LiDAR Point Cloud Compression by Vertically Placed Objects based on Global Motion Prediction

Junsik Kim1, Seongbae Rhee1, Hyukmin Kwon2, and Kyuheon Kim1

1Department of Electronic Engineering, Kyung Hee University, Korea
2Media Research Division, Electronics & Telecommunications Research Institute, Korea

Corresponding author: Kyuheon Kim (e-mail: kyuheonkim@khu.ac.kr).

This research was supported by the MSIT (Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (Grant number: IITP-2021-0-02046) and (Grant number: 2020-0-00452) supervised by the Institute of Information & communications Technology Planning & evaluation (IITP).

ABSTRACT A point cloud acquired through a Light Detection And Ranging (LiDAR) sensor can be illustrated as a continuous frame with a time axis. Since the frame-by-frame point cloud has a high correlation between frames, a higher compression efficiency can be obtained by using an inter-prediction scheme, and for this purpose, Geometry-based Point Cloud Compression (G-PCC) in the Moving Picture Expert Group (MPEG) opened Inter-Exploratory Model (Inter-EM) which experiments on continuous LiDAR based point cloud frames compression through inter-prediction. The points of the LiDAR based point cloud have two different types of motion: global motion brought about by a vehicle with a LiDAR sensor and local motion generated by an object e.g., a walking person. Thus, Inter-EM consists of a compression structure in terms of both global and local motion, and the Inter-EM's global motion compensation technology increases the compression efficiency via a single matrix describing the global motion of points. However, this is difficult to predict with a single matrix, which causes imprecise global motion estimation since the objects in a LiDAR-based point cloud show variable global motion according to object characteristics such as shape and position. Therefore, this paper proposes a global motion prediction and compensation scheme that considers the characteristics of objects for efficient compression of LiDAR-based point cloud frames. The proposed global motion prediction and compensation scheme achieved higher overall gain in terms of the Bjontegaard-Delta-rate (BD-rate), and effectively compressed the LiDAR-based sparse point cloud.

INDEX TERMS Geometry-based point cloud compression, global motion prediction, point cloud classification, global motion information encoding.

I. INTRODUCTION

A point cloud is 3D content that represents the surface of the content with a large number of points. Each point in a point cloud consists of location information expressed in x, y, and z coordinate systems, as well as reflectance and RGB attributes corresponding to its location. The point cloud can be captured by a Light Detection And Ranging (LiDAR) sensor [1]-[3] and a set of fixed RGBD cameras [4]-[8]. This camera and sensor-based point cloud has an advantage in processing real world objects compared to the polygon-based mesh method, which is used to handle virtual 3D objects. As a consequence of this characteristic, point clouds are gaining attention as a next-generation 3D content expression method in various fields such as autonomous driving [9] and Augmented, Virtual, Mixed Reality (AR/VR/MR) [10][11]. A point cloud is visualized by hundreds of thousands or millions of points with 3D coordinate and attribute information, which requires a larger number of bits compared to 2D media; thus compression is essential for transmission and storage. The Moving Picture Expert Group (MPEG) under ISO/IEC JTC1, an international standardization organization, classifies point cloud content into three categories, Categories 1 [12], 2 [13] and 3 [14], according to their characteristics for compression; MPEG also provides standards for two compression methods, Geometry-based Point Cloud Compression (G-PCC) [15] and Video-based Point Cloud Compression (V-PCC) [16]. V-PCC
In general, 2D video compression technologies have used a "motion" concept, for example, it analyzed and searched similarity tendencies of pixels in a certain neighboring area such as a macro-block. This macro-block-based motion determines a local property rather than a global one. Since a 2D video sequence is densely composed of pixels, this local motion can provide enough redundancy information for compression, which could be applied to a sparse point cloud such as that captured by a LiDAR sensor in a vehicle. To reflect redundancy information, it is necessary to define a larger 3D macro-cubic because of its sparse point composition. However, this larger macro-cubic-based motion search results in a huge computing complexity, such as widening the search range or increasing the number of points in the search. To use LiDAR-based sparse point cloud compression in particular low-power processors such as vehicles, this increased computing cost in terms of motion search with a computational complexity of $O(n^3)$ is fatal. Thus, motion occurring in the LiDAR-based sparse point cloud is better analyzed in terms of both local and global motion, where local motion is the movement itself of an individual dynamic object, and global motion is determined via a static object such as LiDAR movement. On the basis of this property, Inter-EM of G-PCC [18] designed a compression structure that considers both local and global motion to effectively compress the characteristics of sparse point clouds, as shown in Figure 1, where $PC_{t}$ is the current point cloud frame and $PC_{t-1}$ the previous point cloud frame.

This paper proposes a global motion prediction and compensation scheme that considers object characteristics for efficient compression of LiDAR-based point cloud frames. This paper is organized as follows. Related techniques associated with 3D motion prediction are reviewed in Chapter II, and Chapter III introduces the global motion estimation and compensation scheme proposed in this paper. In Chapter IV, the results obtained using the proposed technology are compared with ones with the current Inter-EM technologies developed by MPEG. Finally, Chapter V concludes this paper and discusses future work.

