Hybrid Projection For Encoding 360 VR Videos

Jintao Tang*  Xinyu Zhang†

School of Computer Science & Software Engineering, East China Normal University, China

ABSTRACT
During the past five years, tons of economic 360 VR cameras (e.g., Ricoh Theta, Samsung Gear360, LG 360, Insta 360) are sold in the market. While 360 VR videos become ubiquitous very soon, 360 VR video standardization is still under discussion in the digital industry, and more concrete efforts are desired to accelerate its standardization and applications. Though ERP has been widely used for projection and packing layout while encoding 360 VR videos, it has severe projection distortion near poles. In this paper, we introduce a new format for encoding and storing 360 VR videos using hybrid cylindrical projection after thoroughly analyzing the problems with ERP. We show that our new hybrid format can minimize stretching distortion and generate well balanced pixel distribution in the resulting projection.

Index Terms: Computing methodologies—Computer graphics—Graphics systems and interfaces—Virtual reality

1 INTRODUCTION
Mobile virtual reality (VR) breakthrough creates a fast-growing demand for 3D immersive content. Compared to computer-generated content, 360 VR cameras [1, 4, 12, 14, 20] can rapidly capture immersive 360-degree VR videos by directly recording a surrounding environment in every direction at the same time. During playback, the viewer locates at the center of a spherical screen on which 360 VR videos are projected. 360 VR videos allow viewers feel actually in the midst of all the captured environment. There are many applications for 360 VR videos such as filming [2], live sports broadcasting [27], VR tourism [1], social communication [15], teleoperation in robotics [16, 19].

In the past few years, a few industrial leaders such as Google [1], Facebook [4] and Samsung [21], and a few new startups [12, 13] launched various 360 cameras. Tons of economic 360 VR cameras (e.g., Ricoh Theta, Samsung Gear360, LG 360, Insta 360 and more others) are sold in the market. While 360 VR videos become ubiquitous very soon, 360 VR video standardization is still under intensive discussion and concrete efforts are urgently desired to bridge the standardization gap in the digital industry.

In order to support efficient storage, access and processing, a 360 video requires to be projected onto a 2D domain (i.e., a transformation from a spherical surface to a plane) and stored in conventional video format. Among these projections, Equi-Rectangular Projection (ERP) [23] is widely used in gaming industry and cartography. This projection unfolds and flattens a sphere into a 2D rectangle. Cubemap Projection (CMP) [9] is another frequently used method. It deforms a sphere into a cube and flattens a 1/6 spherical surface into a cube’s face. Then the cube’s six faces are unfolded and packed onto a rectangle. A straightforward deformation is to embed the sphere in a cube and project the spherical surface outwards onto the cube’s faces. CMP offers better pixel uniformity than ERP. A few variations like Google’s Equi-Angular Cubemap (EAC) [8] were also suggested. EAC further reduces projection distortion. Both CMP and EAC use six squares for bijective mapping and have been used in 360 videos by Google. The work in [11] used two stages to perform uniform mapping. Other methods use more complex domains for projections, such as Compact Octahedron Projection (COHP) [5, 17] and Compact Icosahedron Projection (CISP) [6]. COHP and CISP project spherical triangular surfaces onto planar triangular domains. COHP uses eight triangles and CISP uses twenty triangles. However, due to its sophisticated bijective mapping and a large number of internal packing edges, it is not clear if CISP is useful in practice.

3 CYLINDRICAL PROJECTION
Our goal is to build a map between a sphere and a rectangle while minimizing distortions and maintaining evenly distributed pixel density. The main idea is to find an appropriate mapping function between a planar rectangle and a spherical surface (see Figure 1). A good projection will allow us to generate high quality 360 VR videos. Here, we denote a point on the spherical surface as \((\theta, \phi)\) with longitude \(\theta\) and latitude \(\phi\), and a corresponding point on the \(uv\) projection rectangle as \((u, v)\). A general explicit projection form is given below.

