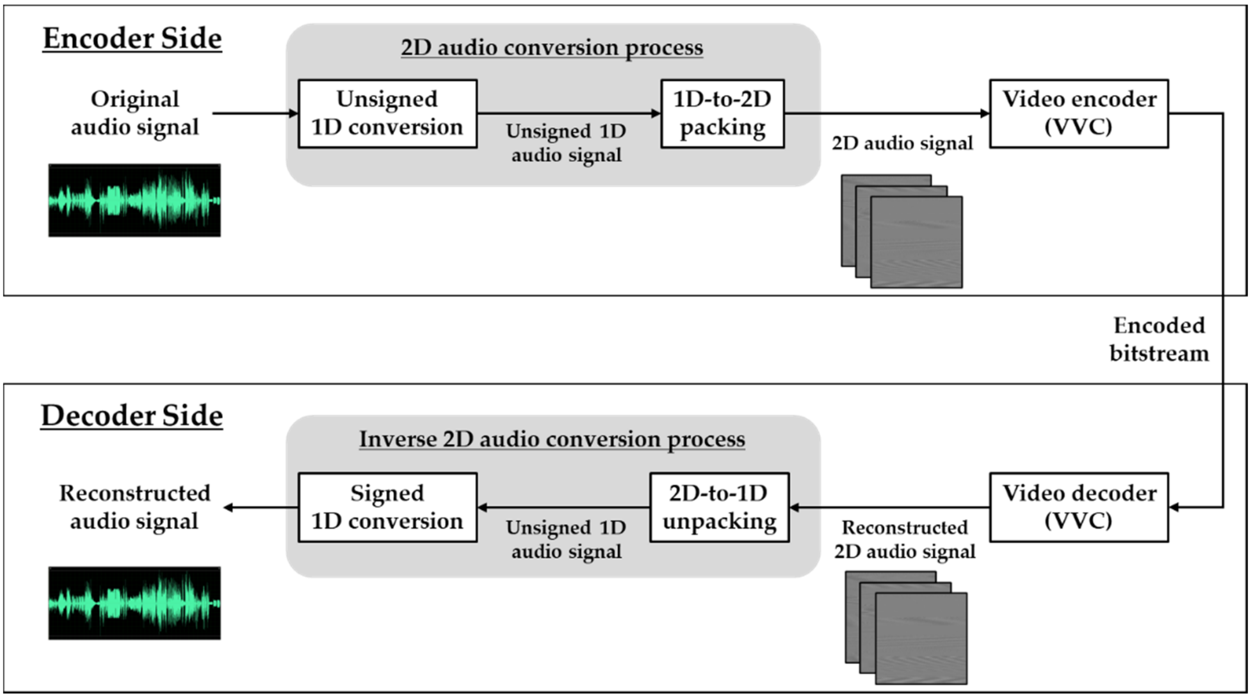


Figure 3. Overall architectures of VVC-based 2D audio coding.

NLM by Mu-law [27] is primarily used in 8 bits PCM digital telecommunication systems as one of two versions of ITU-T G.711. This reduces the dynamic range of an original signed 1D audio signal, as expressed in Equation (1):

$$NLM(I) = sgn(I) \cdot \frac{ln(1 + (2^N - 1) \cdot |I|)}{ln(1 + (2^N - 1))}, \tag{1}$$

where $sgn(I)$ and N denote the sign function and the bit-depth of output samples, respectively.

Similarly, linear mapping (LM) maps the dynamic range of an original signed 1D audio signal into a specified sample range depending on the maximum value (Max) of samples. As shown in Figure 4, the sample's dynamic range of LM is from -Max to Max. This range can be divided into uniform intervals and each interval is represented with the type of unsigned N bits by Equation (2).

$$LM(I) = \left\lfloor \frac{I + Max(|I|)}{2 \cdot Max(|I|)} \cdot \left(2^N - 1\right) \right\rfloor, \tag{2}$$

where $\lfloor x \rfloor$ and $Max(|I|)$ indicate the rounding operation and the maximum value of all samples, respectively. Note that the encoder side transmits the value of $Max(|I|)$ with the bit-depth of 16 bits for signed 1D conversion. Since this method has to search for the maximum value in the overall audio samples, this process is slow compared with the length of the audio signal. In addition, an unused sample range, called the dead zone, may exist in the range of the unsigned 1D signal, as shown in Figure 4. To resolve the aforementioned problems, we considered adaptive linear mapping (ALM). As shown in Figure 5, ALM firstly searches for both the $Min(I)$ and $Max(I)$ values among the signed 1D audio samples and then performs the LM conversion to represent the unsigned N bits 1D audio signal, as expressed in Equation (3). Because a video encoder generally compresses input data in the unit of a 2D frame, ALM can reduce the encoding delay in the unit of a frame and avoid the dead zone problem compared with LM. Therefore, the encoder side has to transmit both $Min(I_k)$ and $Max(I_k)$ values for each frame, where k is the index of the frames.

