Pixels and Panoramas

An enhanced cubic mapping scheme for video/image-based virtual-reality scenes.

By Yibo Fan, Yize Jin, Zihao Meng, and Xiaoyang Zeng

Virtual reality (VR) is rapidly appearing in various fields, such as navigation, robotics, and documentation. Spherical panoramic video, compared to 3D modeling, provides immersive and omnidirectional views in a much more convenient way. However, state-of-the-art video or image-encoding techniques, such as high-efficiency video coding or JPEG, require rectangular input sequences. Spherical videos are traditionally projected onto a plane or a cube for convenient encoding, but mapping quality and encoding efficiency are not considered. In this article, we propose GVScube projection, a method using a cube-Snyder (Scube) projection along with a gradually varied (GV) sampling method to generate panoramic video. This method achieves better pixel uniformity and less area deviation than other methods.

CHALLENGES IN MAKING VIRTUAL ENVIRONMENTS

With the development of VR technologies, some have proposed image-based rendering (IBR) as a substitute for traditional 3D computer graphics for creating virtual environments [1]–[6]. Panoramic videos provide immersive experiences by displaying 360° virtual environments. However, their spherical nature introduces difficulties in encoding. Since current mature encoding techniques require input videos to be planar and rectangular, different mapping methods have been proposed to represent panoramic videos in a proper format.

Cylindrical projections and cube-map projections are currently the most frequently used mapping schemes [7], [8], as shown in Figure 1. In cylindrical projection, a sphere is divided by latitude and longitude, and every grid is considered a pixel. In cube-map projection, an environment map is projected onto six faces of a cube, and images on these faces can be easily rearranged as six tiles of a frame.

Although cylindrical projection and cube-map projection both unfold the sphere, the mapping efficiency still must be considered. In cylindrical projection, pixel density is much greater around poles than at the equator, and in cube-map projection, oversampling occurs around the corners of the cube, while undersampling occurs around the centers of faces. This nonuniform sampling in the sphere causes data redundancy, and visual quality is unstable. The uniform distribution of spatial pixels, unlike ordinary pixels in images or videos, cannot have both the same area (solid angle) and shape.

Many map projections have been proposed to solve this problem. For example, Fu et al. offered rhombic dodecahedron (RD) mapping [9], which uses an RD instead of a cube model. Ho et al. suggested unicube mapping [10], which uses gradually varied sampling strategy. Yu et al. [11] discussed a content-adaptive representation by adjusting the sampling density of different latitudes based on the video content. However, because the process is so time-consuming, the adjustment can only be determined at the first frame of the video.

In this article, we present GVScube by implementing Snyder's projection on a cube model and then adjusting the subdivision scheme on the cube model. Our proposed scheme can provide more uniform spatial pixel density than other existing projections, including cylindrical projection, cubic projection, RD mapping, and unicube mapping, and it has less standard pixel area deviation than the other projections, which implies a more uniform area mapping and a more faithful record of the environment. A detailed description of Snyder's projection is presented in this article, along with experimental results.

BACKGROUND

Computer graphics are widely used to provide 3D environments and immersive experiences in VR experiments. Although the constructed models are projected onto the screen with accurate depth information, their limitations are obvious:

Digital Object Identifier 10.1109/MCE.2018.2880809
Date of publication: 6 February 2019

2162-2248/19©2019IEEE

1) The modeling process can be rather complicated. To construct a detailed 3D environment manually is time-consuming and laborious.
2) Although texture in the models can be rendered exquisitely, the differences between modeled scenes and real scenes can still be easily recognized.
3) The complexity of modeled scenes is restricted by computing capacity.

An IBR approach is used to reconstruct VR environments by mosaicking overlapping photographs captured from different directions into a 360° image [1]. Many different mapping schemes have been developed to record and store omnidirectional image signals. Equirectangular projection, which is a cylindrical projection, is widely used in VR scenes today [7]. However, equirectangular projection cannot provide a uniform spatial pixel distribution, since pixels are concentrated at the poles of the sphere and thin out around the equator.