\[
\begin{align*}
    u &= F(\theta, \phi) \\
    v &= G(\theta, \phi)
\end{align*}
\]

3.1 Stretching Ratio
We will use two metrics: Area Stretching Ratio (ASR) [24] and Linear Stretching Ratio (LSR) to measure projection distortion. Note
We use two cylindrical projections: regular Equi-Rectangle Projection (ERP) and Transverse Cylindrical Projection (TCP) to generate a hybrid projection. Both ERP and TCP belong to the category of cylindrical projection family, in which the projection wraps the sphere with a tangent cylinder and maps the spherical surface to a rectangle. Intuitively, ERP unfolds a vertically ‘standing’ cylinder and then wraps the sphere. Since adjacent longitudes have the same distance, ERP is also called Equidistant Cylindrical Projection. TCP unfolds a horizontally ‘lying’ cylinder in the constraint of area-preserving. The regular cylindrical projection (i.e., ERP) is illustrated in Figure 2 and the axis of the cylinder in TCP lies in the equatorial plane.

**3.2 Cylindrical Projections**

ERP and TCP show different performances in terms of ASR and LSR. By taking their respective advantages, we can obtain better performance than each individual projection. We give a detailed analysis as follows.

### 3.2.1 ERP

ERP has the simplest form formulated in Eq. (7).

\[
\begin{align*}
  & u = \theta \\
  & v = \varphi 
\end{align*}
\]

From Eqs. (6) and (7), its LSR is derived as

\[
\text{LSR}_{\text{ERP}} = \frac{1}{\cos \varphi}
\]

As a result, LSR\(_{\text{ERP}}\) varies with respect to the latitude \(\varphi\). From Eq. (1) and Eq. (7), we derive that \(J_{\text{ERP}} = 1\), and then we have

\[
\text{ASR}_{\text{ERP}} = \cos \varphi.
\]

### 3.2.2 TCP

When using a transverse cylinder, we formulate an equal-area TCP as

\[
\begin{align*}
  & u = \cos \varphi \sin \theta \\
  & v = \arctan \left( \frac{\tan \varphi \sin \theta}{\cos \varphi \cos \theta} \right)
\end{align*}
\]

Using Eqs. (6) and (10), we derive its LSR shown below

\[
\text{LSR}_{\text{TCP}} = \frac{1}{\cos \varphi} \cdot \sqrt{\left( \tan^2 \varphi \sin^2 \theta + \cos^2 \varphi \cos^2 \theta \right) + \frac{\cos^2 \varphi \cos^2 \theta}{\sin^2 \varphi \sin^2 \theta + \cos^2 \varphi}}
\]

This is a function of longitude \(\theta\) and latitude \(\varphi\). From Eq. (10), we have

\[
J_{\text{TCP}} = \cos \varphi \cdot \frac{\cos^2 \varphi \cos^2 \theta + \sin^2 \varphi}{\sin^2 \varphi \sin^2 \theta + \cos^2 \varphi} = \cos \varphi
\]

As a result, its ASR can be computed using Eq. (13), which is a constant of 1.

\[
\text{ASR}_{\text{TCP}} = \frac{\cos \varphi}{J_{\text{TCP}}} = 1
\]

For intuitive analysis, we plot these functions: \(\text{LSR}_{\text{ERP}}, \text{ASR}_{\text{ERP}}\) and \(\text{LSR}_{\text{TCP}}\) given in Eqs.(8), (9) and (11), respectively. As shown in Figure 3, both \(\text{LSR}_{\text{ERP}}\) and \(\text{ASR}_{\text{ERP}}\) are cosine functions with respective to latitude \(\varphi\). Their stretching ratios are acceptable near equator (i.e., \(\varphi \leq \varphi_c\), where \(\varphi_c\) is a specified threshold). Our TCP is an area-preserving projection (i.e., \(\text{ASR} = 1\)). \(\text{LSR}_{\text{TCP}} \approx 1\) for specified latitude \(\varphi\) (\(|\varphi| \geq \varphi_c\)). Intuitively speaking, the regular ERP has good performance around equator while TCP has good performance near poles. Note that \(\text{ASR}_{\text{TCP}}\) is a constant.

