September 15, 2017

Virtual Reality Video Compression

Andrew Ian Russell

Matthew Milton Pharr

Evan Rapoport

Virtual Reality Video Compression

Abstract

Uncompressed light field files tend to be big, and may take gigabytes of space for storing a small portion of a scene. However, there can be redundancy in the data. Therefore, compressing the data can result in significant reduction of data size for transmission and/or storage. Further, interpolating uncaptured light-fields can be based on two blurry images generated from a frame of streaming video.

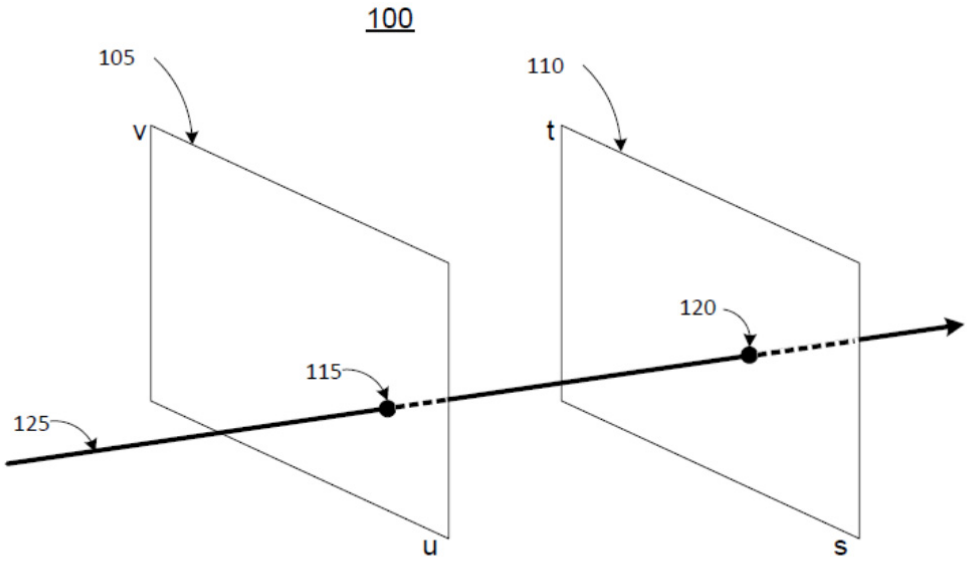


FIG. 1

FIG. 1 illustrates an example light-field coordinate system 100.

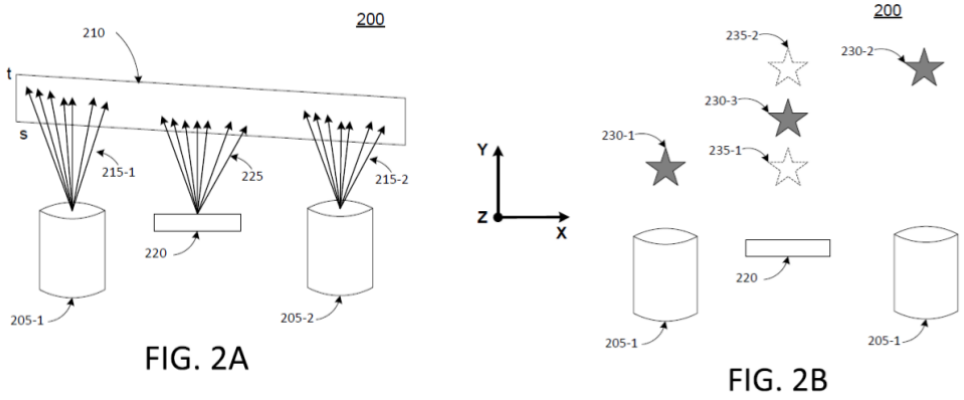
As shown in FIG. 1, the coordinate system on a first plane 105 is (u, v), hereinafter u-v plane 105 and on a second plane 110 is (s, t), herein after s-t plane 110. An oriented line 125 is defined by connecting a point 115 on the u-v plane 105 to a point 120 on the s-t plane 110. This representation is referred to herein as a light-field. The light-field represents the beam of light entering one quadrilateral and exiting another quadrilateral. In other words, the light-field represents the beam of light entering the u-v plane 105 and exiting the s-t plane 110 (or vice versa).

The s-t plane 110 can be at a lens of a light-field camera and the u-v plane 105 can be a photo sensor of the light-field camera. Accordingly, lines representing light-fields (e.g., oriented line 125) can be parameterized by two points, or by a point and a direction. Using point and a direction can be used for constructing light fields representing a three-dimensional object in a two dimension image (e.g., as done by a light-field camera).

