Occlusion-Aware Volumetric Video Streaming for Bandwidth-Efficient 3D Viewing

Final Project Report for 3DCV Course – Dec 2025

Group 12: D13949003 Mohammadreza Kamrani, M11207327 吳忠翰

Contents

1. Introduction

Volumetric video, often represented as point clouds, enables immersive 6-Degrees-of-Freedom (6DoF) experiences in Mixed Reality (MR) headsets. Unlike traditional 2D videos, volumetric content allows users to view scenes from any angle, providing a highly realistic sense of presence. However, streaming high-fidelity volumetric content presents significant challenges. It requires extremely high bandwidth—often exceeding 3.6 Gbps for 1 million points per frame—and substantial computational power.

Current mobile MR headsets, such as the HoloLens 2, have limited compute resources and battery life, making real-time decoding and rendering of raw point clouds difficult. Without optimization, users suffer from low visual quality, high latency, and poor Quality of Experience (QoE).

To address these issues, we propose a bandwidth-efficient streaming system inspired by "Theia". Our goal is to enable high-quality, low-bandwidth volumetric streaming on mobile headsets. We adopt a "Foveated Streaming" strategy, which leverages the human visual system's characteristics by streaming high-quality content only to the foveal area [1] (where the user is looking) while reducing quality in the peripheral vision. Additionally, we implement an occlusion-aware mechanism to ignore hidden points, further reducing unnecessary data transmission.

The log-polar transformation mimics the non-uniform photoreceptor distribution in the human retina. In the fovea—the central region responsible for sharp vision—cones are densely packed, providing high spatial acuity. Toward the periphery, receptor density decreases significantly. By mapping visual space logarithmically in the radial direction (eccentricity), the transformation naturally allocates higher sampling density to the central view point, similar to retinal organization. This allows aggressive data reduction in peripheral areas where the human visual system is less sensitive to detail, enabling substantial bandwidth savings without perceptible quality loss in the user's direct field of view. (Figure 1)

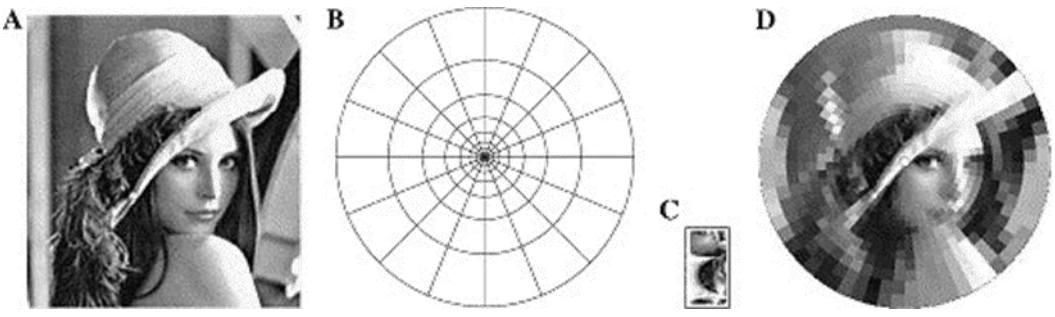

Figure 1-Log-Polar Transformation simulates human foveated vision [1]

2. Methodology

Our methodology focuses on mapping 3D point clouds to a 2D space for efficient processing and transformation and filtering and then back to 3D for rendering. Before detailing the transformation pipeline, we first introduce the dataset used to benchmark our system and define the bandwidth challenge.

2.1. Dataset

To evaluate the performance of our system on high-fidelity volumetric content, we utilized the industry-standard 8i Voxelated Full Bodies (8i VFB) dataset [2]. Specifically, we selected two sequences representing different geometric characteristics (Figure 2):

- LongDress: Characterized by complex geometry and self-occlusions due to the flowing dress.
- Loot: Characterized by high surface detail and texture.

Figure 2-. The Loot and LongDress Dataset [2]

These datasets represent a significant challenge for mobile streaming due to their high density. Each frame contains approximately 800,000 points (834K for LongDress, 794K for Loot). Without compression, the raw bitrate for transmission exceeds 2.8 Gbps (3,002 Mbps for LongDress, 2,858 Mbps for Loot). This massive bandwidth requirement serves as the baseline for evaluating our compression efficiency.

