A Geometric Space-View Redundancy Descriptor for Light Fields: Predicting the Compression Potential of the JPEG Pleno Light Field Datasets

Márcio P. Pereira*, Gustavo Alves*, Carla L. Pagliari*†, Murilo B. de Carvalho*§, Eduardo A. B. da Silva* and Fernando Pereira‡

*PEE/COPPE/UFRJ, Universidade Federal do Rio de Janeiro, Rio de Janeiro, Brazil

†PEE/PGED/IME, Instituto Militar de Engenharia, Rio de Janeiro, Brazil

‡Instituto Superior Técnico, Universidade de Lisboa - Instituto de Telecomunicações, Lisbon, Portugal

§TET/CTC, Universidade Federal Fluminense, Niterói, Brazil

***Abstract*—The representation of data in terms of its statistical properties is valuable in many applications. This work uses statistics obtained from 4D scene geometry to characterize, in terms of redundancy, the content produced by lenslet-based light field cameras and by high-density arrays of cameras for the JPEG Pleno Call for Proposals on Light Field Coding. This paper proposes a novel so-called geometric space-view redundancy (GSVR) descriptor, which is able to characterize the amount of redundancy in light fields thus bringing information about the trade-offs involved in effectively exploring redundancy for efficient coding. The redundancy is here measured by the probability, for each block size and range of views, that the image of a given 3D point belongs to the block in all views. Therefore it is a descriptor that has application on dataset selection and encoder control and optimization. The JPEG Pleno datasets are analyzed in terms of the GSVR descriptor in all views.**

I. Introduction

Recently, the JPEG has issued a call for proposals on light field coding technologies, namely JPEG Pleno Call for Proposals on Light Field Coding [1]–[3]. It requests contributions for coding solutions in the area of light fields encompassing, among other things, both lenslet-generated light field images and light field images obtained using high density 2D camera arrays (HDCA). The coding solutions, as well as other contributions, are part of the first steps towards JPEG Pleno specifications.

Aiming to provide useful data for light field encoding schemes, lenslet and HDCA test materials were made available together with the document JPEG Pleno Call for Proposals: Submission Process Details [4]. This work contributes towards the JPEG Pleno effort by introducing the geometric space-view redundancy (GSVR) descriptor. Its main idea is to characterize light field images datasets in terms of 4D geometric redundancy that can be exploited by prospective coding schemes. The GSVR is related to the probability that a pixel inside a block in one view will have a counterpart, in the corresponding block, in other views. Therefore, it is related to the amount of inter-view redundancy provided by the 4D geometry of the light field. In addition, this work shows that the GSVR values obtained for the HDCA datasets are very different from the ones obtained for the lenslet datasets, suggesting its usefulness to classify datasets according to their 4D geometries.

978-1-5090-3649-3/17/$31.00 2017 IEEE

The remaining of this paper is organized as follows. Section II describes the JPEG Pleno test materials. The proposed GSVR is described in Section III. The results are discussed in Section IV and Section V presents the conclusions.

II. JPEG Pleno test materials

In general, datasets are designed with a specific purpose. The JPEG Pleno test materials were designed to address prospective coding technologies for content produced by plenoptic content [1]. It includes richer imaging modalities, such as depth, omnidirectional, point cloud, holographic and light fields, that are being offered by new capturing systems. In this work, we analyze the light field imaging modality in the JPEG Pleno CfP [4], namely lenslet and HDCA datasets.

A plenoptic function [5] parameterizes each light ray of a point in space with its 3D location, its direction of arrival, and its time-varying power spectral density. This 7D function can be reduced to a 4D representation, where each color component of a light ray is defined by its intersection with a camera plane (u, v) and a focal plane (x, y). A 4D light field [6] image captures the rays of light from illuminated objects. Two ways of sampling these fields of light (light fields) [5] are recording either lenslet-based images or conventional images of the object from a huge number of viewpoints, generating a massive amount of data. In both cases, the light field can be analyzed as 2D arrays of images, with viewpoints evenly spaced on a 2D grid indexed with u and v, where x and y are the image coordinates in each (u, v) view [5], [6]. Clearly, an efficient compression scheme is essential to reduce this large amount of data.

A. Lenslet-based Datasets

In the lenslet case one uses an imaging technology that is able to capture a 4D light field by means of multiplexing its data into the camera's 2D sensor resolution [7]. This

Fig. 1: Lenslet camera dataset images (central views).

multiplexing is done through an array of microlenses placed between the main lens and the camera sensor. The array of microlenses samples the 3D points in space. Each microlens generates a microimage formed by the light rays emitted by the corresponding 3D point in several directions. The selection of the pixels with the same coordinates relative to every microlens forms a subaperture image. These subaperture images are the microlens perspective of the scene captured through the main lens.