$$ALM(I_k) = \left\lfloor \frac{I_k - Min(I_k)}{Max(I_k) - Min(I_k)} \cdot \left(2^N - 1\right) \right\rfloor \quad (3)$$

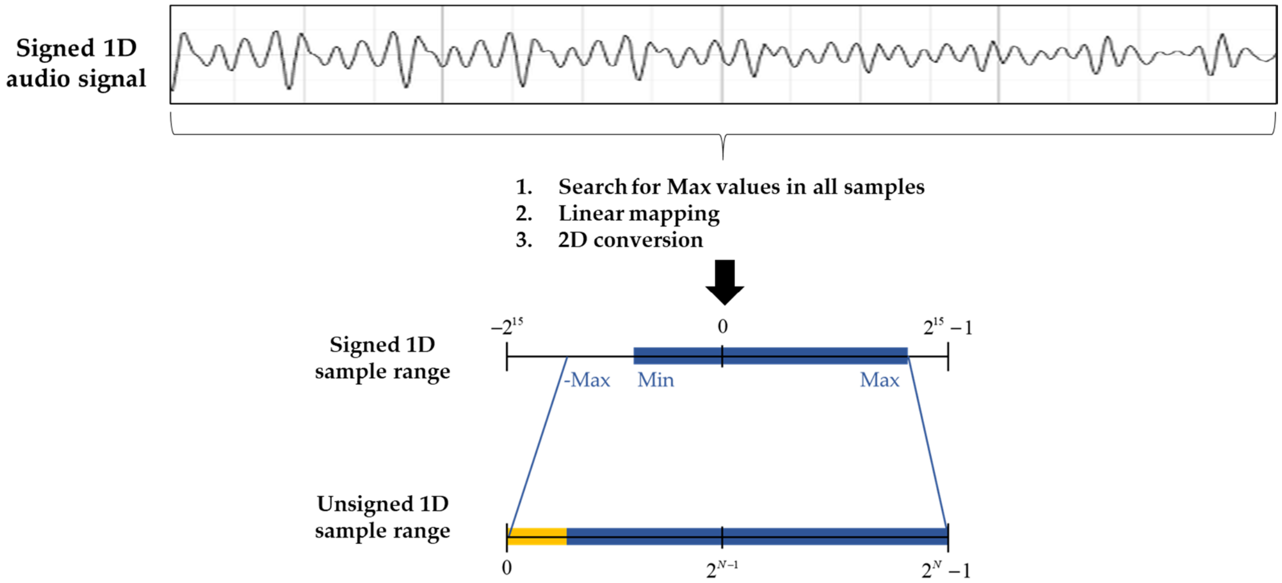


Figure 4. Graphical representation of linear mapping where the orange area indicates the unused sample range called the dead zone.