A cubic map projection is used as a convenient means for storage and transmission of panoramic videos [8]. It solves the problem of distortion at the poles. However, spatial pixel density derived by cubic mapping is not distributed uniformly.

Uneven spatial pixel distribution may lead to an undesirable visual experience. For example, users may find visual quality varying with changes in the viewing direction. When cylindrical projection is used, the viewport around the North Pole shows better visual quality than those around the Equator. Moreover, an unusually high pixel density in some regions may lead to pixel overlap in the viewport. When the texture of a cylindrical projection is complicated, pixels will overlap around poles and cause distortion in the original image.

An RD is used instead of a cube to obtain a more uniform distribution of pixels [9]. The RD projection provides better results because, with more faces, the model is closer to a sphere, so perspective projection is more likely to give an even distribution. However, the complexity of the RD model

Many different mapping schemes have been developed to record and store omnidirectional image signals.

requires more calculations during the projection process, and the perspective projection used to project subdivided pixels of the model onto a sphere causes uneven distribution.

Isocube mapping, proposed in [12], uses the traits of a cube model and can provide good uniformity. By dividing the sphere directly into two polar base faces and four equatorial base faces and then by subdividing base faces via specified curve equations, isocube mapping provides a much more isotropic distribution than cubic projection.

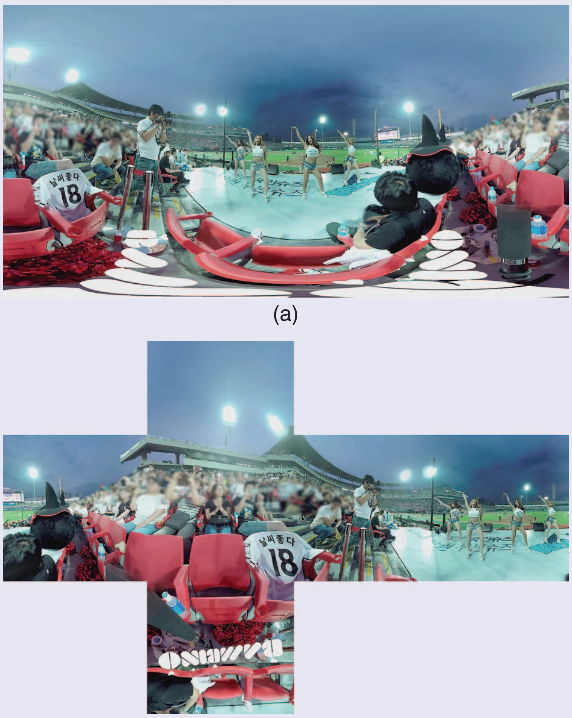

FIGURE 1. A (a) cube map and (b) cylindrical map.

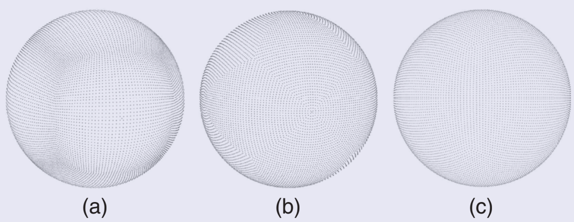

FIGURE 2. Sampling models of different mapping schemes. A (a) cubic map, (b) Scube projection, and (c) unicube mapping.

Unicube mapping has been proposed as an improvement [13]. Instead of dividing the sphere directly, unicube mapping only needs retreatment with a "different-subdivision" strategy on the cube model, which gives almost the same spatial pixel uniformity and can facilitate dynamic environment mapping in real time.

PROPOSED SNYDER'S EQUAL-AREA PROJECTION BASED ON CUBE MODEL

CUBE-SNYDER PROJECTION

Snyder's equal-area projection, first proposed for cartography by Snyder [14], ensures point-to-point mapping between a polyhedron and a sphere, with areas of correlated regions unchanged. Pixels on faces of a polyhedron are projected via a modified Lambert azimuthal equal-area projection. Although Snyder's equal-area projection can be implemented on different kinds of polyhedrons, we chose a cube as the projection model for the following reasons: 1) In contrast with cubic projection and unicube mapping, Scube projection is based on the same projection model, but has better data uniformity; 2) although Snyder's projection with more complicated models provides an even more uniform distribution, the shapes of faces (such as pentagons or hexagons) are not suitable for encoding; as for simpler models, their pixel uniformity does not meet expectations; 3) with fewer faces than the RD model, videos generated via a cube model lead to better coding efficiency.