The central views of the light fields produced by the lenslet-based camera system are displayed in Figure 1. From left to right and top to bottom they are: Bikes (I01), Danger de Mort (I02), Stone Pillars Outside (I04), Fountain&Vincent 2 (I09) and Friends 1 (I10).

The lenslet dataset is part of the EPFL dataset presented in [8], where the images were captured using a Lytro Illum B01 (10-bit) [9] light field camera. In the JPEG Pleno CfP [1] the Matlab implementation of the Light Field Toolbox v0.42 [10] is employed to convert lenslet light field camera raw sensor data to subaperture images according the process described in [1], including the demosaicing and devignetting pre-processing steps. The subaperture images, containing samples from the microlenses, are stacked forming a 5D array of $15\times15\times625\times434\times4$ dimensions. The JPEG Pleno team inserted a column with zeros in the subaperture images so that their spatial resolution becomes 626×434. In this work, we opted to discard the first two and last two rows and columns of the subaperture images, as they are blackened due to devignetting, leading to a matrix of 11×11 views. The natural outdoor images from this dataset, pictured in Figure 1, present different levels of spatial information, objects at different depths and repetitive patterns.

B. HDCA Datasets

Another way of capturing light fields is through 2D arrays of conventional cameras [6]. One needs multiple cameras, uniformly distributed on a planar surface, to capture long-baseline light fields of a dynamic scene. The sampling density of the light field in space has to be sufficiently large so that it can be recovered from its samples, thus requiring that the spacing among adjacent cameras to be sufficiently small [11].

The central views of the light fields produced by the HDCA system are displayed in Figure 2. From left to right and top to bottom they are: TableTop I (Set 2), TableTop II (Set 6), Lightfield Production (Set 9) and Workshop (Set 10). All

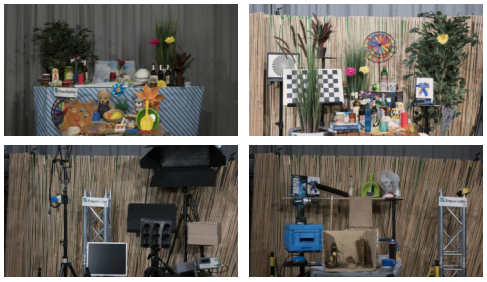


Fig. 2: HDCA dataset images (central views).

lenslet images are of natural outdoor scenes, while the HDCA images are of indoor (studio) scenes.

The original HDCA 14-bit test materials, with 101 horizontal images and 21 vertical images of 7956×5304 spatial resolution were made available in a 10-bit version already pre-processed, which included global registration, as well as resolution and chroma down-sampling [1]. The final spatial resolution of 3840×2160 pixels is used in this work. The geometry of acquisition employed an horizontal step size of 4 mm (distance between adjacent cameras in the horizontal direction) and a vertical step size of 6 mm (distance between adjacent cameras in the vertical direction). The indoor images present different levels of detail, specularities, regular patterns and objects at different depths.

III. Proposing a Geometric Space-View Redundancy descriptor

In image and video coding several techniques exploit spatial and temporal redundancy targeting bitrate reduction. In general, a block intra prediction scheme may use previously encoded block samples from its spatial neighborhood blocks to predict its block samples. For example, the current state-of-the-art video coding standard HEVC [12] extracts the upper and left boundaries blocks samples to perform the intra prediction. Motion modeling plays a fundamental role in the same coding standard [12], employing block-based translational motion prediction. A block-based matching is performed at the target frame, taking the current frame as reference, producing a motion vector and a motion compensated block residue. In general, the similarity (correlation) among pixels within a small neighborhood tend to be high. Image compression schemes use large blocks, to exploit redundancy, in regions presenting a low degree of texture (spatial complexity), whereas smaller blocks are employed in regions that exhibit high spatial complexity.

Lenslet-based light field cameras and HDCA systems tend to produce images that present a strong degree of correlation among neighboring views. Hence, this redundancy in information can be exploited to compress the images. More specifically, the view geometry of lenslet-based cameras and HDCA systems provide additional redundancy that can be exploited for image coding purposes.

A. GSVR Descriptor Definition

As this work aims to characterize the JPEG Pleno datasets to contribute with the current effort on light field image coding, one key aspect is the largest region in the 4D space that

presents 4D geometric redundancy worthy to be exploited for encoding purposes.

In order to estimate the largest region that presents 4D geometric redundancy, the proposed method takes 4D image blocks of $L \times L \times K \times K$ dimensions, with $L \times L$ being the block size inside each view and with $K \times K$ being the number of horizontal and vertical views in the region. Let $XY(u, v)$ be an $L \times L$ square region in view (u, v). For each 4D block, the method evaluates the probability that a point in 3D space, with a projection onto $XY(u_0, v_0)$ will also have a projection onto another view $XY(u_1, v_1)$ of the same 4D block.