2.2. Method

In Figure 3 we have shown an overview of our processing pipeline. In the first step we transform

3D points cloud to a dummy 2D plane that its distance is 1 unit from the view point.

2.2.1. Simple 3D-2D Transformation

This transformation is a regular 3D-2D transformation:

$P_{point} = (X, Y, Z)$) → 3D coordinates of a point in the scene.

$P_{gaze} = (X_g, Y_g, Z_g)$) → 3D coordinates of the user's view origin (e.g., eye or head position).

$D_{gaze} = (d_x, d_y, d_z)$ → Normalized view direction vector.

First, compute the vector from the view origin to the point:

$P_{relative} = P_{point} - P_{gaze} = (X - X_g,; Y - Y_g,; Z - Z_g)$

Project $P_{relative}$ onto the view direction to find its depth (distance along the view axis):

$d_{point} = P_{relative} . D_{gaze} = (X - X_g) . d_x + (Y - Y_g) . d_y + (Z - Z_g) . d_z$

The dummy plane is defined perpendicular to (D_{gaze}) and located at a unit distance (or any chosen distance) along the view direction.

To find the 2D coordinates (x, y) on this plane:

$(x, y) = P_{project} = P_{relative} / d_{point}$

That is:

$x = (X - X_g)/d_{point}$, $y = (Y - Y_g)/d_{point}$

The Z component is normalized away, leaving a 2D planar projection relative to the view direction.

These (x, y) coordinates represent the angular offset from the view direction on the dummy plane.

They are later used for **Occlusion detection** (via depth-buffering in this 2D space), **Log-polar transformation**.

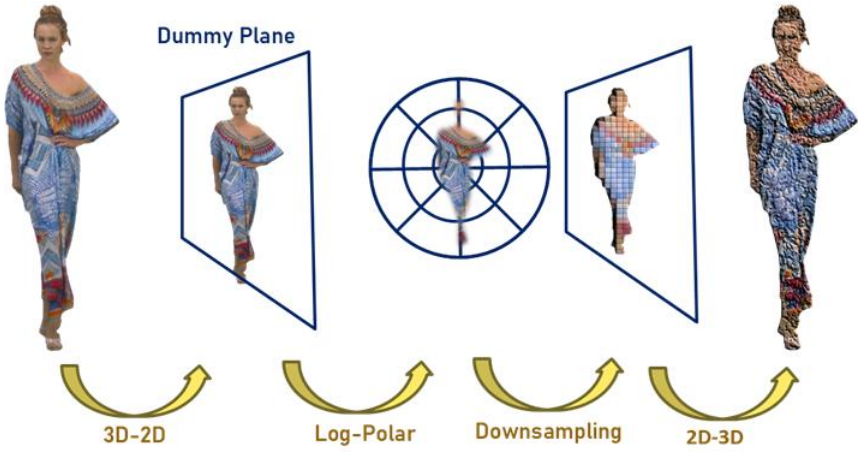

Figure 3-. Overview of the proposed coordinate transformation pipeline

2.2.2. Log-Polar Transformation

To simulate human vision and reduce data size, we utilize a Log-Polar transformation. This transformation maps the 3D points onto a 2D "Dummy Plane" relative to the user's view direction. The transformation allows us to maintain high sampling density in the center (fovea) and exponentially decrease density towards the periphery.

The Log-Polar coordinates (u, v) are computed as follows:

$$u = \log_b \frac{\rho}{\tan(MAR_0)}$$

$$v = \frac{\theta}{2\pi W}$$

Where ρ is the radial distance $\sqrt{x^2 + y^2}$, θ is the angular coordinate arctan $(\frac{y}{x})$, and MAR_0 (Minimum Angle of Resolution) is set to 1 arcminute, and b we set as 1.002. This mathematical mapping ensures that points in the peripheral vision are aggressively downsampled, while central points are preserved.