Here we provide intuitive illustrations of three sampling methods mentioned previously. For cube-map projection and Scube projection, we chose 48×48 pixels uniformly distributed on every face of a cube to ensure that the total pixel number of every projection is almost the same. Mapping results of five projection formats are shown in Figure 2. It is obvious that the Scube projection and unicube mapping have better data uniformity than the others.

SPHERE-TO-CUBE MAPPING

Sphere-to-cube mapping gives formulas of mapping points on the sphere model to associated points on the cube model. Given a point on a sphere, the associated point on the cube model can be determined by the formulas. Three steps are taken to retrieve pixel data: 1) Locate the slice and the face on which point P lies and calculate its polar angle and radius under polar coordinates; 2) implement Snyder's equal-area projection to transform the polar angle and radius on the sphere into a polar angle and radius on faces of the cube; 3) transform the polar angle and radius on the cube to Cartesian coordinates, and the location of the cell in which P' lies is determined. It is illustrated as follows:

$$\begin{gathered}(X_p, Y_p, Z_p)\ \textit{location of } P \textit{ on the unit sphere}\\ \to (\alpha, t)\ \textit{polar coordinates of } P \textit{ on the unit sphere}\\ \to (\theta, \rho)\ \textit{polar coordinates of } P' \textit{ on the cube model}\\ \to (x, y)\ \textit{coordinates of } P' \textit{ on a face of the cube model}\\ \to (m, n)\ \textit{location of the cell in which } P' \textit{ lies.}\end{gathered} \tag{1}$$

Every face of the cube model and its associated region on the sphere is divided into eight slices, each representing an isosceles right triangle, as in Figure 3(a) and (b), and every slice has its own polar coordinates, so that the polar angle is restricted to the range $[0,\pi/4]$. For a polar angle and radius on the sphere, we define polar angles to be dihedral angles and the radius to be the length defined by a great circle between the point and the pole. In Figure 3(a), for example, point P lies in slice ANI, whose radial axis is part of the great circle passing through I and A, pointing from I toward A. The dihedral angle α between plane POI and plane AOI is the polar angle for point P, and the radius t is the spherical distance between point P and point I.

Snyder's projection gives a point-to-point mapping between polar coordinates, $(\alpha,t)\rightarrow(\rho,\theta)$, as shown in Figure 3. To implement Snyder's equal-area projection on a cube model, without losing generality, the radius of the inscribed sphere of the cube is set to be $r=\sqrt{\pi/6}\cdot R$, where R is the radius of the sphere model, so that their surface areas are equal. Readers could refer to Snyder's paper for a more detailed description [14]. Simplified formulas for Scube projection are given as follows:

$$\begin{aligned}
H &= \arccos(\sin(\alpha)/2-\cos(\alpha)/2)\\
\theta &= \operatorname{arccot}(r^2/((\alpha+H-2\pi/3)\times R^2)-1)\\
q &= \arctan(\sqrt{2}/(\cos(\alpha)+\sin(\alpha)))\\
\rho &= \frac{r\times\sin((t/2)/r)}{\sin(\theta+\pi/4)\times\sin((q/2)/r)}.
\end{aligned} \tag{2}$$

In (2), α is the polar angle of pixel P on the sphere, and t is its radius, while θ is the polar angle of the pixel on the cube, and ρ is its radius. H represents spherical angle PAI, an interior angle of spherical triangle API, and S is its area, as in Figure 3(a) and (b). After α and t on the sphere model are found by geometric calculation, θ and ρ on the cube model are obtained via Scube projection.

Transforming a polar angle and radius into Cartesian coordinates on the cube model can be accomplished quickly. As in Figure 4(a) and (b), a simple polar coordinate transformation enables the easy calculation of coordinates (x, y) of P' on the face according to its polar coordinates (ρ,θ).