II. Related Works

In general, 2D video compression technologies have used a "motion" concept, for example, it analyzed and searched similarity tendencies of pixels in a certain neighboring area such as a macro-block. This macro-block-based motion determines a local property rather than a global one. Since a 2D video sequence is densely composed of pixels, this local motion can provide enough redundancy information for compression, which could be applied to a sparse point cloud such as that captured by a LiDAR sensor in a vehicle. To reflect redundancy information, it is necessary to define a larger 3D macro-cubic because of its sparse point composition. However, this larger macro-cubic-based motion search results in a huge computing complexity, such as widening the search range or increasing the number of points in the search. To use LiDAR-based sparse point cloud compression in particular low-power processors such as vehicles, this increased computing cost in terms of motion search with a computational complexity of $O(n^3)$ is fatal. Thus, motion occurring in the LiDAR-based sparse point cloud is better analyzed in terms of both local and global motion, where local motion is the movement itself of an individual dynamic object, and global motion is determined via a static object such as LiDAR movement. On the basis of this property, Inter-EM of G-PCC [18] designed a compression structure that considers both local and global motion to effectively compress the characteristics of sparse point clouds, as shown in Figure 1, 2, where $PC_{t}$ is the current point cloud frame and $PC_{t-1}$ the previous point cloud frame.

![Figure 1. Compression structure of the Inter-EM](image)

To increase the compression efficiency of an input point cloud frame, Inter-EM compensates the $PC_{t-1}$ frame in terms of global and local motion for encoding the $PC_{t}$ frame as follows [19]-[22]. Firstly, a global motion estimation and compensation module calculates the global motion of the point cloud according to the movement of the LiDAR sensor and then outputs the Global Motion Compensated Point Cloud ($GCP_{t-1}$), which reduces the range of local motion and increases the similarity of local motion vectors. Secondly, the local motion estimation is performed in both the $GCP_{t-1}$ and $PC_{t-1}$ frames, which provides more samples for a suitable motion search and increases the compression efficiency. This
local motion estimation is performed in terms of cubic units of a predefined size, and local motion vectors are compressed with Context-Based Adaptive Binary Arithmetic Coding (CABAC) [23] as arithmetic coding. The Local Motion Compensated Point Cloud \( (LCP_{t-1}) \), the result of local motion compensation, is used to select and reduce the context of the \( PC_t \) frame in a geometry encoding, and these generated contexts are compressed with CABAC. Inter-EM's compression structure is effective in handling sparse point clouds, but still has limitations in estimating more precise global and local motions because it does not consider the object characteristics in the sparse point cloud.

Local and global motions are respectively obtained by each dynamic and static object in a point cloud, where local motion would be different depending on dynamic objects, and global motion is the same as individual LiDAR sensor movement. However, global motion appears to be different depending on the characteristics of static objects, as shown in Fig. 2.

As shown in Fig. 2, global motion from the road, building, vehicle and street tree objects are represented by yellow and white circles, where the motion in white shows a constant motion, while the motion in yellow exhibits random directions. This is because static objects placed vertically to a road such as buildings, pillars, and street trees displayed in white are perpendicular to the direction of the LiDAR sensor, and thus produce better LiDAR movement. However, static objects placed horizontally such as roads shown in yellow are not suitable for providing global motion produced from LiDAR movement. On the basis of the above analysis, this paper proposes an efficient compression architecture that classifies horizontally placed objects and vertically placed objects to a LiDAR sensor, and calculates the global motion by using those vertically placed objects.

### III. VERTICALLY PLACED OBJECTS BASED ON GLOBAL MOTION PREDICTION FOR GEOMETRY-BASED POINT CLOUD COMPRESSION

As explained in the previous chapter, it was found that compression of a sparse point cloud is more efficient when using both global and local motion, especially when static objects are placed vertically to produce more precise global motion information. Thus, this paper proposes the compression architecture that uses Vertically Placed Objects (VPO) based on Global Motion Prediction considering the characteristics of static objects in a sparse point cloud, as described in Fig. 3.

The proposed efficient global motion estimation and compensation methods are shown in red in Fig. 3. When point cloud frames \( (PC_t, PC_{t-1}) \) are taken as input, VPO point clouds \( (VO_t, VO_{t-1}) \) are obtained using the Histogram-based Point Cloud Classification module from \( PC_t \) and \( PC_{t-1} \). Histogram-based Point Cloud Classification module produces a classification threshold and classifies the Horizontally Placed Objects (HPO) and VPO based on specific heights obtained through histogram analysis. The produced \( VO_t \) and \( VO_{t-1} \) frames are used to calculate a global motion matrix in the Global Motion Estimation module. The global motion matrix and the classification threshold are encoded using a differential compression method in the Global Motion Information Encoding module, which generates encoded motion information. The Global Motion Compensation module generates a compensated point cloud \( GCP_{t-1} \) by applying the global motion matrix to \( PC_{t-1} \). Under the compression conditions using both global and local motion, \( GCP_{t-1} \) can be utilized by the Local Estimation and
Compensation module to generate $\text{LCP}_{t-1}$, which is then used to increase the compression efficiency of $\text{PC}_t$ in the Geometry Encoding module. Otherwise the Geometry Encoding module utilizes $\text{GCP}_{t-1}$ when considering only global motion. The Geometry Encoding module with the same encoding process as Inter-EM [18] produces encoded geometry, which is compressed using Arithmetic Coding. The decompression structure of the proposed VPO Global Motion Prediction is performed in reverse order of the compression, as shown in Fig. 4 where $\text{PC'}_t$, $\text{PC'}_{t-1}$, $\text{GCP'}_{t-1}$ and $\text{LCP'}_{t-1}$ indicate the decoded point cloud.

The Arithmetic Decoding module outputs the encoded motion information, encoded geometry, and decoded local motion vectors. The Encoded geometry of the intra frame is input to the Geometry Decoding module for the decoded point cloud. For the encoded geometry of the inter frames, $\text{GCP'}_{t-1}$ or $\text{LCP'}_{t-1}$ can used depending whether global motion or both global and local motion are applied. The proposed VPO Global Motion predicts global motion based on the classified VPO, which is then verified for a more accurate global motion matrix in subsequent chapters.