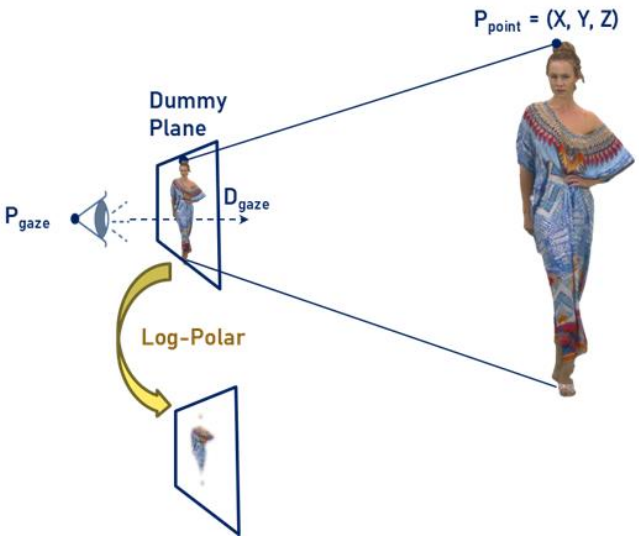

Figure 4- Geometric illustration of the view-driven projection model

2.2.3. Occlusion Awareness (Implicit HPR)

A major inefficiency in traditional streaming is transmitting points that are occluded (blocked by other points) from the user's perspective. Our system implements Hidden Point Removal (HPR) implicitly through the 3D-2D mapping process. When multiple 3D points map to the same 2D pixel

in the Log-Polar grid, only the point closest to the camera is retained. This effectively filters out hidden points without complex geometric calculations.

3. Results

We evaluated our system using the 8i Voxelized Full Bodies dataset. To demonstrate the adaptability of our Log-Polar sampling strategy, we conducted experiments under two distinct parameter configurations: High-Efficiency Mode (maximizing compression) and High-Fidelity Mode (maximizing visual detail).

3.1. System Specifications for Implementation

We tried our idea on two different system with the Table 1and Table 2 specifications.

Table 1-System I specifications

System I	
Component	Technical Specifications
GPU (Graphics Processing Unit)	NVIDIA GeForce RTX 3070 (8GB VRAM)
Driver & CUDA Toolkit	Driver v580.95.05, CUDA Toolkit v13.1.80
CPU (Central Processing Unit)	Intel Core i7-10700K (Base: 3.80 GHz)
System Memory (RAM)	31 GB
Operating System	Ubuntu 24.04.3 LTS (64-bit)
Development Environment	Conda 25.9.1, Python 3.13.9
Compilers	GCC/G++ 13.3.0, NVCC 13.1.80

Table 2--System I specifications

System II	
Component	Technical Specifications
GPU (Graphics Processing Unit)	NVIDIA RTX 6000 Ada Generation
Driver & CUDA Toolkit	Driver v580.76.05, CUDA Toolkit v12.1.105
CPU (Central Processing Unit)	Intel(R) Xeon(R) Gold 6426Y
System Memory (RAM)	503 GB
Operating System	Ubuntu 22.04.4 LTS
Development Environment	Conda 24.5.0, Python 3.10.15
Compilers	GCC/G++ 11.4.0, NVCC 12.1.105

3.2. Extreme Compression (High-Efficiency Mode)

we can control the rate of compressing with rate_adapt and r_bins parameters. First, we applied aggressive foveation parameters (higher rate_adapt and lower r_bins) on the "LongDress" sequence.

- Data Reduction: The input frame contained 765,821 points, while the output was reduced to 24,704 points.
- Compression Ratio: This achieved a massive 31:1 compression ratio.
- Performance: The processing time was 11 ms, enabling extremely low-latency transmission (< 90 Mbps) suitable for congested mobile networks (ran on 3070).

The results are shown in Figure 5 and Table 3.

Figure 5- Original and hidden points removed version of on frame of LongDress Point Cloud Dataset

Table 3- Performance for aggressive compressing

Input Points	765,821 points (single frame)
Theia Output	24,704 points
Processing Time	11 ms
Compression Ratio	31:1 (points)

3.3. High Fidelity (Quality-First Mode)

In contrast, the "Loot" sequence processed with parameters tuned for quality.

- Detail Preservation: From an input of 784,142 points, the system retained 170,959 points.
- Compression Ratio: The ratio is 4.6:1, retaining significantly more visual information in the para-

foveal and peripheral regions.

- Performance: The processing time remained extremely fast at 5.65 ms (ran on 6000 ada).

Figure 6- Original and compressed version of Loot Point Cloud Dataset with higher output quality

Table 4- Performance for moderate compressing

Input Points	784,142 points (single frame)
Theia Output	170,959 points
Processing Time	5.65 ms
Compression Ratio	4.6:1 (points)

These two experiments illustrate the system's flexibility. By adjusting the Log-Polar grid density (r_bins) and the decay rate (rate_adapt), users or applications can dynamically trade-off between bandwidth savings and visual resolution.