With this method, pixels are uniformly distributed on the face of the cube model. In the section "Gradually Varied Sampling on the Cube Model," we will use nonuniformly distributed pixels, whose areas differ according to their positions, so that it can counteract the nonisotropic solid angle distribution over the sphere.

CUBE-TO-SPHERE MAPPING

Cube-to-sphere mapping provides the mapping scheme from source points on a cube to target points on a sphere via a process that is the inverse of the steps introduced in the former section. After obtaining the polar coordinates (ρ,θ) of a source point from its Cartesian coordinate (x, y) on a face of the cube, the Newton–Raphson method is applied to map the point P' on the cube model to its corresponding point P on

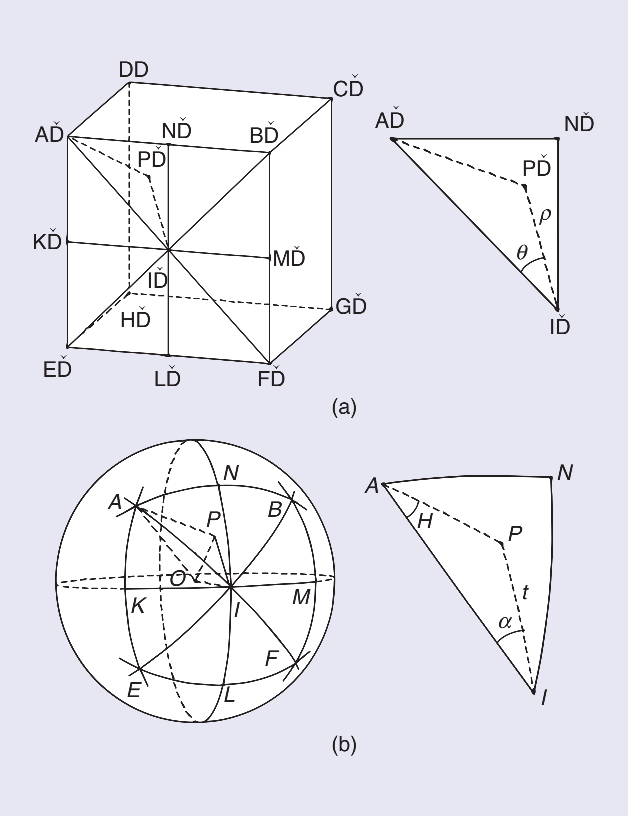

FIGURE 3. The proposed Snyder's equal-area projection based on a cube model. Every face of the (a) cube model and its associated region on the (b) sphere is divided into eight slices.

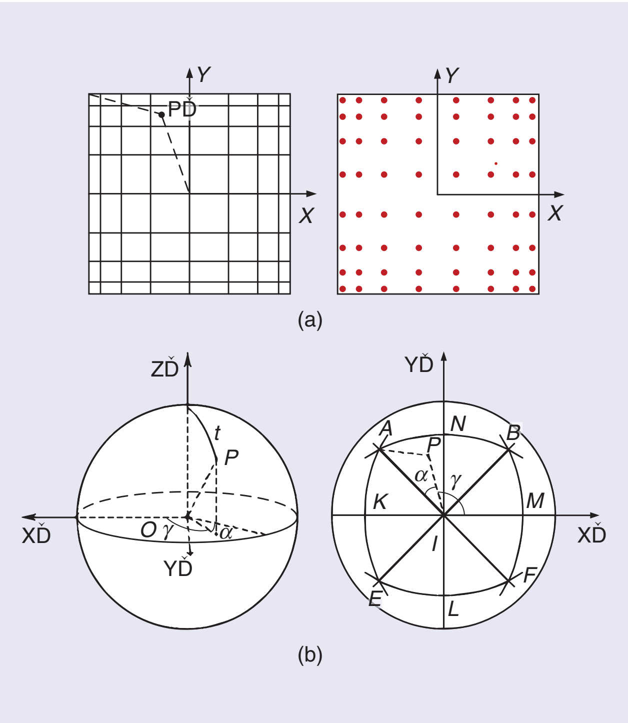

FIGURE 4. (a) The gradually varied sampling strategy and (b) the polar coordinate transformation.