Light-field cameras can sample a four-dimensional (4D) optical phase space or light-field and in doing so capture information about the directional distribution of the light rays. This information, as captured by a light-field camera, may be referred to as the light-field or radiance. A light-field can be a 4D record of all light rays in three-dimensions (3D). Radiance can describe both spatial and angular information, and can be defined as density of energy per unit of area per unit of stereo angle (in radians). A light-field camera captures radiance. Therefore, light-field images originally taken out-of-focus may be refocused, noise may be reduced, viewpoints may be changed, and other light-field effects may be achieved. Light-fields may be captured with a conventional camera as well. For example, MxN images of a scene can be captured from different positions with a conventional (e.g. digital) camera. If, for example, 8x8 images are captured from 64 different positions, 64 images are produced. The pixel from each position (i, j) in each image are taken and placed into blocks, to generate 64 blocks.

Captured light-fields can be saved and/or stored as a 2D image that contains an array of tiles or image portions. In other words, a 2D image for the u-v plane 105 and a 2D image for the s-t plane 110 can be saved and/or stored as the light field. The 2D slices of light-fields are equivalent to conventional pictures. Accordingly, the uncompressed files tend to be big, and may take gigabytes of space for storing a small portion of a scene. In other words, a point of view (e.g., as captured by one camera) of a scene can include a large amount of pixel

(e.g., RGB, YUV and the like) data. However, there can be redundancy in the data (e.g., all rays starting from a surface point can have approximately the same radiance). Therefore, compressing the data can result in significant reduction of data size for transmission (e.g., streaming) and/or storage.



FIGS. 2A and 2B illustrate a portion of an image capture system 200.

As shown in FIG. 2A, the image capture system 200 includes cameras (or photo sensors) 205-1 and 205-2. An s-t plane 210 can be a plane representing a scene or a portion of a scene to be captured by cameras (or photo sensors) 205-1 and 205-2. Images captured by cameras (or photo sensors) 205-1 and 205-2 can represent light-fields 215-1 and 215-2. According to one technique, the image capture system 200 does not capture all light-fields of the s-t plane 210 representing a scene. In FIG. 2A, the uncaptured light-fields are shown as light-fields 225 as focused on a camera void 220. Accordingly, the scene represented by light-fields 225 at a point in time may be interpolated in order to generate a full 360 degree representation of the scene.

As shown in FIG. 2B, the image capture system can be configured to capture (e.g., as an image or image representing a light-field) an object 230-1 and 230-2 associated with the scene in the s-t plane 210 using cameras (or photo sensors) 205-1 and 205-2. The object 230-

1, 230-2 (and the scene as a whole) is illustrated as having movement in the Y-direction. The movement in the Y-direction is not the result of movement of the object 230-1, 230-2, instead, the movement in the Y-direction is due to the different position (and thus point of view) of the cameras (or photo sensors) 205-1 and 205-2.

The object 230-3 (e.g., associated with the uncaptured light-fields or light-fields 225) is illustrated as having movement or point of view movement as well. When interpolating images associated with a light-field for the uncaptured light-fields associated with light-fields 225, this movement can be ignored or assumed to be linear in relation to object 230-1 which is illustrated as object 235-1. Alternatively, when interpolating images associated with a light-field for the uncaptured light-fields associated with light-fields 225, this movement can be ignored or assumed to be linear in relation to object 230-2 which is illustrated as object 235-2. In either case, an image including object 230-3 can be a copy of an image including one of object 230-1 or 230-2.

However, the interpolated image (e.g., including object 230-3) can be based on the motion where the object 230-3 is shifted in the Y-direction as compared to object 230-1 and 230-2. Accordingly, interpolated image(s) configured to represent (or include) light-fields 225 can be based on at least one co-planar image (e.g., images associated with object 230-1 and/or 230-2 and/or captured images in the s-t plane 210) and a motion vector.

A first blurry image is generated from a frame of streaming video. For example, an image filter can apply a directional blur in two dimensions. Accordingly, the image filter can have two (e.g., x and y, u and v, s and t, and the like) variable inputs. For the first blurry image, the two variable inputs can be pre-configured based on a distance between cameras on a camera rig. Then a motion field is estimated from the first blurry image. For example, a motion vector can be determined based on a predicting the pixel values in a block of a picture

relative to reference samples in neighboring blocks of the same picture. A difference in position can be determined. The difference in position can be used to determine the motion vector.