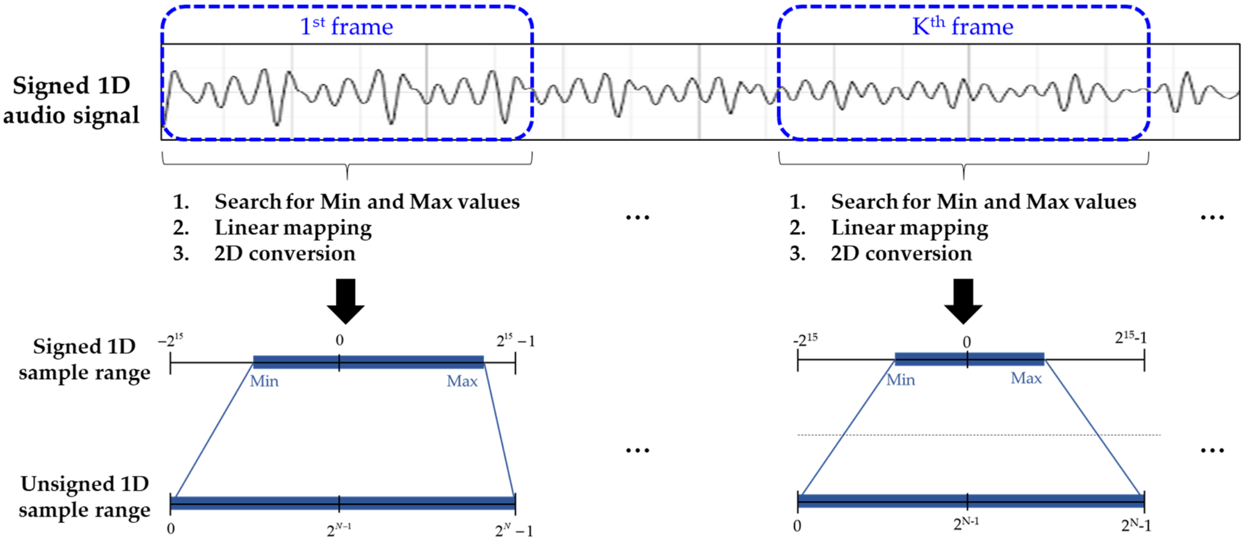


Figure 5. Graphical representation of adaptive linear mapping.

After performing unsigned 1D conversion, the 2D audio signal is generated from the unsigned 1D audio signals used as the inputs of the VVC encoder. For example, if the spatial resolution (width × height) of a frame is 256 × 256, the unsigned 1D audio signals, whose total number is 256 × 256, are arranged a frame