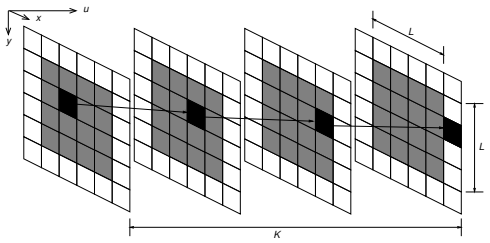


Fig. 3: A point moving across four 4×4 blocks (gray area) in four horizontal views.

Figure 3 illustrates the idea, in 3D, that is, considering a light field with only horizontal views (i.e., $v = 0$). There, a point (represented as a black pixel) moves across four blocks, of size 4×4, (depicted in gray) belonging to four different horizontal views ((x, y) planes) along the u axis. In this case $L = 4$ (block size) and $K = 4$ (number of horizontal views). The first 4×4 block belongs to view $u = 0, v = 0$, the second to view $u = 1, v = 0$, the third to view $u = 2, v = 0$ and the fourth to view $u = 3, v = 0$. The set of gray blocks correspond to the 4×4×4 space-view 3D block. Note that the black pixel belongs to the 4×4 spatial block in only three views, falling outside it in the fourth view. By computing, for each pixel, the number of views in which it is still inside the gray area, one can estimate the desired probability.

The example pictured in Figure 4 details the above reasoning by illustrating the trajectories of 3 different points (A, B and C) along the horizontal and vertical views. Figure 4 shows 16 spatial blocks belonging to 16 views in the (u, v) plane, where 3 points (A, B and C) are mapped onto the views. Point A, located in a (x, y) coordinate inside view $u = 0, v = 3$, appears horizontally displaced by dx_1 in view $u = 1, v = 3$. In view $u = 2, v = 3$ the cumulative horizontal shift is of $dx_1 + dx_2$. The value of dx_3 shows that point A is mapped outside the spatial block in view $u = 3, v = 3$. Point A is also inside the block in view $u = 0, v = 2$, appearing displaced by the vertical shift dy_1. The cumulative shift $dy_1 + dy_2$ shows that point A is mapped outside the spatial block in view $u = 0, v = 1$. The cumulative vertical shift $dy_3 + dy_4$ shows that point B is mapped outside the spatial block in view $u = 1, v = 1$. Point B presents only vertical shifts, while point C appears horizontally displaced inside the spatial block along all displayed views.

One way to compute these displacements (shifts) is by using disparity estimation. The correspondence method will provide

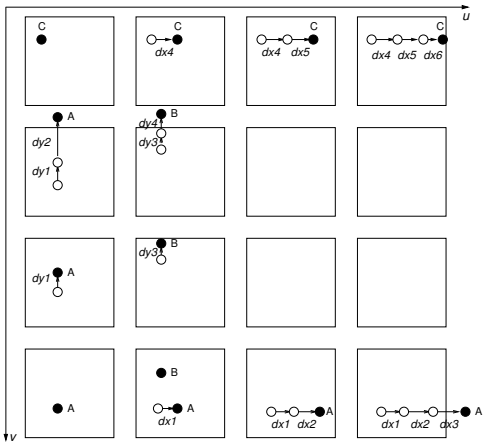


Fig. 4: Points moving across blocks within the horizontal and vertical views.

the displacement (disparity) values for each pixel inside a block in the (x, y) plane, belonging to a view in the (u, v) plane. Section III-B describes the disparity estimation process.

Therefore, one can obtain the geometric space-view redundancy (GSVR) descriptor, by following the steps depicted in Figure 5.

Fig. 5: Block diagram to calculate the GSVR(P) descriptor.

The GSVR will be formally described in what follows. Let $d(n, m, p, q)$ be the disparity values at spatial position (n, m) for the p-th horizontal view and the q-th vertical view. The probability that a point at any position (n^*, m^*, p^*, q^*) inside a block of size $L \times L \times K \times K$ will have a geometrically related counterpart inside the same block can be estimated by:

$$P_h(L, K) = \frac{1}{L^2 K(K-1)} \sum_{n=0}^{L-1} \sum_{m=0}^{L-1} \sum_{p=0}^{K-1} \phi_L \left(\sum_{q=0}^{K-2} m + d(n, m, p, q) \right) \quad (1)$$

$$P_v(L, K) = \frac{1}{L^2 K(K-1)} \sum_{n=0}^{L-1} \sum_{m=0}^{L-1} \sum_{q=0}^{K-1} \phi_L \left(\sum_{p=0}^{K-2} n + d(n, m, p, q) \right) \quad (2)$$

$$P_c(L, K) = P_h + P_v - P_h P_v \quad (3)$$

where $\phi_L(x) = H(x) - H(x - L)$, and $H(x)$ is the Heaviside step function.