A. HISTOGRAM-BASED POINT CLOUD CLASSIFICATION

The VPO-based Global Motion Prediction proposed in this paper classifies HPO and VPO in an input point cloud, and searches for the global motion in the VPO. A conventional classification of a point cloud [24]-[27] requires a higher computational complexity than a two-dimensional image because it is necessary to analyze the distribution of huge point data in a three-dimensional space. However, since the LiDAR-based sparse point cloud is utilized in low-power devices such as vehicles, point cloud classification technology with low computational complexity is required. To effectively classify the input point cloud with lower-computational complexity, this paper proposes a Histogram-based Point Cloud Classification.

A LiDAR-based sparse point cloud is obtained through a LiDAR sensor mounted on a vehicle traveling along the HPO, as shown in Fig. 5. The HPO are represented in the form of a 2D plane placed at a specific height in the point cloud. With respect to a Cartesian coordinate system, the HPO can be expressed as the XY plane located at a specific height on the Z-axis, and the VPO as the XZ or YZ plane. The shape of this plane is the same regardless of the slope of the HPO on which the vehicle is traveling, since the vehicle also tilts while going up or down the slope. Additionally, the sloped road before or after the vehicle enters the slope can be considered a VPO. Since LiDAR sensors use a constant sampling frequency, the sensor will have the same density of points on the surface of the HPO as on the VPO. Thus, the HPO has the highest histogram value when a histogram is generated in terms of the Z-axis values of the input points. On the basis of this observation, this paper classifies the HPO and VPO using this histogram in terms of the Z-axis values of the input point cloud.

The Z-axis-based histogram $h[k]$ utilized in this paper is calculated as follows:

$$k = \text{round}(z_p/C), p(x,y,z) \in \text{PC}$$  (1)
where $z_p$ represents the z-axis value of a point in the input point cloud ($PC$), $k$ is the reference value of the Z-axis-based histogram ($h[k]$) obtained by dividing the scale factor ($C$) by $z_p$, and $N_k$ is the number of $p$ with the same $k$. The scale factor $C$ is used to consider distribution characteristics of the input point cloud, such as the sampling frequency of the LiDAR sensor and the error range of the acquisition points for compensating for HPO placement variation in terms of the Z-axis.

As shown in Fig. 6, the maximum value at a specific $k$ value in the $h[k]$ obtained from (2) can be considered as HPO. However, $k$ values adjacent to the maximum value of $h[k]$ can also provide HPO because of the shape of HPO surface and the noise generated during the capture of the LiDAR-based point cloud. This can cause a large number of HPO points to be incorrectly considered as a VPO point cloud. Thus, this paper proposes a method for generating a classification threshold using the gradient of $h[k]$ as shown in Fig. 7, which can prevent misclassification of HPO and VPO.

Figure 7 shows the proposed gradient based classification method, where Fig. 7 (a) indicates the maximum value of the histogram. As described in Fig. 5, HPO and VPO can be expressed in terms of the XY plane and the XZ or YZ plane, respectively. Thus, VPO appears as a gentle curve with a small gradient in the Z-axis-based histogram, which is clearly shown for the region close to HPO. In order to analyze the Z-axis histogram based on VPO, this paper proposes a histogram gradient-based classification method that calculates the classification threshold through continuously small gradients appearing after (a).

$$g[k] = \begin{cases} g[k-1] + 1, & \text{if } abs \left( \frac{h[k]-h[k-1]}{c} \right) < \delta, k > k_{max} \\ 0, & \text{otherwise} \end{cases}$$

In equation (3), $k_{max}$ is the $k$ value when $h[k]$ has its maximum value, $g[k]$ is a function that stores the number of successive small $h[k]$ gradients at $k$ values above $k_{max}$. $\delta$ is small gradient factor obtained by experimentally and $C$ is the Z-axis scaling factor. To differentiate the HPO and VPO, $g[k]$ is increased by adding 1 to $g[k-1]$ when the gradient of $h[k]$ is less than $\delta$, which makes the number of consecutive $h[k]$ gradients smaller than $\delta$ to be stored in $g[k]$. The classification threshold $\sigma$ between the HPO and VPO is obtained by analyzing the $g[k]$ as follows:

$$R = \{ r \mid g[r] > \gamma \}$$

$$r_{min} = \arg\min(R)$$

$$\sigma = \text{round}(r_{min} \times C)$$

$$VO = \{ (x,y,z) \mid z_p > \sigma, (p(x,y,z) \in PC) \}$$

where $R$ is a set of $r$ values such that $g[r]$ is greater than a continuity factor $\gamma$ obtained experimentally, and $r_{min}$ indicates the minimum value $x$ among elements of $R$. Since the $r$ values of $g[r]$ are obtained by dividing the Z-axis coordinate of the input point cloud by the scale factor $C$, $\sigma$ is calculated by multiplying $r_{min}$ by $C$. The proposed classification is defined in (7), where the VPO point cloud (VO) is a set of points $p(x,y,z)$ in the input point cloud ($PC$) with a Z-axis value $z_p$ greater than $\sigma$.

As described in Chapter II, the VPO can show more precise global motion compared to HPO, and thus it is important to classify the VPO of the input point cloud. The proposed VPO classification method in (7) can improve the accuracy of the global motion matrix, which will improve the compression efficiency.