according to the raster-scan order, as shown in Figure 6. Figure 7 shows the 2D packing results according to the aforementioned 2D conversion approaches. Although 2D audio signals showed different and complex texture patterns between consecutive frames, we confirmed that the 2D signals had a horizontally strong directionality within a frame.

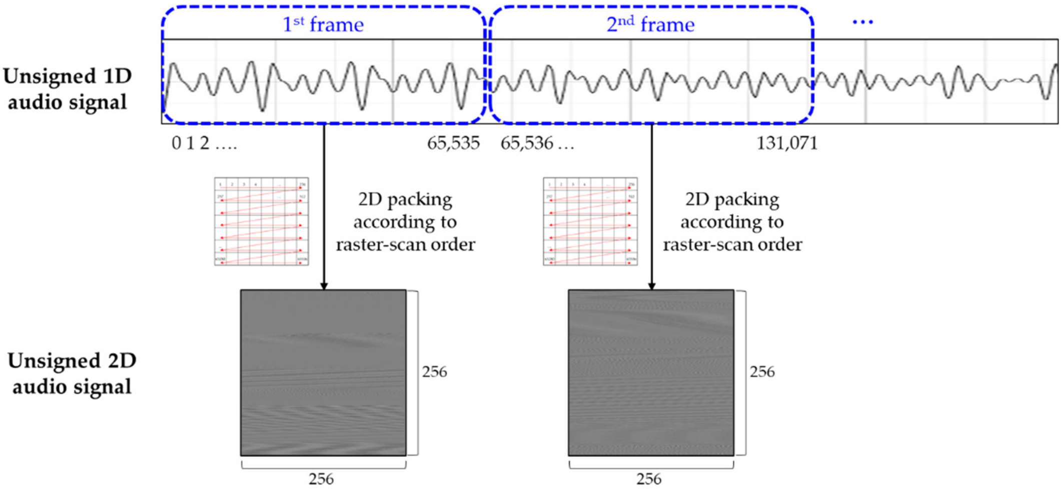
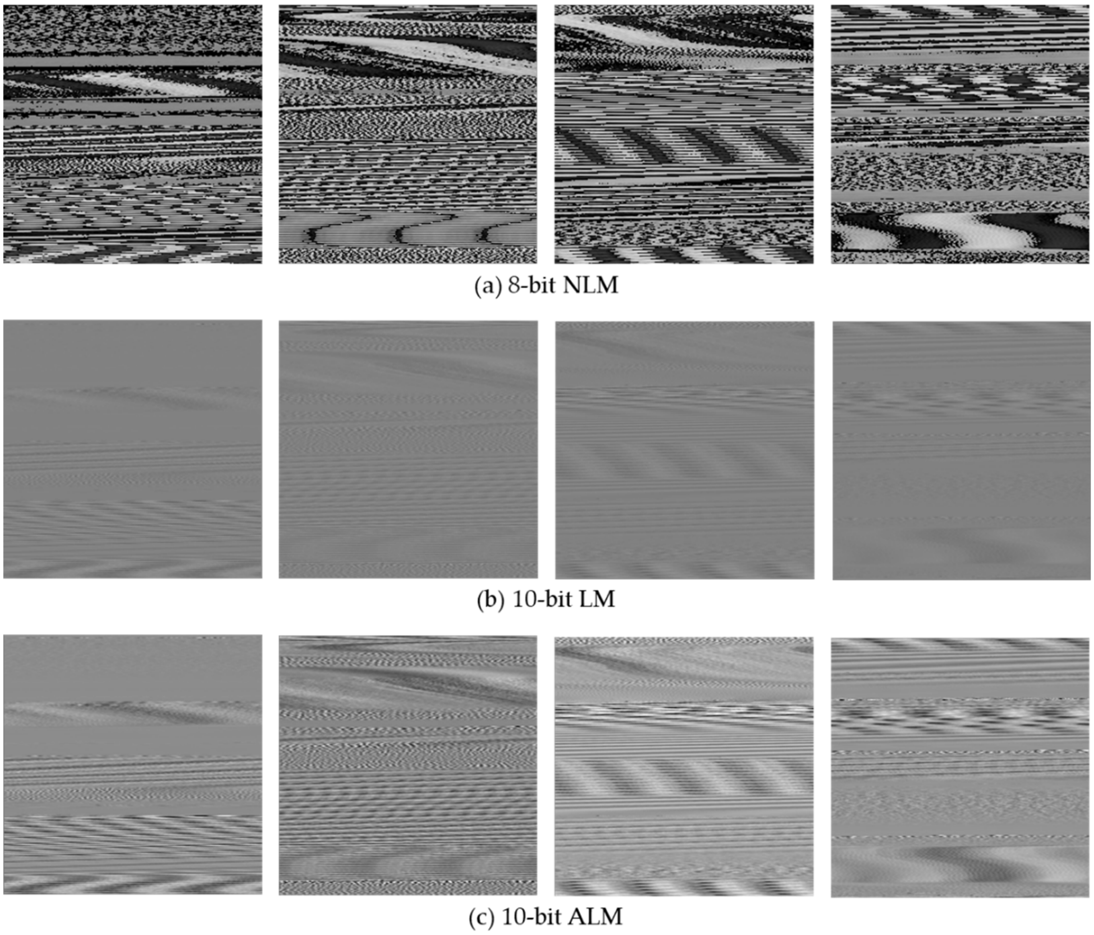