---

**Figure 1:** The LSR and ASR of two corresponding surfaces.

**Figure 2:** Spherical projection used in encoding 360 VR videos. (a) spherical surface; (b) cylindrical projection process and (c) rectangular uv projection plane.
4 HYBRID PROJECTION

4.1 Latitude Threshold

Based on the analysis of ERP and TCP in Section 3, we define the projection latitude threshold \( \phi_t \) to combine the two cylindrical projections into one hybrid projection, as shown in Figure 4.

In order to generate a compact form, the key is to choose the value \( \phi_t \). Here, we introduce two possible values: \( \pi/4 \) and \( \pi/6 \), making the frame-packing simple and easy. While \( \phi_t = \pi/4 \), the resulting hybrid projection combines equal latitude of ERP and TCP, which is an intuitive hybrid combination of both ERP and TCP. While \( \phi_t = \pi/6 \), the resulting hybrid projection discards more polar regions of ERP that exhibits the most distortions, and retains more polar regions of TCP that preserves area, yet has more distortion in terms of LSR metric. Table 1 shows the range of stretching ratios for latitude threshold \( \phi_t \). Table 2 lists the stretching ratios of ERP, CMP [9], TCP, SUM [11], EAC and our hybrid methods. Without prejudice to the results, the lower bounds are scaled to 1.00. Therefore, the smaller upper bound yields the lower projection distortion.

Table 1: Stretching ratio range for latitude threshold \( \phi_t \)

<table>
<thead>
<tr>
<th>( \phi_t )</th>
<th>LSR(_{ERP} )</th>
<th>LSR(_{TCP} )</th>
<th>ASR(_{ERP} )</th>
<th>ASR(_{TCP} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \pi/4 )</td>
<td>[1.00, 1.41]</td>
<td>[0.95, 1.41]</td>
<td>[0.70, 1.00]</td>
<td>1</td>
</tr>
<tr>
<td>( \pi/6 )</td>
<td>[1.00, 1.16]</td>
<td>[0.90, 2.00]</td>
<td>[0.87, 1.00]</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 2: Stretching ratio ranges for ERP, CMP [9], TCP, SUM [11], EAC [8] and our hybrid methods.

<table>
<thead>
<tr>
<th></th>
<th>LSR</th>
<th>ASR</th>
</tr>
</thead>
<tbody>
<tr>
<td>ERP</td>
<td>[1.00, +( \infty )]</td>
<td>[1.00, +( \infty )]</td>
</tr>
<tr>
<td>CMP</td>
<td>[1.00, +( \infty )]</td>
<td>[1.00, +( \infty )]</td>
</tr>
<tr>
<td>TCP</td>
<td>[1.00, +( \infty )]</td>
<td>[1.00, 1.00]</td>
</tr>
<tr>
<td>SUM</td>
<td>[1.00, +( \infty )]</td>
<td>[1.00, 1.00]</td>
</tr>
<tr>
<td>EAC</td>
<td>[1.00, +( \infty )]</td>
<td>[1.00, 1.41]</td>
</tr>
<tr>
<td>Our Hybrid((\pi/4))</td>
<td>[1.00, 1.48]</td>
<td>[1.00, 1.41]</td>
</tr>
<tr>
<td>Our Hybrid((\pi/6))</td>
<td>[1.00, 2.23]</td>
<td>[1.00, 1.15]</td>
</tr>
</tbody>
</table>

4.2 Frame Packing Layouts

Here, we propose three frame-packing layouts for the hybrid projection while adopting the two \( \phi_t \) values. The width-to-height aspect ratios are given in Table 3.