**B. GLOBAL MOTION ESTIMATION AND COMPENSATION**

The proposed global motion estimation and compensation in the compression structure shown in Fig. 3 uses VPO to increase the compression efficiency. Since point cloud data are captured by a Vehicle equipped with LiDAR sensors, global motion can be expressed in terms of rotation and translation (as linear transformations) of a vehicle, which can be
expressed as a 4*4 matrix that considers rotation and translation as follows:

\[
GM_t = \begin{bmatrix}
R_{00} & R_{01} & R_{02} & T_x \\
R_{10} & R_{11} & R_{12} & T_y \\
R_{20} & R_{21} & R_{22} & T_z \\
0 & 0 & 0 & 1
\end{bmatrix}
\]  
(8)

where, \( R \) is the rotation matrix, \( T \) is the translation matrix, and \( GM_t \) is a global motion matrix by analyzing the transition between \( VO_t \) and \( VO_{t-1} \) using conventional 3D object motion estimation method [28]. \( GM_t \) is applied to the previous point cloud (\( PC_{t-1} \)) and helps minimize the difference from the current point cloud (\( PC_t \)), which increases the efficiency of arithmetic coding. As described in Chapter II, global motion can be obtained from the VPO point cloud rather than the HPO, and thus \( GM_t \) in (8) will be applied only to the VPO point cloud (\( VO_{t-1} \)) to generate the compensated point cloud (\( GCP_{C_{t-1}} \)). This is because it is obvious that the HPO from \( PC_t \) and the \( GCP_{C_{t-1}} \) will be significantly different, as shown in Fig. 8, when this \( GM_t \) is applied to the HPO point cloud.

An example of global motion compensation is shown in Fig. 8, where (a) displays an input point cloud (\( PC_t \)), (b) a previous point cloud (\( PC_{t-1} \)), (c) a compensated point cloud where \( GM_t \) is applied only to \( VO_{t-1} \), and (d) a compensated point cloud where \( GM_t \) is applied to all points of \( PC_{t-1} \). Also (e) shows the results of overlapping (a) and (c), and (f) the result of overlapping (a) and (d). GCHPO and GCVPO in Fig. 8 represent the globally compensated VPO and HPO, respectively, as displayed in yellow. As shown in Fig. 8, (f)’s HPO and (a)’s HPO in (e) are located in a similar space, but contrarily (d)’s GCHPO and (a)’s HPO in (f) have a spatial difference because (b) has been compensated into (d) by \( GM_t \). Thus, this paper proposes global motion compensation as follows:

\[
\hat{p}(x, y, z) = \begin{cases} 
GM_t \ast p(x, y, z), & \text{if } p(x, y, z) \in VO_{t-1} \\
p(x, y, z), & \text{otherwise}
\end{cases}
\]  
(9)

where \( p(x, y, z) \) is a point of \( PC_{t-1} \) and \( \hat{p}(x, y, z) \) is a point of \( GCP_{C_{t-1}} \). \( GM_t \) is applied to \( p(x, y, z) \) to produce \( GCP_{C_{t-1}} \) only when \( p(x, y, z) \) belongs to \( VO_{t-1} \).

The proposed global motion estimation and compensation effectively compresses the LiDAR point clouds using the VPO-based global motion of the LiDAR sensor. Since Inter-EM supports the local motion compression method for individual objects that are not clearly recognized as the VPO, this could provide higher compression when the proposed global motion is combined with a local motion. Thus, this paper proposes a local motion-based compression method using the proposed global motion scheme. When the global motion is applied to the input point clouds, there are two reference point clouds: an input point cloud (\( PC_{t-1} \)) and a global motion compensated point cloud (\( GCP_{C_{t-1}} \)) available for local motion; these are used to increase the number of points utilized for local motion estimation. When \( PC_{t-1} \) and \( GCP_{C_{t-1}} \) from both HPO and VPO are used for local motion search, the motion search probability in HPO can be increased, as shown in Fig. 9. Thus, the global motion compensation method when using local motion compression is proposed as follows:

\[
\hat{p}(x, y, z) = GM_t \ast p(x, y, z), \quad (p(x, y, z) \in PC_{t-1})
\]  
(10)

where \( \hat{p}(x, y, z) \) is a point of the \( GCP_{C_{t-1}} \), \( GM_t \) is applied to all points of the \( PC_{t-1} \) when local motion estimation is used, as described in (10). The \( GCP_{C_{t-1}} \) generated through the proposed technology is an input for the Geometry Encoding module or the Local Motion Estimation module to increase the compression efficiency of \( PC_t \), and each module is performed according to the compression procedure for Inter-EM, as shown in Fig. 4. Consequently, the global motion compensation technology presented in equations (9) and (10) can be differently applied depending on whether local motion compression is used or not. Whether to use local motion compression is determined by the encoder according to the
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2022.3148252, IEEE Access

GLOBAL MOTION COMPENSATION

VOLUME XX, 2021

D. GLOBAL MOTION INFORMATION DECODING AND GLOBAL MOTION COMPENSATION

The decompression structure of the of the proposed VPO-based Global Motion Prediction reconstructs the decoded Global Motion Compensated Point Cloud (GCPC\(_{t-1}\)) using global motion information, as shown in Fig. 4, where the Arithmetic Decoding module produces encoded global motion information, decoded local motion vectors, and encoded geometry. As explained in Section C, the encoded global motion information contains the encoded global motion matrix and encoded classification threshold, of which each is decoded as follows:

\[
\sigma'_t = \Delta \sigma'_t + \sigma'_{t-1}
\]

\[
GM'_t = \Delta GM'_t + GM'_{t-1}
\]