Figure 6. The 2D packing of 1D audio signals (256 × 256 spatial resolution).



(a) 8-bit NLM

(b) 10-bit LM

(c) 10-bit ALM

Figure 7. Visualization of consecutive 2D audio signals in the unit of frames (256 × 256 spatial resolution).

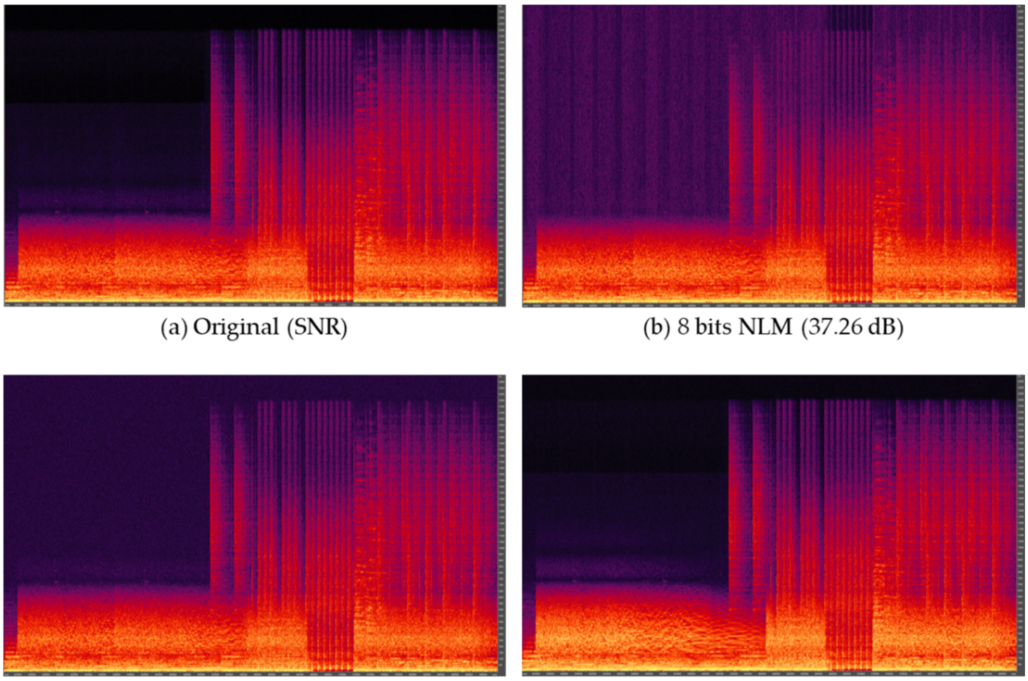
4. Experimental Results

In this study, we used an original audio signal (48 kHz, 16 bits) whose length was approximately 195 s and consisted of five music, five speech, and five mixed items, as described in Table 2. To compress a 2D audio signal using the VVC encoder, we first set the spatial resolution to 256 × 256. Because the original audio signal had 9,371,648 samples, we generated 143 frames with a 256 × 256 spatial resolution. As demonstrated in the spectrograms in Figure 8, ALM is superior other conversion methods in terms of SNR.

After generating 2D audio sequences with 143 frames, they were compressed by the VVC encoder under the experimental environments presented in Table 3.

Table 2. Description of audio signal items.

Category	Item Name	Description
Music	salvation	Classical chorus music
	te15	Classical music
	Music_1	Rock music
	Music_3	Pop music
	phi7	Classical music
Speech	es01	English speech
	louis_raquin_15	French speech
	Wedding_speech	Korean speech
	te1_mg54_speech	German speech
	Arirang_speech	Korean speech
Mixed	twinkle_ff51	Speech with pop music
	SpeechOverMusic_1	Speech with chorus music
	SpeechOverMusic_4	Speech with pop music
	HarryPotter	Speech with background music
	lion	Speech with background music


(a) Original (SNR)

(b) 8 bits NLM (37.26 dB)

(c) 10 bits LM (38.58 dB)

(d) 10 bits ALM (45.34 dB)

Figure 8. Spectrograms of the music_1 audio signal.

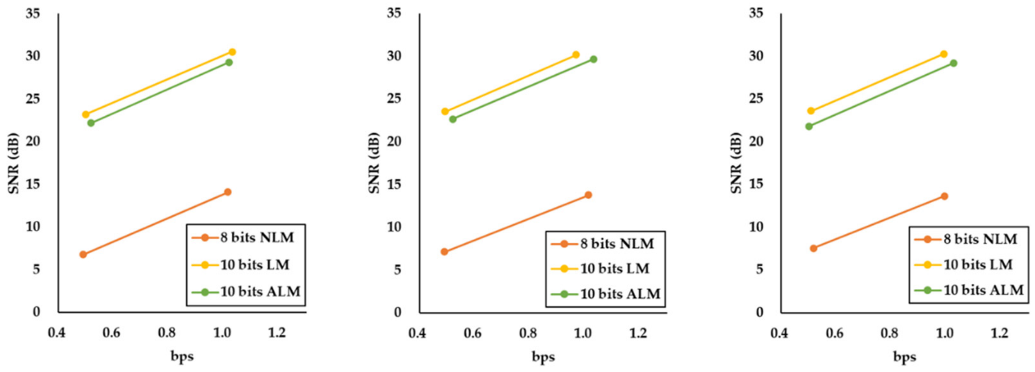
To evaluate the coding performance, we measured the SNR between the original and reconstructed 1D audio signals after decoding the compressed bitstream. Table 4 shows that NLM produced the worst coding performance due to the incorrectness of the signed 1D conversion. In addition, 10 bits LM had a higher SNR than ALM regardless of the encoding mode, as shown in Table 4 and Figure 9.

Table 3. Experimental environments for 2D audio encoding in the VVC.

	8 bits NLM	10 bits LM	10 bits ALM
Input bit-depth	8 bits	10 bits	10 bits
VTM version		VTM10.0	
Chroma format		YUV4:0:0	
Encoding mode		All-intra, Low-delay-B, Random-access	
Target bps		0.5 and 1.0	

Table 4. Experimental results according to the different conversion modes.