3.4. Quality Perception and Resource Allocation Strategy

To balance visual fidelity with bandwidth constraints, we implemented a multi-zone quality strategy based on the human visual system's acuity fall-off. We partitioned the user's Field of View (FoV) into three distinct concentric regions: Foveal, Para-foveal, and Peripheral.

Distinct parameters were assigned to each region:

- Foveal Region (Central): This covers the immediate view center 0.75 degrees. Here, we prioritize quality over compression, aiming for a Target PSNR of 45-50 dB (Near lossless). Although we apply a moderate point reduction (50-60%), this region is allocated a significant portion of the bitrate (30-40%) relative to its small size to ensure critical details are preserved.
- Para-foveal Region (Middle): Spanning 2.5 degrees to 3.75 degrees, this transition zone maintains structural context. We apply a high point reduction rate (85-90%) to achieve a "Good" quality rating (38-42 dB).
- Peripheral Region (Outer): Extending up to 15 degrees, this region occupies the largest visual area. To prevent bandwidth saturation, we apply extreme compression, removing 95-98% of the points3. Despite this reduction, due to the vast volumetric space it covers, it still utilizes 35-45% of the total bitrate to maintain acceptable spatial awareness (30-35 dB).

3.5.Elliptical Log-Polar Transformation

3.5.1. Physiological Motivation

According to Professor Chu-Song Chen advise about the different resolutions for vertical and horizontal sensitivity of human visual system, we studied about it and found that while the horizontal field of view is wider (approximately 180-200°), the vertical field is more limited (130-135°). More importantly, humans demonstrate slightly better sensitivity to vertical edges and details—a phenomenon attributed to evolutionary adaptation for detecting upright objects like trees, buildings, and facial features. The standard circular log-polar transformation treats both dimensions equally, overlooking this perceptual asymmetry.

3.5.2. Mathematical Formulation

We extend the standard log-polar transformation to an elliptical model by introducing separate scaling factors for horizontal (x) and vertical (y) dimensions:

Forward Transformation:

$$x_{ellip} = \frac{x}{a} \quad , \quad y_{ellip} = \frac{y}{b}$$

$$\rho_{ellip} = \sqrt{{x_{ellip}}^2 + {y_{ellip}}^2}$$

$$\theta_e = arctan2(y_{ellip}, x_{ellip})$$

$$u = log_b(\frac{\rho_e}{tan(MAR_0)}) \quad , \quad v = (\frac{\theta_e}{2\pi}) \times W$$

And inverse Transformation:

$$x = a\,\rho_e \times cos(\theta_e) \quad , \quad y = b\,\rho_e \times sin(\theta_e)$$

Where:

a = 1.6: Horizontal compression factor (larger = more compression)

b = 1.0: Vertical compression factor (preserves vertical detail)

ρ_e: Elliptical radius

θ_e: Elliptical angle

The ratio a/b =1.6 indicates 60% more compression in the horizontal direction, aligning with the broader horizontal field of view while preserving critical vertical details.

3.5.3. Implementation and Results

We implemented both circular and elliptical transformations using identical view parameters and processing pipelines. The elliptical transformation was integrated into the existing log-polar kernels with minimal computational overhead. Table 6 shows that with this strategy we can save bandwidth about 6.5% more.

Table 5- Performance Comparison - Circular vs. Elliptical Log-Polar

Metric	Circular Log-Polar	Elliptical Log-Polar	Change
Input Points	775,745	775,745	Same
Output Points	24,723	23,124	-6.5%
Compression Ratio	31.4:1	33.5:1	+6.7% improvement
Processing Time	10 ms	9 ms	-10%
Perceptual Quality	Good	Enhanced vertical detail	Improved
Bandwidth Saved	Baseline	Additional 6.5%	Significant

3.5.4. Key Findings and Implications

1. **Increased Compression Efficiency**: The elliptical transformation achieves **33.5:1 compression ratio** compared to 31.4:1 for circular, representing a **6.7% improvement** in data reduction while maintaining perceptual quality.
2. **Computational Efficiency**: Processing time slightly decreased (9 ms vs 10 ms), demonstrating that elliptical transformation imposes no computational penalty.

4. Differential Frame Streaming

After optimizing the point cloud based on human visual system (HVS) characteristics, we introduce another bandwidth reduction technique: **differential frame streaming**.