where \(\sigma'_t\) is the decoded classification threshold and \(GM'_t\) is decoded global motion matrix. The Global Motion Information Decoding module in Fig. 4 utilizes both encoded global motion information (\(\Delta GM'_t\)) and the previous global motion information (\(\sigma'_{t-1}, GM'_{t-1}\)) to produce the current global motion information (\(\sigma'_t, GM'_t\)), which are used for the Global Motion Compensation module to generate GCPC\(_{t-1}\). Since the proposed VPO-based Global Motion Prediction method is applied only to global motion or both global and local motion as described in Section B, GCPC\(_{t-1}\) will vary depending on whether or not local motion prediction is used as follows:

\[
\hat{p}'(x, y, z) = \begin{cases} 
GM'_t \ast \hat{p}'(x, y, z), & \text{if } z_{p'} > \sigma'_t, \hat{p}'(x, y, z) \in PC'_{t-1} \\
p'(x, y, z), & \text{otherwise}
\end{cases}
\]

where \(\hat{p}'(x, y, z)\) and \(\hat{p}'(x, y, z)\) are a point of GCPC\(_{t-1}\). \(p'(x, y, z)\) is a point of \(PC'_{t-1}\), and \(z_{p'}\) is the z-axis value of the \(p'(x, y, z)\). When local motion prediction is not used, global motion should be applied only to \(VO_{t-1}\), and thus the decoder classifies \(VO_{t-1}\) based on the conditions in (15), such as \(z_{p'}\) greater than \(\sigma'_t\). This is because \(VO_{t-1}\) is not transmitted to the decoder. Additionally, in the case of local motion prediction being used, \(GM'_t\) is applied to all points as

![Flowchart of the adaptive global motion compensation based on local motion compression condition](Image 97x613 to 502x777)

\[\Delta \sigma_t = \sigma_t - \sigma_{t-1}\]

\[\Delta GM_t = GM_t - GM_{t-1}\]
described in (16). The $\text{GCPC}'_{t-1}$ generated by the Global Motion Compensation module is delivered into the Geometry Decoding module or Local Motion Compensation module as shown in Fig. 4, which each operates in the same way as in the Inter-EM procedure.

This chapter proposed a global motion prediction structure suitable for LiDAR-based sparse point clouds through the proposed VPO-based Global Motion Prediction method. The compression effectiveness of the proposed technology verifies the performance of the VPO-based Global Motion Prediction method, as will be presented in Chapter IV.

IV. EXPERIMENTAL RESULTS

This chapter verifies the performance of the proposed VPO-based Global Motion Prediction in terms of the Histogram-based Point Cloud Classification, Global Motion Estimation and Compensation, and Global Motion Information Encoding modules, and evaluates the compression efficiency of the LiDAR point clouds. These proposed modules were applied to the Inter-EM v2.0 [18] based on the G-PCC reference software v7 [29] for the LiDAR-based point cloud MPEG test sequences “Ford-01-q-1mm” (1500 frames), “Ford-02-q-1mm” (1500 frames), “Ford-03-q-1mm” (1500 frames), “qxadas-junction-approach” (74 frames), “qxadas-junction-exit” (74 frames), “qxadas-motorway-join” (500 frames) and “qxadas-navigating-bends” (300 frames) under the Common Test Conditions (CTC) [30] in G-PCC. The parameters scale factor $C$, small gradient factor $\delta$, and continuity factor $\gamma$ used in the experiment are determined according to characteristics of the input point cloud, such as resolution, density, and type of object. Since the geometry precision of all input point clouds used in this experiment is 18 bits, 100, 3 and 3 were used for $C$, $\delta$ and $\gamma$, respectively, in order to obtain optimal results.

A. RESULTS OF THE HISTOGRAM-BASED POINT CLOUD CLASSIFICATION

This paper proposed the Histogram-based Point Cloud Classification, which classifies the HPO and VPO in the input point cloud via histogram analysis to search for a precise global motion matrix.

Fig. 11 shows the results of classifying the input point cloud into HPO in the XY plane and VPO perpendicular to the HPO, where the former is represented as green and the latter as red. As shown in Fig. 11, the Histogram-based Point Cloud Classification proposed in this paper accurately classifies the HPO and VPO.

B. RESULTS OF THE GLOBAL MOTION ESTIMATION AND COMPENSATION

As described in Chapter II, the global motion estimation module of Inter-EM [18] is found to be inaccurate due to its use of the unclassified previous point cloud $PC_{t-1}$. To increase the accuracy of global motion estimation, the proposed Global Motion Estimation is performed based on the VPO for improved accuracy of the global motion matrix, and the obtained global motion matrix is applied to the previous point cloud $PC_{t-1}$ in the Global Motion Compensation module.

![FIGURE 12. Example of global motion between the current point cloud ($PC_{t}$) and previous point cloud ($PC_{t-1}$)](image)

Fig. 12 (a) and (b) show the input point cloud frames $PC_{t}$ and $PC_{t-1}$ in green and yellow. The global motion can be identified in Fig. 12 (c) by overlapping those two frames. This global motion is estimated by Inter-EM and the proposed global motion estimation methods, as shown in Fig. 13.

The estimated global motion by Inter-EM and the proposed global motion estimation is shown in Fig. 13, where (a) displays the point cloud $PC_{t}$, (b) Inter-EM's $\text{GCPC}'_{t-1}$ (purple), (c) the proposed $\text{GCPC}'_{t-1}$ (red), (d) overlapping comparison between $PC_{t}$ and Inter-EM compensated point cloud

![FIGURE 13. Global motion estimation and compensation results of Inter-EM and the proposed technology](image)
and the proposed compensated point cloud $GCPC_{t-1}$, and (e) overlapping comparison between $PC_t$ and
the proposed compensated point cloud $GCPC_{t-1}$. As demonstrated in (d) and (e) of Fig. 13, the Inter-EM
compensated point cloud is seen to have more outliers from the input point cloud $PC_t$ with the proposed global motion estimation method. This is because Inter-EM’s global motion estimation utilizes the HPO for global motion estimation, which resulted in inaccurate global motion. Thus, the more accurate global motion estimation proposed in this paper improves the compression efficiency, which is verified in Section D.