Encoding Mode	Target bps	8 bits NLM		10 bits LM		10 bits ALM	
		SNR (dB)	bps	SNR (dB)	bps	SNR (dB)	bps
All-intra	1.0	14.09	1.01	30.55	1.03	29.30	1.02
	0.5	6.78	0.49	23.21	0.50	22.18	0.52
Low-delay-B	1.0	13.78	1.01	30.17	0.97	29.66	1.03
	0.5	7.15	0.49	23.55	0.49	22.64	0.52
Random-access	1.0	13.63	0.99	30.28	0.99	29.21	1.03
	0.5	7.55	0.52	23.62	0.51	21.82	0.50



(a) All-intra (b) Low-delay-B (c) Random-access

Figure 9. Rate–distortion (RD) curves according to the different conversion methods.

We further investigated if the objective performance was well-reflected in the subjective quality. In terms of subjective assessment, we conducted a MUSHRA test [28] which is commonly used to evaluate subjective quality in the MPEG audio group. In this experiment, we used 10 audio items with a length of 10 s, and seven listeners participated. To enable comparison with the state-of-the-art audio coding method, the same audio signal was coded by USAC [29]. Because the encoded frames have different quantization parameters (QPs) according to the temporal layer of LD and RA configuration, SNR per frame can fluctuate between consecutive frames. Therefore, we performed MUSHRA tests under the AI configuration without QP variations. In the AI configuration, we set the target to 0.5 bps and conducted MUSHRA listening tests for the original, USAC, ALM, LM, and NLM. As shown in Figure 10, the subjective quality of 10 bits ALM was higher than that of 10 bits LM, which opposes the results in Table 4.

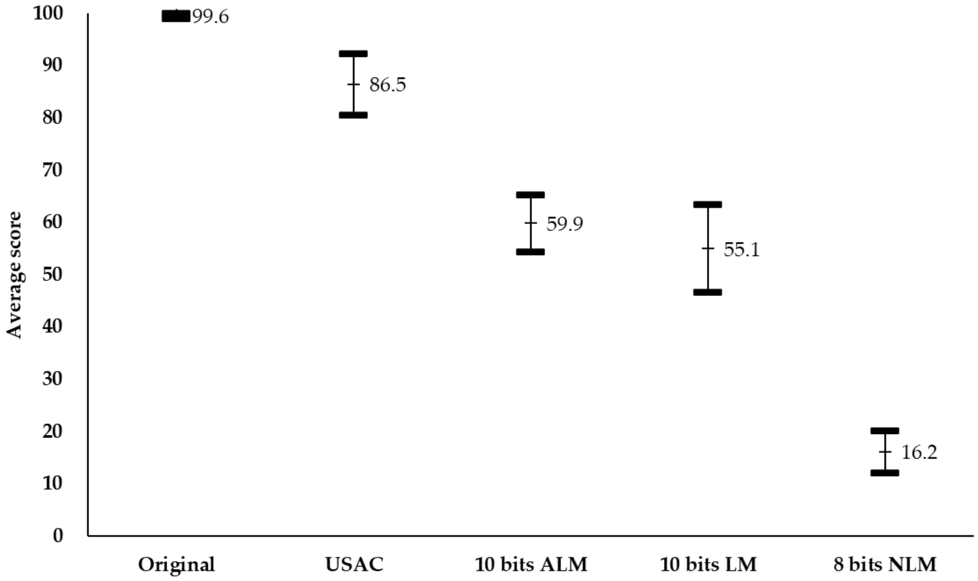


Figure 10. Scores of the MUSHRA listening test with a 95% confidence interval for different methods.

Although the average score of ALM or LM was lower than that of USAC, it was necessary to further investigate the possibility of researching 2D audio encoding with respect to 2D packing and video codec optimization. Firstly, the coding performance is strongly dependent on how the 1D signal is converted to a 2D signal. Because we used simple 2D packing in this study, it was necessary to consider a more elaborate 2D packing method suitable for video coding tools. For example, if we generate the 2D signals to efficiently perform the block based inter prediction, the coding performance of LD or RA configurations can be improved. Secondly, the current VVC has to transmit many unnecessary coding parameters due to the adoption of new tools, even though these tools are not used in the 2D audio encoding process. If a video encoder is optimized with core coding tools and light syntax structure, the coding performance will be improved.

5. Conclusions

In this paper, we proposed a pre-processing method to generate a 2D audio signal as an input of a VVC encoder, and investigated its applicability to 2D audio compression using the video coding scheme. To evaluate the coding performance, we measured both signal-to-noise ratio (SNR) and bits per sample (bps). Additionally, we conducted a MUSHRA test to evaluate subjective quality. The experimental results showed the possibility of researching 2D audio encoding using video coding schemes.