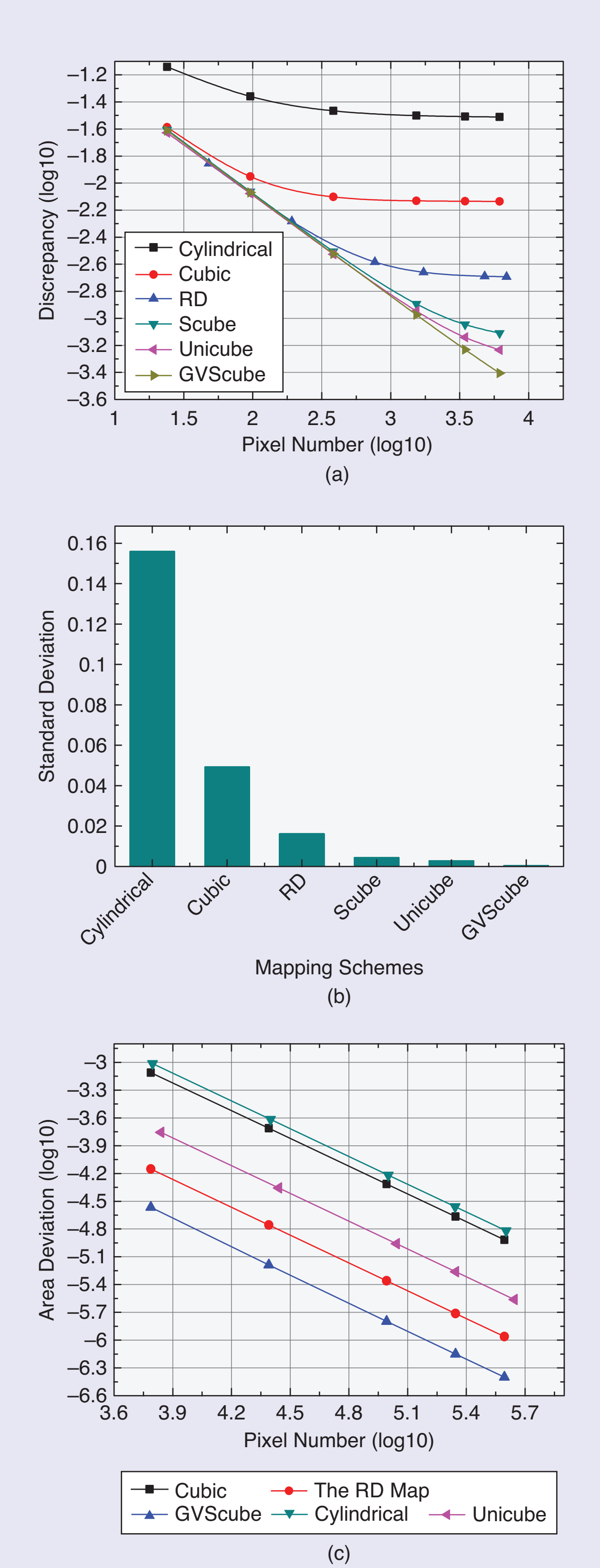

FIGURE 5. (a) Discrepancy values of different mapping schemes, (b) the standard deviation of point distribution, and (c) the area deviation.

the sphere model, and the process $(\rho, \theta) \rightarrow (\alpha, t)$ is performed in cube-to-sphere mapping.

By applying the Newton–Raphson method, we set the nonlinear function to be

$$F(\alpha) = S - (H + \alpha - 2\pi/3) \times R^2.$$

By setting the initial value of polar angle α to be θ, an approximate value of α is generated after several iterations. After a dihedral angle α is obtained, the radius t can be easily calculated by the last two equations in (1).