A second blurry image is generated from the frame of streaming video based on the motion field. The second blurry image can be generated using the image filter. The two input values can be based on a difference between the two variable inputs used to generate the first blurry image and two input values calculated based on the motion vector. Then a motion field is estimated from the second blurry image. For example, the motion vector can be determined based on a predicting the pixel values in a block of a picture relative to reference samples in neighboring blocks of the same picture. A difference in position can be determined. The difference in position can be used to determine the motion vector.

If a motion and linear interpolation deviation is above threshold. For example, a motion vector for a linear interpolation can be determined as discussed above with regard to FIG. 2B. This linear motion vector can be compared to a motion vector calculated based on the motion field (e.g., the two input values used to generate the second blurry image). If the motion and linear interpolation deviation is above threshold a light-field is interpolated based on the estimated motion field from second blurry image. For example, a new light-field is generated between two light-fields based on data associated with at least one of the adjacent light-fields and the estimated motion vector. The threshold can be based on a distance between cameras on a camera rig.

As discussed above, the amount of light field data representing a frame of video can be large. Accordingly, a portion of the light-field data can be removed. Selecting the light-field data to be removed can be based on a fixed routine (e.g., every other light-field) or based on a varied determination. If a bandwidth associated with a frame is below a target or threshold

processing continues information about marked light-fields is determined based on included light-fields. For example, a motion vector based on an adjacent light-field can be calculated for each of the light-fields marked for removal. The motion vector can be calculated based on a difference (e.g., in position and/or color) between the adjacent light-field and the light-field marked for removal. An encoded frame can be streamed together with the information. For example, the frame encoded above can be packaged together with the information about marked light-fields in a data-packet to be streamed to a computer associated with a requesting viewer. After decoding, removed light-field(s) are generated based on the information and other light-field(s). For example, the information can include a motion vector and an indication of the light-field (e.g., adjacent light-field) the motion vector is based on. The removed light-field can be interpolated or rebuilt based on the motion vector and the associated light-field and inserted into a corresponding position in the light-field data.

A light-field data can be retrieved in the default u-v-s-t storage order. The light-field data can be in the reordered to an s-t-u-v order. The reordered data can be encoded first the s-t plane and then the u-v plane. Encoding the light-field data is a lossy process. However, encoding encoding the s-t plane before the u-v plane can result in less lossy (e.g., fewer compression artifacts) compressed data when encoding the s-t plane after the u-v plane. Therefore, a rendered frame based on a reordered light-field data compression can be of relatively higher quality. When decoded, the light-field data can be in the reordered to a u-v-s-t order before rendering.

Semantically interesting things in a scene can be indicated as hot, whereas non-interesting things can be indicated as cold. This may require contextual understanding of the scene. In one example implementation, a creator or supplier (e.g., a director or producer) of a video can code or otherwise indicate the hot spots. In another example implementation, content can be

indicated as of interest measuring movement (e.g., moving likely more interesting). For example, in a concert a guitarist moving around the stage is likely of more interest. In yet another example implementation, track what users look at over time can be tracked (e.g., anonymously tracked). For the computer implemented algorithms, a higher value can be assigned based on more interest (e.g., the higher a value, the more the interest).

Hot spots of a scene having missing light-fields can be interpolated using one of the techniques described above. Video is encoded including the interpolated light-field(s). For example, the frame including the interpolated light-fields can be encoded using a standard protocol or one of the techniques described herein.

Texture compression is a lossy compression technique that results in a relatively fast decompression (e.g., as compared to other compression techniques. Texture compression can include fixed rate compression of small block sizes using standardized techniques (e.g., Adaptive Scalable Texture Compression (ASTC)). ASTC can support bit rates ranging from 8bpp down to less than 1bpp in very fine steps in order to control speed vs. quality. In ASTC images can be partitioned into fixed-size blocks, which are encoded as vectors of, for example, 128 bits. Color spaces are then specified by pairs of points (e.g., coefficients) that define line segments in a color space. Each compressed block can have the same size for all blocks. A list of pointers, one for each block, can point to the memory location of the compressed data for the block. In the UV color space, each UV coordinate from the corresponding NxM block includes a coordinate with a location based on the block number and the number of compressed bits in the block. Accordingly, decompression of a block of an image or frame of video can be as needed (e.g., based on what is being rendered) by accessing the block using the calculated memory location.