Author Contributions: Conceptualization, S.K. and D.J.; methodology, S.K. and D.J.; software, S.K.; validation, D.J., B.-G.K., S.B., M.L. and T.L.; formal analysis, S.K. and D.J.; investigation, S.K. and D.J.; resources, D.J.; data curation, S.B., M.L. and T.L.; writing—original draft preparation, S.K.; writing—review and editing, D.J. and B.-G.K.; visualization, S.K.; supervision, D.J. and B.-G.K.; project administration, D.J. and B.-G.K.; funding acquisition, S.B., M.L. and T.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the research of basic media content technologies, grant number 21ZH1200.

Acknowledgments: This work was supported by an Electronics and Telecommunications Research Institute (ETRI) grant funded by the Korean government. (21ZH1200, the research of basic media content technologies).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Bross, B.; Chen, J.; Liu, S.; Wang, Y. Versatile Video Coding Editorial Refinements on Draft 10. Available online: https://jvet.hhi.fraunhofer.de/ (accessed on 10 October 2020).
2. Sullivan, G.J.; Ohm, J.R.; Han, W.J.; Wiegand, T. Overview of the High Efficiency Video Coding (HEVC) Standard. *IEEE Trans. Circuits Syst. Video Technol.* **2012**, *22*, 1649–1668. [CrossRef]
3. ISO/IEC. *Information Technology—MPEG Audio Technologies—Part 3: Unified Speech and Audio Coding*; International Standard 23003-3; ISO/IEC: Geneva, Switzerland, 2012.
4. Bossen, F.; Li, X.; Suehring, K. AHG Report: Test Model Software Development (AHG3). In Proceedings of the 20th Meeting Joint Video Experts Team (JVET), Document JVET-T0003, Teleconference (Online), Ljubljana, Slovenia, 10–18 July 2018.
5. VVC Reference Software (VTM). Available online: https://vcgit.hhi.fraunhofer.de/jvet/VVCSoftware_VTM (accessed on 21 March 2021).
6. HEVC Reference Software (HM). Available online: https://vcgit.hhi.fraunhofer.de/jvet/HM (accessed on 21 March 2021).
7. Bossen, F.; Boyce, J.; Suehring, K.; Li, X.; Seregin, V. VTM Common Test Conditions and Software Reference Configurations for SDR Video. In Proceedings of the 20th Meeting Joint Video Experts Team (JVET), Document JVET-T2021, Teleconference (Online), 6–15 January 2021.
8. Auwera, G.; Seregin, V.; Said, A.; Ramasubramonian, A.; Karczewicz, M. CE3: Simplified PDPC (Test 2.4.1). In Proceedings of the 11th Meeting Joint Video Experts Team (JVET), Document JVET-K0063, Ljubljana, Slovenia, 10–18 July 2018.
9. Ma, X.; Yang, H.; Chen, J. CE3: Tests of Cross-Component Linear Model in BMS. In Proceedings of the 11th Meeting Joint Video Experts Team (JVET), Document JVET-K0190, Ljubljana, Slovenia, 10–18 July 2018.
10. Racapé, F.; Rath, G.; Urban, F.; Zhao, L.; Liu, S.; Zhao, X.; Li, X.; Filippov, A.; Rufitskiy, V.; Chen, J. CE3-Related: Wide-Angle Intra Prediction for Non-Square Blocks. In Proceedings of the 11th Meeting Joint Video Experts Team (JVET), Document JVET-K0500, Ljubljana, Slovenia, 10–18 July 2018.
11. Pfaff, J.; Stallenberger, B.; Schäfer, M.; Merkle, P.; Helle, P.; Hinz, T.; Schwarz, H.; Marpe, D.; Wiegand, T. CE3: Affine Linear Weighted Intra Prediction. In Proceedings of the 14th Meeting Joint Video Experts Team (JVET), Document JVET-N0217, Geneva, Switzerland, 19–27 March 2019.
12. Zhang, H.; Chen, H.; Ma, X.; Yang, H. Performance Analysis of Affine Inter Prediction in JEM 1.0. In Proceedings of the 2nd Meeting Joint Video Experts Team (JVET), Document JVET-B0037, San Diego, CA, USA, 20–26 February 2016.