Table 3: Width-to-height ratios of three frame-packing layouts under hybrid projection

<table>
<thead>
<tr>
<th>Frame-Packing Layout</th>
<th>( \phi_t )</th>
<th>Width-to-Height Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>( \pi/4 )</td>
<td>6.83</td>
</tr>
<tr>
<td>B</td>
<td>( \pi/4 )</td>
<td>1.71</td>
</tr>
<tr>
<td>C</td>
<td>( \pi/6 )</td>
<td>4.10</td>
</tr>
</tbody>
</table>

As shown in Figure 5, frame-packing layout A for \( \phi_t = \pi/4 \) is similar to Segmented Sphere Projection [28] and it has 2 packing seams indicated by red dash lines. The width-to-height ratio is 6.83. Frame-packing layout B for \( \phi_t = \pi/4 \) divides both ERP and TCP into eastern and western hemispheres, and stitches them into 4 discontinuous regions, making the hybrid projection 4 packing seams. The width-to-height ratio is 1.71, which is very close to industrial standard aspect ratio 16:9. Frame-packing layout C for \( \phi_t = \pi/6 \) stitches four segments with 3 packing seams. Its width-to-height ratio is 4.10.
5 EXPERIMENTS AND RESULTS

To demonstrate performances of hybrid projection and frame-packing layouts described in Section 4, we conducted extensive experiments of quality of evaluation (QoE). We also show the computational costs of different projections and frame-packing layouts.

5.1 Discontinuity, Distortion and Pixel Density

To perform intuitive comparison, we use discontinuity graph, distortion graph and pixel density graph to present the performance of hybrid projection.

As shown in the first row of each graph in Figure 6, we use a color stripped texture to illustrate discontinuity in resulting packing layouts. Textures are preserved very well while discontinuity merely occurs at the packing edges. The second row of each graph is designed to demonstrate area distortion during projection. A given number of red dots in the same size are evenly spreading on a spherical surface. A good projection minimizes distortion by preserving projection areas. Thanks to the advantages of both ERP and TCP, the distortion of hybrid projection is well controlled. The results shown in Figure 6 confirms our analysis given in Table 2. From the pixel density in the third row of each graph in Figure 6, we observe that pixel density at equator is well preserved using ERP format, and the polar regions are condensed using TCP format.
Figure 6: QoE of Projections. (a)(b) are two basic cylindrical projections ERP and TCP. (c)(d)(e) are three frame-packing layouts under hybrid projections. The top, middle and bottom graphs from (a) to (e) is to demonstrate discontinuity, distortion and pixel density.

5.2 PSNR & BD-rate

Peak signal-to-noise ratio (PSNR) has been widely used to perform objective QoE for image/video sequences. However, for 360 VR videos, since pixels are located on a spherical surface, it is non-trivial to evaluate spherical images/videos using PSNR. Spherical-PSNR (S-PSNR) [25] was proposed to evaluate omnidirectional video content, and Craster Parabolic Projection PSNR (CPP-PSNR) [25] was suggested to evaluate distortions. Each type of PSNR has three metrics suggested in [29], also shown in Table 4. Here we applied Cross Format type and End-to-End type of S-PSNR and CPP-PSNR to evaluate the process of projections.

Table 4: PSNR Metrics Used for Objective QoE

<table>
<thead>
<tr>
<th>Name</th>
<th>Evaluate Process</th>
</tr>
</thead>
<tbody>
<tr>
<td>CodeC</td>
<td>Encode/Decode</td>
</tr>
<tr>
<td>Cross Format</td>
<td>Forward conversion, Encode/Decode</td>
</tr>
<tr>
<td>End-to-End</td>
<td>Forward/Inverse format conversion, Encode/Decode</td>
</tr>
</tbody>
</table>

We used four test sequences for video encoding and projection after converting them into six different resolution levels listed in Table 5. Then their PSNRs are calculated.

To make comparisons between PSNR/Bitrate curves, we use the method suggested by Bjontegaard [3] to calculate the BD-rate. The BD-rate calculates an average reduction among different PSNR values. Intuitively, for a given PSNR value, the smaller a video bitrate is, the better quality it has. We take a few PSNR values of different bitrates for two videos and then we use linear interpolation to perform regression between these PSNRs and bitrate values. We average the bitrate reductions (BD-rate) and compare them against standard ERP. The comparison between individual projections are plotted in Figure 7.

