Dynamic Point Cloud Texture Video Compression using the Edge Position Difference Oriented Motion Model

Ashek Ahmed[1,2], Manoranjan Paul[2] and Mark Pickering[1]

[1]University of New South Wales, Australia.
[2]Charles Sturt University, Australia.

Data Compression Conference, 2021

Point Clouds

– A point cloud refers to a set of points in three dimensional (3D) space and can be employed to describe a 3D surface.

– Each point contains geometry information i.e. the X-,Y-, and Z-coordinate values.

– Along with the geometry information, each point may also contain attribute information in the form of color, normal vector and material reflection.

Point Clouds

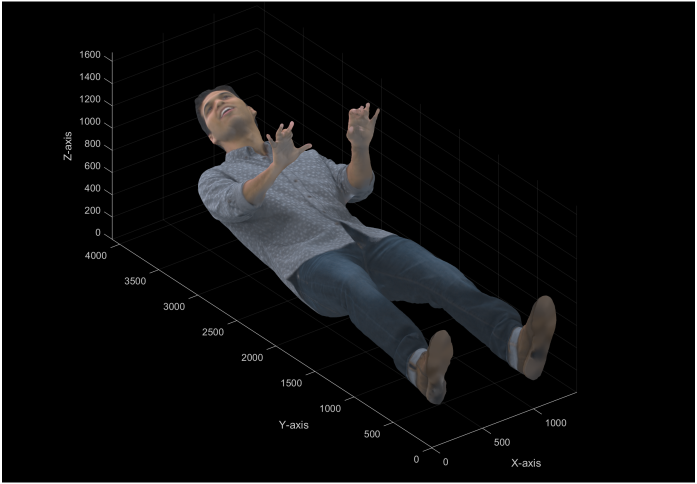

Figure 1: The *Loot* dynamic point cloud[1]

[1] E. d'Eon, B. Harrison, T. Myers, and P. A. Chou, "8i voxelized full bodies - a voxelized point cloud dataset," *inISO/IEC JTC1/SC29 Joint WG11/WG1 (MPEG/JPEG)input document WG11M40059/WG1M74006*, Geneva, Jan. 2017.

Point Clouds

– Due to their 3D object recovering property, point clouds have found applications in the fields of virtual reality (VR), augmented reality (AR), and autonomous driving.

– However, point clouds in their raw format occupy extensive amount of memory for storage or bandwidth for transmission.

– For instance, if a typical dynamic point cloud used in entertainment is considered (1 million points per frame and frame rate of 30 Hz), a total bandwidth of 3.6 Gbps is required without any compression[1].

[1] D. Graziosi, O. Nakagami, S. Kuma, A. Zaghetto, T. Suzuki, and A. Tabatabai, "An overview of ongoing point cloud compression standardization activities: video-based (v-pcc) and geometry-based (g-pcc)," *APSIPA Transactions on Signal and Information Processing*, vol. 9, pp. e13, 2020.

Dynamic Point Clouds and its compression

– A dynamic point cloud of moving objects is referred to as a dynamic object point cloud and is considered to be a video with a point cloud.

– The Moving Picture Expert Group (MPEG) have been developing a compression technique for dynamic point clouds (DPC) leveraging on existing video codecs.

– Briefly, a DPC is projected into geometry and texture video signals and these videos are encoded using modern video compression standards like H.264/AVC and HEVC.

Dynamic Point Clouds and its compression

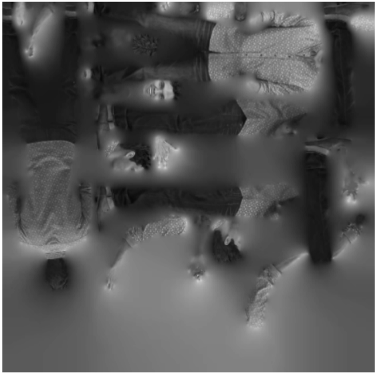

Figure 2: Reference frame, R (POC 19)

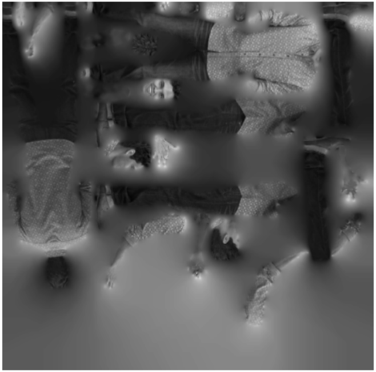

Figure 3: Current frame, C (POC 20)

Frames from the *Loot* DPC texture video sequence. The difference frame between R and C has a PSNR of 33.84 dB.