C. RESULTS OF THE GLOBAL MOTION INFORMATION ENCODING

The Global Motion Information Encoding module performs differential encoding to increase the compression efficiency of global motion information in the arithmetic coder. Since global motion information such as the global motion matrix and classification threshold are highly similar between successive frames as mentioned in Chapter III-C, the resulting values of the differential encoding will be close, as shown in Fig. 14, where (a) is the histogram of the Ford-01-q-1mm’s global motion matrices, (b) is the histogram of the qnxadas-junction-approach’s global motion matrices, (c) is the histogram of the Ford-01-q-1mm’s encoded global motion matrices, and (d) is the histogram of the qnxadas-junction-approach’s encoded global motion matrices.

From Fig. 14, it is confirmed that the changing range of (c) and (d) is smaller than that of (a) and (b), and thus, the encoded global motion matrices are effectively compressed through arithmetic coding.

Additionally, the histogram results of the classification threshold and the encoded classification are shown in Fig. 15, where (a) is the histogram of the Ford-01-q-1mm’s classification thresholds, (b) is the histogram of the qnxadas-junction-approach’s classification thresholds, (c) is the

D. PERFORMANCE OF THE COMPRESSION EFFICIENCY USING THE PROPOSED VPO BASED ON GLOBAL MOTION PREDICTION

The VPO based on Global Motion Prediction proposed in this paper supports both lossy compression and lossless compression. The lossy compression result is verified in terms of the Bjontegaard-Delta-rate (BD-rate) [31] with both the Peak Signal-to-Noise Ratio (PSNR) and bitrate, as well as lossless compression in terms of the BPIP (Bits Per Input Point) ratio. The lossy compression condition utilizes predefined quantization parameters in CTC to obtain PSNR changes at various bitrates as shown in Table I, where the quantization parameters are divided into 6 steps from R01 to R06 for higher quality quantized point cloud.

| R1 to R6 | Position Quantization Scale | Attribute Quantization Parameter |
|----------|-----------------------------|-------------------------------|
| R01      | 0.001953125                 | 51                            |
| R02      | 0.00390625                  | 46                            |
| R03      | 0.015625                    | 40                            |
| R04      | 0.03125                     | 34                            |
| R05      | 0.125                       | 28                            |
| R06      | 0.25                        | 22                            |
In this table, the position quantization scale is used to change the number of points, and the attribute quantization parameter improves the accuracy of attribute values of each point, especially for the LiDAR-based sparse point clouds. Thus, the PSNR of the decoded point cloud is verified in terms of its geometry and attributes. The PSNR of the geometry is obtained using both point-to-point and point-to-plane distances [32], where the point-to-point distance is determined by the Euclidean distance between a reference point and a nearest point, and the point-to-plane distance by estimating the distance between the reference point and a projected point along a normal direction [32]. Here, the point-to-point distance PSNR is denoted by D1, and the point-to-plane distance PSNR by D2. BDBR is verified in terms of “BD-Geom rate”, “BD-Attr rate”, and “BD-total rate”, where the BD-Geom rate represents PSNR changes according to geometry bitstream change, the BD-Attr rate represents PSNR changes according to attribute bitstream change, and the BD-total rate represents PSNR changes according to total bitstream change.

The BPIP ratio used for performance verification under lossless conditions compares the number of bits used to represent one input point between the anchor and the proposed technology, and is expressed by as follows:

\[
BPIP = \frac{\text{size of output bitstream}}{N_{pc}} \quad (17)
\]

\[
\text{BPIP ratio} = \frac{BPIP_{\text{proposed}}}{BPIP_{\text{anchor}}} \times 100 \quad (18)
\]

where \(N_{pc}\) is the number of input points, BPIP is obtained by dividing \(N_{pc}\) by the size of the output encoded bitstream, and

| Sequence                  | BD-Geom rate | BD-Attr rate | BD-total rate | BPIP ratio [%] |
|---------------------------|--------------|--------------|---------------|----------------|
|                           | D1 | D2 | reflectance | D1 | D2 | reflectance | Geometry |
| Ford-01-q-1mm             | -6.8% -6.8% | 0.0% -6.1% | -6.1% -6.1% | -6.7% | 97.6% |
| Ford-02-q-1mm             | -7.2% -7.2% | 0.0% -6.5% | -6.5% -6.5% | -7.0% | 97.4% |
| Ford-03-q-1mm             | -3.1% -3.1% | 0.0% -2.8% | -2.8% -2.8% | -3.3% | 98.8% |
| qnxadas-junction-approach | -21.6% -21.5% | 0.0% -20.8% | -20.8% -20.8% | -21.3% | 89.3% |
| qnxadas-junction-exit     | -22.0% -21.9% | 0.0% -21.2% | -21.2% -21.2% | -21.7% | 88.8% |
| qnxadas-motorway-join     | -2.6% -2.6% | 0.0% -2.5% | -2.5% -2.5% | -2.7% | 98.4% |
| qnxadas-navigating-bends | -2.6% -2.6% | 0.0% -2.5% | -2.5% -2.5% | -2.7% | 98.8% |
| Average                   | -9.4% -9.4% | 0.0% -8.9% | -8.9% -8.9% | -9.5% | 95.9% |
the BPIP ratio is a comparison of the BPIP between the proposed technology and anchor as described in (18). The Histogram-based Point Cloud Classification module proposed in this paper improves the compression efficiency by increasing the accuracy of the global motion matrix, as explained in Chapter III-A. Thus, The Histogram-based Point Cloud Classification module is verified along with the Global Motion Estimation module. In addition, our results are verified experimentally using only global motion prediction to accurately track the effect of the improved global motion matrix, as shown in Table II.