GRADUALLY VARIED SAMPLING ON THE CUBE MODEL

Although Scube projection provides relatively good sampling uniformity compared with other mapping schemes, the data uniformity can be further improved. The density of Scube projection is the lowest at a latitude near $\pi/4$, and it is relatively larger at the poles and equator, where centers of faces lie. This indicates that pixels thin out near edges and crowd near the centers of faces. As mentioned previously, a traditional subdivision scheme partitions every face of the cube model uniformly, and after Scube projection, these points are concentrated at centers of faces, unlike a cube map. To solve the problem, we modified the subdivision scheme by using gradually varied sampling, which squeezes sample points on every face of the cube model to the edges [Figure 4(a)].

In the GVScube method, we implement trimming mapping on each face of the cube model at the beginning of cube-to-sphere mapping, and in sphere-to-cube mapping, we determine the cell where target point P' lies via inverse trimming mapping.

In the beginning of cube-to-sphere mapping of a Scube projection, Cartesian coordinates (x, y) are obtained by directly calculating the Euclidean distance between P' and I'. Additionally, in GVScube, we map point (x, y) to point (x', y'), and use (x', y') instead in the following steps of cube-to-sphere mapping. Formulas are presented in (3), where r is the inscribed radius of the cube model, which is set to be $\sqrt{\pi/6}$, since we defined the side length of the cube model as $\sqrt{2\pi/3}$:

$$\begin{cases} x' = r \times \sin(x \times \pi/8r)/\sin(\pi/8) \\ y' = r \times \sin(y \times \pi/8r)/\sin(\pi/8). \end{cases} \quad (3)$$

For sphere-to-cube mapping, the only change is to map the Cartesian coordinates (x, y) to (x', y') after transforming polar coordinates to Cartesian coordinates. We use the inverse mapping of (3), and the calculation of cell index (m, n) is based on the mapped point (x', y') in the same way.

By implementing the GVScube projection, a much more uniform distribution is obtained, and the increased time consumption is negligible compared to that of the whole mapping scheme. Moreover, our proposed method is better than a skew great circle subdivision scheme for the RD map, which requires laborious calculation to get the best skew factor.

Gradually varied sampling is applied in the process of mapping and is robust under different resolutions.

EXPERIMENTAL RESULTS

Three metrics are used to analyze mapping schemes: sampling uniformity, density variation by latitude, and area deviation. Sampling uniformity is determined by calculating the variation of distances between sampling points. Density variation by latitude intuitively shows the spatial pixel density at different latitudes, which also helps us examine the distribution more conveniently. Area deviation gives the standard deviation of the pixel area.

1) *Sampling uniformity*: We used an approach to measure the uniformity of the point distribution [9]. We used the formula for discrepancy calculation from [15]. A smaller discrepancy indicates better uniformity. As shown in Figure 5(a), Snyder's equal-area projection with GVScube has a smaller discrepancy than each of the other three mapping schemes.
2) *Density variation of point distribution*: The density variation of sample points directly indicates sampling uniformity. With lower fluctuation of density, a point distribution can be considered more uniform. We calculated the sampling density of different mapping schemes by latitude. Due to symmetricity, the latitudes we chose span the upper half of the sphere. We compared the standard deviation of densities, as seen in Figure 5(b), and found that GVScube projection has the lowest pixel-density fluctuation among all the mapping schemes for high latitudes to the equator.
3) *Area deviation*: A smaller standard deviation of area implies a more faithful record of a spherical scene. Since pixel areas of ordinary images or videos are the same, spatial pixel-area variation leads to fluctuation in mapping area. Regions on the sphere with smaller spatial pixel areas are mapped to larger areas on the plane, while regions with larger spatial pixel areas are mapped to smaller areas. As in Figure 5(c), GVScube projection has the smallest area deviation, which implies the smallest area distortion when recording panoramic scenes.

CONCLUSION

Mapping efficiency, though seldom discussed as an important problem in the field of panoramic video generation, does have a significant effect on visual quality and video transmission. Traditional mapping schemes, such as cylindrical mapping, cause data redundancy around some regions of a spherical model, with an oversampling effect around others, which causes undesirable visual experiences.