13. Gao, H.; Esenlik, S.; Alshina, E.; Kotra, A.; Wang, B.; Liao, R.; Chen, J.; Ye, Y.; Luo, J.; Reuzé, K.; et al. Integrated Text for GEO. In Proceedings of the 17th Meeting Joint Video Experts Team (JVET), Document JVET-Q0806, Brussels, Belgium, 20–26 February 2016.
14. Jeong, S.; Park, M.; Piao, Y.; Park, M.; Choi, K. CE4 Ultimate Motion Vector Expression. In Proceedings of the 12th Meeting Joint Video Experts Team (JVET), Document JVET-L0054, Macao, China, 3–12 October 2018.
15. Sethuraman, S. CE9: Results of DMVR Related Tests CE9.2.1 and CE9.2.2. In Proceedings of the 13th Meeting Joint Video Experts Team (JVET), Document JVET-M0147, Marrakech, Morocco, 9–18 January 2019.
16. Chiang, M.; Hsu, C.; Huang, Y.; Lei, S. CE10.1.1: Multi-Hypothesis Prediction for Improving AMVP Mode, Skip or Merge Mode, and Intra Mode. In Proceedings of the 12th Meeting Joint Video Experts Team (JVET), Document JVET-L0100, Macao, China, 3–12 October 2018.
17. Ye, Y.; Chen, J.; Yang, M.; Bordes, P.; François, E.; Chujoh, T.; Ikai, T. AHG13: On Bi-Prediction with Weighted Averaging and Weighted Prediction. In Proceedings of the 13th Meeting Joint Video Experts Team (JVET), Document JVET-M0111, Marrakech, Morocco, 9–18 January 2019.
18. Zhang, Y.; Chen, C.; Huang, H.; Han, Y.; Chien, W.; Karczewicz, M. Adaptive Motion Vector Resolution Rounding Align. In Proceedings of the 12th Meeting Joint Video Experts Team (JVET), Document JVET-L0377, Macao, China, 3–12 October 2018.
19. Luo, J.; He, Y. CE4-Related: Simplified Symmetric MVD Based on CE4.4.3. In Proceedings of the 13th Meeting Joint Video Experts Team (JVET), Document JVET-M0444, Marrakech, Morocco, 9–18 January 2019.
20. Chen, J.; Ye, Y.; Kim, S. Algorithm description for Versatile Video Coding and Test Model 11 (VTM 11). In Proceedings of the 20th Meeting Joint Video Experts Team (JVET), Document JVET-T2002, Teleconference (Online), 10–18 July 2018.
21. Chien, W.; Boyce, J.; Chen, W.; Chen, Y.; Chernyak, R.; Choi, K.; Hashimoto, R.; Huang, Y.; Jang, H.; Liao, R.; et al. JVET AHG Report: Tool Reporting Procedure (AHG13). In Proceedings of the 20th Meeting Joint Video Experts Team (JVET), Document JVET-T0013, Teleconference (Online), 7–16 October 2020.
22. Fan, Y.; Sun, H.; Katto, J.; Ming'E, J. A fast QTMT partition decision strategy for VVC intra prediction. *IEEE Access* **2020**, *8*, 107900–107911. [CrossRef]
23. Jin, Z.; An, P.; Yang, C.; Shen, L. Fast QTBT partition algorithm for intra frame coding through convolutional neural network. *IEEE Access* **2018**, *6*, 54660–54673. [CrossRef]
24. Yang, H.; Shen, L.; Dong, X.; Ding, Q.; An, P.; Jiang, G. Low-complexity CTU partition structure decision and fast intra mode decision for versatile video coding. *IEEE Trans. Circuits Syst. Video Technol.* **2019**, *30*, 1668–1682. [CrossRef]
25. Fast Intra Mode Decision Algorithm for Versatile Video Coding. Available online: https://doi.org/10.1109/TMM.2021.3052348 (accessed on 5 May 2021).
26. Park, S.; Kang, J. Fast affine motion estimation for versatile video coding (VVC) encoding. *IEEE Access* **2019**, *7*, 158075–158084. [CrossRef]

27. ITU-T G.711 Pulse Code Modulation (PCM) of Voice Frequencies. Available online: http://handle.itu.int/11.1002/1000/911 (accessed on 21 March 2021).
28. International Telecommunication Union. Method for the Subjective Assessment of Intermediate Sound Quality (MUSH-RA). ITU-T, Recommendation BS 1543-1. Available online: https://www.itu.int/pub/R-REC/en (accessed on 1 January 2016).
29. Beack, S.; Seong, J.; Lee, M.; Lee, T. Single-Mode-Based Unified Speech and Audio Coding by Extending the Linear Prediction Domain Coding Mode. *ETRI J.* **2017**, *39*, 310–318. [CrossRef]