The proposed Histogram-based Point Cloud Classification module and Global Motion Estimation module produced the maximum BD-Geom rate gain of -22.0% and average of -9.4% for lossy compression, and the maximum geometry BPIP ratio gain of 88.8% and average of 95.9% for lossless compression. Further, it is observed in Table II that the proposed method produced almost the same reflectance compression results because it uses the same reflectance compression method as Inter-EM.

The proposed method as shown in Fig. 16 is verified in terms of the Rate-Distortion(R-D) of the D1 PSNR to confirm changes of the PSNR according to changes of the bitrate under the compression condition of using only global motion prediction. As shown in Fig. 16, the proposed method in this paper in general produced higher PSNR at the same bitrate compared to the anchor Inter-EM. Thus, these test results show that the Histogram-based Point Cloud Classification module and Global Motion Estimation module proposed in this paper determine more accurate global motion compared to Inter-EM, thus increasing the compression efficiency.

### TABLE III
**COMPARISON OF COMPRESSION RESULTS WITH THE GLOBAL MOTION MATRIX APPLIED ONLY TO THE VPO AND TO ALL POINTS IN THE INPUT POINT CLOUD**

| Sequence               | BD-Geom rate | BD-Attr rate | BD-total rate | BPIP ratio [%] |
|------------------------|--------------|--------------|---------------|----------------|
|                         | D1           | D2           | D1            | D2            | reflectance | reflectance | Geometry |
| Ford-01-q-1mm          | -1.0%        | -1.0%        | 0.0%          | -0.9%         | -0.9%       | -0.9%       | 99.6%     |
| Ford-02-q-1mm          | -1.0%        | -1.0%        | 0.0%          | -0.9%         | -0.9%       | -0.8%       | 99.7%     |
| Ford-03-q-1mm          | -0.6%        | -0.6%        | 0.0%          | -0.5%         | -0.5%       | -0.5%       | 99.9%     |
| qxadas-junction-approach | -0.3%        | -0.2%        | 0.0%          | -0.2%         | -0.2%       | -0.2%       | 100.0%    |
| qxadas-junction-exit   | -0.4%        | -0.3%        | 0.1%          | -0.3%         | -0.3%       | -0.3%       | 100.0%    |
| qxadas-motorway-join   | 0.0%         | 0.0%         | -0.1%         | 0.0%          | 0.0%        | -0.1%       | 100.1%    |
| qxadas-navigating-bends | -0.2%        | -0.2%        | 0.0%          | -0.2%         | -0.2%       | -0.2%       | 100.0%    |
| Average                | -0.5%        | -0.5%        | 0.0%          | -0.5%         | -0.5%       | -0.4%       | 99.9%     

**FIGURE 17.** D1 Geometry R-D curve results of the proposed Global Motion Compensation module
The proposed Global Motion Information Encoding module is used or not, as described in Table IV. Since the Global Motion Information Encoding module does not change the PSNR, it is verified in terms of the size of the compressed global motion information bitstream depending on if the proposed Global Motion Information Encoding module is used or not, as described in Table IV.

The proposed Global Motion Information Encoding module is confirmed to reduce the size of compressed global motion information by up to 20.11% and on average by 14.07%. This very high compression gain considering it is in the context of lossless conditions.