Cube-Snyder projection is used to generate panoramic videos in this article. By selecting a cube model, rectangular-based videos are generated for the convenience of encoding. Despite its encoding-friendly property, pixel distribution is highly uniform, and every pixel has almost the same area if the cube model is uniformly subdivided. Moreover, we proposed GVScube projection, which subdivides the cube model proportionally, according to a sine function, and further improves uniformity.

ABOUT THE AUTHORS

Yibo Fan (fanyibo@fudan.edu.cn) is an associate professor at the College of Microelectronics of Fudan University, Shanghai, China.

Yize Jin (14210720072@fudan.edu.cn) earned his M.S. degree in microelectronics and solid electronics from Fudan University, Shanghai, China, in 2017.

Zihao Meng (18212020028@fudan.edu.cn) earned his B.S. degree in microelectronics from Fudan University, Shanghai, China, in 2018.

Xiaoyang Zeng (xyzeng@fudan.edu.cn) is a full professor at the College of Microelectronics of Fudan University, Shanghai, China.

REFERENCES

[1] S.E. Chen, "QuickTime VR—An image-based approach to virtual environment navigation," in *Proc. 22nd Annu. Conf. Computer Graphics and Interactive Techniques (SIGGRAPH)*, 1995, pp. 29–38.
[2] H. Y. Shum, S. B. Kang, and S. C. Chan, "Survey of image-based representations and compression techniques," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 11, pp. 1020–1037, 2003.
[3] L. Brun and A. Gasparini, "Enabling 360° visual communications: Next-level applications and connections," *IEEE Consum. Electron. Mag.*, vol. 5, no. 2, pp. 38–43, Apr. 2016.
[4] P. Rosedale, "Virtual reality: The next disruptor: A new kind of worldwide communication," *IEEE Consum. Electron. Mag.*, vol. 6, no. 1, pp. 48–50, Jan. 2017.
[5] Y. Yuan, "Changing the world with virtual/augmented reality technologies," *IEEE Consum. Electron. Mag.*, vol. 6, no. 1, pp. 40–41, Jan. 2017.
[6] B. Markwalter, "Entertainment and immersive content: What's in store for your viewing pleasure," *IEEE Consum. Electron. Mag.*, vol. 4, no. 1, pp. 83–86, Jan. 2015.
[7] Wikipedia, "Equirectangular projection." Accessed on: Jan. 2017. [Online]. Available: https://en.wikipedia.org/wiki/Equirectangular_projection
[8] K.-T. Ng, S.-C. Chan, and H.-Y. Shum, "Data compression and transmission aspects of panoramic videos," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, pp. 82–95, 2005.
[9] C. W. Fu, L. Wan, and T. T. Wong, "The rhombic dodecahedron map: An efficient scheme for encoding panoramic video," *IEEE Trans. Multimedia*, vol. 11, pp. 634–644. June 2009.
[10] T. Y. Ho, L. Wan, C. S. Leung, P. M. Lam, and T. T. Wong, "Unicube for dynamic environment mapping," *IEEE Trans. Vis. Comput. Graph.*, vol. 17, no. 1, pp. 51–63, Jan. 2011.
[11] M. Yu, H. Lakshman and B. Girod, "A framework to evaluate omnidirectional video coding schemes," in *Proc. IEEE Int. Symp. Mixed and Augmented Reality*, 2015, pp. 31–36.
[12] L. Wan, C. S. Leung, P. M. Lam, S. K. Mak, and T. T. Wong, "Isocube-based spherical wavelet for image-based relighting," *IET Comput. Vision*, vol. 5, no. 3, pp. 151–163, May 2011.
[13] T. Y. Ho, L. Wan, C. S. Leung, P. M. Lam, and T. T. Wong, "Unicube for dynamic environment mapping," *IEEE Trans. Vis. Comput. Graph.*, vol. 17, no. 1, pp. 51–63, Jan. 2011.
[14] J. P. Snyder, "An equal-area map projection for polyhedral globes," *Cartographica*, vol. 29, pp. 10–21, 1992.
[15] J. Cui and W. Freeden, "Equidistribution on the sphere," *SIAM J. Sci. Comput.*, vol. 18, no. 2, pp. 585–609, 1997.