The key insight is that an MR device:

- Renders 3D points into 2D display frames
- Maintains a buffer of previous frames for reference

Since human eyes cannot detect small changes between consecutive frames, we can skip sending points that move less than a defined threshold.

Implementation:

1. **Send the first frame as a key frame** (full point cloud)
2. **For subsequent frames, send only points that move beyond a threshold distance** from their positions in the previous frame
3. **If changed points exceed 50% of the total**, send the entire frame as a new key frame

This approach significantly reduces bandwidth while maintaining perceptual quality.

Table 6 shows the result for different thresholds.

Table 6- Bandwidth save for Differential Frame Streaming

Distance Threshold (mm)	Points (#)	Save (%)
0	24723	0
1	22095	10.6
2	14463	41.4
5	3765	84.7

5. Conclusion

This project has successfully demonstrated a holistic framework for enabling high-quality volumetric video streaming on resource-constrained Mixed Reality headsets. By synergistically integrating three perceptual and computational optimization strategies, we achieve dramatic reductions in bandwidth and processing requirements without compromising the user's immersive experience.

Key Innovations:

- Occlusion-Aware Point Culling – By intelligently identifying and removing hidden points not visible from the user's viewpoint, we eliminate redundant data transmission before it ever reaches the network, ensuring that only truly visible geometry is processed and streamed.
- Human Visual System (HVS)-Informed Foveated Streaming – Our implementation advances foveated streaming through two complementary approaches: a) Circular Log-Polar Transformation – We first implement a standard circular transformation that applies uniform compression across all directions, achieving a 31.4:1 compression ratio and demonstrating the fundamental viability of gaze-contingent streaming. b) Elliptical Log-Polar Transformation – Building on this foundation, we introduce an elliptical model that incorporates the inherent asymmetry of human vision. With a 1.6:1 horizontal-to-vertical compression ratio, this biologically-inspired approach achieves 33.5:1 compression – a 6.7% improvement over circular methods – while better preserving critical vertical details like facial features and text.
- Differential Frame Streaming – Exploiting temporal coherence between consecutive frames, we transmit only points that change beyond a perceptible threshold. This method capitalizes on the fact that human vision cannot detect minor inter-frame variations, allow substantial bandwidth savings while maintain smooth visual continuity.

Together, these techniques form a multi-layered optimization pipeline that shifts computational burden to the server side, where powerful GPUs perform real-time, perception-aware preprocessing. The result is a lightweight, adaptive stream that mobile MR headsets can decode and render efficiently—enabling high-fidelity 6DoF experiences even under strict network and hardware constraints.

Future Directions include integrating deep learning for view prediction, dynamic parameter adaptation based on network conditions, and formal user studies to quantify Quality of Experience (QoE) across diverse scenarios.

In summary, this work bridges the gap between immersive volumetric content and practical deployment, paving the way for scalable, accessible, and visually compelling MR applications

6. References

[1] Bo Han, Yu Liu, and Feng Qian. ViVo: Visibility-Aware Mobile Volumetric Video Streaming. In The 26th Annual International Conference on Mobile Computing and Networking (MobiCom '20). Association for Computing Machinery, NY, USA, Article 16, 1–14.

[2] Yu Liu, Bo Han, Feng Qian, Arvind Narayanan, and Zhi-Li Zhang. Vues: Practical Mobile Volumetric Video Streaming Through Multiview Transcoding. In The 28th Annual International Conference on Mobile Computing and Networking (MobiCom '22). Association for Computing Machinery, NY, USA, 367–381.

[3] Yongjie Guan, Xueyu Hou, Nan Wu, Bo Han, and Tao Han. MetaStream: Live Volumetric Content Capture, Creation, Delivery, and Rendering in Real Time. In The 29th Annual International Conference on Mobile Computing and Networking (ACM MobiCom '23). Association for Computing Machinery, NY, USA, 1–15

[4] Zhongzheng Yuan, Yifei Zhu, Yuye Zhang, Zhigi Peng, Wei Tsang Ooi, and Yunxin Liu. Theia: View-driven and Perception-aware Volumetric Content Delivery for Mixed Reality Headsets. In Proceedings of the 22nd Annual International Conference on Mobile Systems, Applications and Services (MobiSys '24). Association for Computing Machinery, NY, USA, 302–315