V. CONCLUSION

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2022.3148252, IEEE Access

REFERENCES

[1] J. Zhou, H. Xu, Z. Ma, Y. Meng and D. Hui, "Sparse Point Cloud Generation Based on Turntable 2D Lidar and Point Cloud Assembly in Augmented Reality Environment," 2021 IEEE International Instrumentation and Measurement Technology Conference, pp. 1-6, 2021.
"A new inter mode for geometry coding in TMC3, document
On motion compensation for geometry coding in TM3, document
Description of Exploration Experiment 13.2 for G-PCC: on
PCC Test Model Category 13 V2, document N17519,
V-PCC codec description, document N00012,
ISO/IEC X. Yue, B. Wu, S. A. Seshia, K. Keutzer and A. L.
J. Fang et al., Simulating LIDAR point cloud for autonomous
driving using real-world scenes and traffic flows, Nov. 2018,
[online] Available: https://arxiv.org/abs/1811.07112.
3. X. Yue, B. Wu, S. A. Seshia, K. Keutzer and A. L.
Sangiovanni-Vincentelli, A LiDAR point cloud generator:
From a virtual world to autonomous driving, Mar. 2018,
[online] Available: https://arxiv.org/abs/1804.00103.
4. C. Tsai and S. Tsai, "Simultaneous 3D Object Recognition and
Pose Estimation Based on RGB-D Images," in IEEE Access,
vol. 6, pp. 28859-28869, 2018.
5. S. Orts-Escolano et al., “Holoportation: Virtual 3d
teleportation in realtime,” in Proceedings of the 29th Annual
Symposium on User Interface Software and Technology, New
York, USA: ACM, pp. 741–754, 2016.
6. A. Pujol-Miro, J. Ruiz-Hidalgo, and J. R. Casas, “Registration
of images to unorganized 3d point clouds using contour cues,”
in 25th European Signal Processing Conference, pp. 81–85,
August 2017.
7. C. Perra, F. Murgia, and D. Giusto, “An analysis of 3d point
cloud reconstruction from live field images,” in 2016 Sixth
International Conference on Image Processing Theory, Tools
and Applications, pp. 1–6, December 2016.
8. H. Fan H. Su and L. J. Guibas "A point set generation network
for 3D object reconstruction from a single image" Proc. IEEE
Conf. Comput. Vis. Pattern Recognit. pp. 605-613 Jul. 2017.
9. P. Sun, X. Zhao, Z. Xu, R. Wang and H. Min, "A 3D LiDAR
Data-Based Dedicated Road Boundary Detection Algorithm
for Autonomous Vehicles," in IEEE Access, vol. 7, pp. 29623-
29638, 2019.
10. Y. Ishikawa, et al., “Semantic Segmentation of 3D Point
Cloud to Virtually Manipulate Real Living Space,” 2019 12th
Asia Pacific Workshop on Mixed and Augmented Reality
(APMAR), Ikoma, Nara, Japan, pp. 1-7, 2019.
11. K. Ma, F. Lu and X. Chen, "Robust Planar Surface Extraction
from Noisy and Semi-Dense 3D Point Cloud for Augmented
Reality," 2016 International Conference on Virtual Reality
and Visualization, Hangzhou, pp. 453-458, 2016.
12. "Point Cloud Compression Test Model for Category 1 V0,
document N17223," ISO/IEC JTC1/SC29/WG11 MPEG,
Macau, China, Oct. 2017.
13. "PCC Test Model Category 2 V0, document N17248;"
ISO/IEC JTC1/SC29/WG11 MPEG, Macau, China, Oct. 2017.
14. "PCC Test Model Category 3 V0, document N17249;"
ISO/IEC JTC1/SC29/WG11 MPEG, Macau, China, Oct. 2017.
15. "G-PCC codec description, document N00011;" ISO/IEC
JTC1/SC29 WG7 3DG, Online, Oct. 2020.
16. "V-PCC codec description, document N00012;" ISO/IEC
JTC1/SC29 WG7 3DG, Online, Oct. 2020.
17. "PCC Test Model Category 13 V2, document N17519;"
ISO/IEC JTC1/SC29/WG11 MPEG, San Diego, CA, USA,
Apr. 2018.
18. "Description of Exploration Experiment 13.2 for G-PCC: on
Inter-prediction, document N00020;" ISO/IEC JTC1/SC29
WG7 3DG, Online, Oct. 2020.
19. "How to use a predictive set of points for geometry coding in
TMC3, document m42520;" ISO/IEC JTC1/SC29/WG11
MPEG, San Diego, CA, USA, Apr. 2018.
20. "On motion compensation for geometry coding in TM3,
document m42521;" ISO/IEC JTC1/SC29/WG11 MPEG, San
Diego, CA, USA, Apr. 2018.
21. "A new inter mode for geometry coding in TMC3, document
m43601;" ISO/IEC JTC1/SC29/WG11 MPEG, Ljubljana,
Slovenia, Jul. 2018.
22. "Global motion compensation for point cloud compression in
TM3, document m44751;" ISO/IEC JTC1/SC29/WG11
MPEG, Macau, China, Oct. 2018.
23. Marpe, D., Schwarz, H., and Wiegand, T., “Context-Based
Adaptive Binary Arithmetic Coding in the H.264/AVC Video
Compression Standard,” IEEE Trans. Circuits and Systems for
Video Technology, Vol. 13, No. 7, pp. 620–636, July, 2003.
24. C. Wencan and J. H. Ko, “Segmentation of Points in the Future:
Joint Segmentation and Prediction of a Point Cloud,” in IEEE
Access, vol. 9, pp. 52977-52986, 2021.
25. S. Hu, J. Cai and Y. Lai, “Semantic Labeling and Instance
Segmentation of 3D Point Clouds Using Patch Context
Analysis and Multiscale Processing,” in IEEE Transactions on
Visualization and Computer Graphics, vol. 26, no. 7, pp. 2485-
2498, 1 July 2020.
26. L. Ma, Y. Li, J. Li, W. Tan, Y. Yu and M. A. Chapman,
"Multi-Scale Point-Wise Convolutional Neural Networks for
3D Object Segmentation From LiDAR Point Clouds in Large-
Scale Environments,” in IEEE Transactions on Intelligent
Transportation Systems, vol. 22, no. 2, pp. 821-836, Feb. 2021.
27. J. Li, Q. Sun, K. Chen, H. Cui, K. Huangfu and X. Chen,
"3D Large-Scale Point Cloud Semantic Segmentation Using
Optimal Feature Description Vector Network: OFDV-Net," in
IEEE Access, vol. 8, pp. 226285-226296, 2020.
28. Zhang, Z., “iterative point matching for registration of free-
form 3d points and surfaces,” International Journal of Computer
Vision, vol. 13, 119-152, 1994.
29. “G-PCC test model v7, document N18664,” ISO/IEC
JTC1/SC29/WG11 MPEG, Gothenburg, SE, Jul. 2019.
30. “Common Test Conditions for G-PCC, document N00032,
ISO/IEC JTC1/SC29 WG7 3DG, Online, Oct. 2020.
31. G. Bjontegaard, “Calculation of average PSNR differences
between RD-curves,” presented at the 13th VCEG-M33
Meeting, Austin, TX, USA, pp. 2–4, Apr, 2001.
32. D. Tian, H. Ochimizu, C. Feng, R. Cohen, and A. Vetro,
“Geometric distortion metrics for point cloud compression,”
IEEE Int. Conf. Image Process. (ICIP), pp. 3460-3464, 2017.