Dynamic Point Clouds and its compression

– In the V-PCC coding framework, a patch-based projection method is used to decompose the DPC into multiple 2D patches.

– These 2D patches from the DPC are organized into a video which is then encoded using the HEVC codec.

– However, similar patches in the current frame and the reference frame could be placed in completely different locations.

– As a result of this, for some HEVC prediction units (PUs), the HEVC motion estimation procedure may fail to locate their best matched block in the reference frame.

Dynamic Point Clouds and its compression

– A DPC texture frame contains significant edge information as can be seen from Fig. 4.
– Moreover, DPC texture sequence has the characteristics that:

Figure 4: Edge map for the frame C.

(i) different regions (therefore associated edges) of the same object may not be adjacent to each other within a projected texture frame.

(ii) there could be significant spatial distance between a region's current frame location and its position in the reference frame.

Edge Position Difference based Motion Modeling

– The goal of the EPD measure is to find the underlying geometrical transformation (M) between the edge maps of the current frame and of the reference frame, denoted by $\{c\}$ and $\{r\}$ respectively.

– the EPD measure can be described as:

$$D = f\Big(\{c\}, \{\widehat{c}\}\Big)$$

where,

$$\{\widehat{c}\} = M^{(\{r\}\rightarrow\{c\})}\Big(\{r\}\Big)$$

is a prediction of the edge map $\{c\}$.

– The operator $f(\,\cdot\,)$ is taken to be the Chamfer distance; computed at pixel-level using the hierarchical chamfer matching algorithm (HCMA)[1].

[1] G. Borgefors, "Hierarchical chamfer matching: a parametric edge matching algorithm," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 10, no. 6, pp.849–865, Nov. 1988.

Edge Position Difference based Motion Modeling

– the optimal motion model is obtained by solving the following problem:

$$M^{(\{r\}\rightarrow\{c\})} = \underset{M^{(\{r\}\rightarrow\{c\})}}{\arg\min} \quad f\Big(\{c\}, M^{(\{r\}\rightarrow\{c\})}\big(\{r\}\big)\Big)$$

Edge Position Difference based Motion Modeling

An example of the Chamfer distance is shown in Fig. 5.

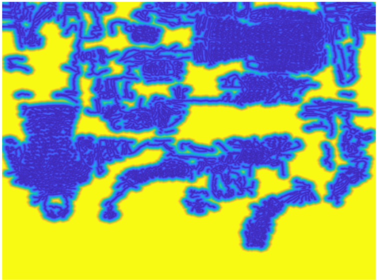

Figure 5: Chamfer distance image of $\{\hat{c}\} = M^{(\{r\}\rightarrow\{c\})}\left(\{r\}\right)$ where $M^{(\{r\}\rightarrow\{c\})}$ is the identity transformation (initialized value for the motion model).

– If a pixel is positioned over an edge, the corresponding Chamfer distance is 0 (which is the minimum Chamfer distance and represented using the color blue).

– The further a pixel is from any edge, the higher the corresponding Chamfer distance (represented using different lighter shades of blue, culminating at yellow).

Edge Position Difference based Motion Modeling

– The optimal affine motion parameters $M^{(\{r\}\rightarrow\{c\})}$ are employed to find a prediction, R_{epd}, of C[1].

– Experimental result show that if the R_{epd} frame is used as an additional reference frame to predict C within a modified HEVC encoder, a bit rate savings of 3.15% is achieved over standalone HEVC.

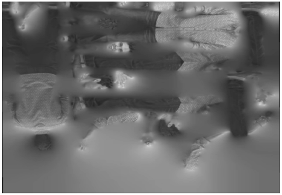

Figure 6: the EPD based motion model compensated frame R_{epd}. PSNR: 36.35 dB (better than 2.5 dB compared to R).

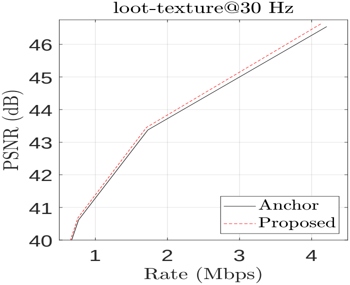

Figure 7: delta rate: −3.15%

[1] A. Ahmmed, D. Taubman, A. T. Naman, and M. Pickering, "Homogeneous motion discovery oriented reference frame for high efficiency video coding," *Picture Coding Symposium*, 2016, pp. 1–5.

Conclusions

- An approach is presented that attempts to model the motion in DPC texture video frames.
- Can generate predictions of superior PSNR and thereby yield bit rate savings.
- Increased computational complexity due to an additional reference frame and its generation process.