Among these three frame-packing layouts using hybrid projection, the majority of BD-rate reductions on Cross-Format S-PSNR and CPP-PSNR, End-to-End S-PSNR and CPP-PSNR reaches to 5%, as shown in Figure 7. This shows that our hybrid projection has better performance compared with the regular ERP. More specifically, hybrid projection frame-packing B has the most significant BD-rate reductions. We also show the comparison with other techniques such as EAC [8] and SUM [11]. Figure 8 shows more results for video sequences using three hybrid projections and frame-packing layouts.
Figure 7: BD-rate reduction of different projections and packing layouts compared against standard ERP.
Figure 8: Standard ERP (left-top), TCP (left-bottom) and Hybrid projection (Frame-packing A: right-bottom, Frame-packing B: middle, Frame-packing C: right-top) graphs of test sequences.
5.3 Computational Costs

We evaluate computational performance during video hybrid projection conversions. We use four video sequences to test the performance of 360 video encoding and projection. The timings are measured on a PC with a 17-6700 CPU under Windows.

<table>
<thead>
<tr>
<th>Resolution Level</th>
<th>ERP</th>
<th>TCP</th>
<th>Hybrid-A</th>
<th>Hybrid-B</th>
<th>Hybrid-C</th>
</tr>
</thead>
<tbody>
<tr>
<td>384 × 192</td>
<td>43.0s</td>
<td>43.0s</td>
<td>41.5s</td>
<td>44.2s</td>
<td>44.3s</td>
</tr>
<tr>
<td>768 × 384</td>
<td>86.3s</td>
<td>87.1s</td>
<td>85.3s</td>
<td>87.2s</td>
<td>85.6s</td>
</tr>
<tr>
<td>1152 × 576</td>
<td>149.4s</td>
<td>151.2s</td>
<td>145.9s</td>
<td>149.2s</td>
<td>146.1s</td>
</tr>
<tr>
<td>1536 × 768</td>
<td>226.9s</td>
<td>231.6s</td>
<td>230.7s</td>
<td>227.5s</td>
<td>227.2s</td>
</tr>
<tr>
<td>1920 × 960</td>
<td>319.5s</td>
<td>327.6s</td>
<td>329.1s</td>
<td>320.4s</td>
<td>320.7s</td>
</tr>
<tr>
<td>2304 × 1152</td>
<td>426.6s</td>
<td>443.8s</td>
<td>441.1s</td>
<td>431.9s</td>
<td>431.8s</td>
</tr>
</tbody>
</table>

As shown in Table 5, it is obvious that ERP is fastest due to its simplicity, while TCP is slower than others. Hybrid projections lie between ERP and TCP.

6 CONCLUSIONS AND FUTURE WORK

We present an efficient 360 video format using hybrid cylindrical projection. This format takes the advantages of regular ERP around equator area and TCP near poles. Hence, it can provide a better performance at the polar area than ERP by combining with TCP. Meanwhile, this new format can effectively control area stretching and linear stretching ratios. Among the three hybrid projection packing layout, frame-packing B shows good BD-rate reductions while maintaining a width-to-height ratio close to the industrial standard aspect ratio 16:9. Due to its simplicity, we expect our work will contribute to the ongoing standardization efforts towards efficient storage format for 360 VR videos. We expect this work can inspire more research along this direction.

Our frame-packing layouts has discontinuous edges. For a highly compressed video, some artifacts may occur to the packing seams. However, the artifacts can be reduced using padding techniques. In future work, we would like to investigate other techniques like conformal projection to further improve projection efficiency. In addition, some optimization techniques such as parallel computation, GPU-acceleration and edge padding, can be used to further speed up the entire performance.

ACKNOWLEDGMENTS

This work was supported by the NSFC (No.61631166002, No.61572196). The authors wish to thank Wensong Li and Nikk Mitchell from FXG VR for helpful discussions.

REFERENCES