Let $C(L, K, P) = \{(L, K) | P_c(L, K) = P\}$ be the set of points of the contour curve at probability P. Let $L = \alpha(P) K$ be the best fitting line obtained by a linear regression on the points of $C(L, K, P)$. We define the geometric space-view redundancy descriptor, GSVR(P), as the angular coefficient (α) of this line given by (4):

$$\text{GSVR}(P) = \alpha(P) \quad (4)$$

In Section IV we make a detailed analysis of the GSVR(P) values for the JPEG Pleno datasets. In order to do so, in the next Subsection III-B we briefly describe how the disparities are estimated for the lenslet and HDCA datasets.

B. Disparity Estimation

In this work, two disparity estimation techniques are employed. As the views are horizontally and vertically rectified, given a pixel in a reference view, its corresponding pixels from the same row of the matrix are horizontally displaced, while its corresponding pixels from the same column of the matrix are vertically displaced. Provided the point is not occluded, for each point in a view the horizontal and vertical disparities are the same. We take advantage of this fact by applying for both disparity estimators an horizontal-vertical consistency check. In the lenslet case, these displacements have to be the same, since adjacent views have the same, small, horizontal and vertical baseline. For the HDCA system, where the horizontal and vertical baselines differ, the ratio between the baselines is multiplied to the vertical disparities before the horizontal-vertical consistency check is performed. The first disparity estimator is a block matching algorithm using the HDCA image matrices of size 101×21, performing the following steps:

- Homologous points are determined by selecting the lowest matching cost for three block sizes of 64×64, 32×32 and 16×16, encompassing the pixel(s) under evaluation;
- The block-based matching is performed using a quadratic cost function with quarter-pixel accuracy, achieved by upsampling the views using a bicubic interpolation;
- For disparity maps computed using all block sizes, all horizontal disparities are checked against their corresponding vertical disparities (consistency check) and vice-versa;
- This consistency check, allowing a small pixel error threshold, is used to help to detect low-confidence estimates;
- In the case of low-confidence matches, the smaller block size disparity map looks for the disparity value assigned to the same point on the immediately larger block size;
- Persistent low-confidence matches after these rounds of consistency check are labeled as unknown.

The second disparity estimator is applied to both HDCA and lenslet datasets. As every single 3D scene point is projected onto a single line in the epipolar plane image (EPI) [13], disparity computation can be performed in EPIs, emphasizing a fundamental advantage of the 4D light field structure. Considering that a point in 3D space is projected onto a line in both EPIs $S_{u,x}$ and $S_{v,y}$, the slopes of resulting lines are the horizontal and vertical disparities. Therefore, one can estimate the depth of a point by obtaining the slope of the lines corresponding to that point in the EPIs.

The pipeline for this second technique of disparity estimation is the same for HDCA and lenslets images, except for the number of $U \times V$ blocks analyzed at a time. For the 11×11 lenslet views the entire set of views is used at a time. As for the HDCA there are more columns than rows of views, 21×21 blocks of views are analyzed at a time. We use an overlap of 1 column of views in order to obtain exactly 5 blocks of views from the original 101×21 set. The pixels from the views form $Y \times V$ arrays of $U \times X$ blocks or $X \times U$ arrays of $V \times Y$ blocks. The second disparity estimator performs the following steps:

- The well-known Canny edge detector [14] associated with the Hough transform [15] are applied to the $U \times X$ and $V \times Y$ blocks to detect the lines. The horizontal and vertical disparities are the slopes of the dominant lines orientation;
- The consistency check for the disparity data is done by comparing the values obtained from the $U \times X$ and $V \times Y$ blocks allowing a small pixel error threshold.

For the HDCA datasets, the remaining unknown disparity values resulting from the application of the first method are replaced by the values of the disparity maps generated by the second method, whenever these are not unknown. Consequently, the resultant final dense horizontal and vertical disparity maps may present a small amount of disparity information labeled as unknown, which are points that have unreliable disparity values (probably due to occlusions, to lack of texture, etc), that are discarded during the statistical analysis performed in Section IV.

IV. Analyzing the Light Field Datasets

In this Section we analyze both the HDCA and lenslet datasets of the JPEG Pleno CfP using the GSVR(P) proposed in Section III.

The statistics are computed for $L = \{4, 8, 16, ..., 64\}$ for the HDCA sets and for $L = \{1, 2, 3, ..., 16\}$ for the lenslet sets, due to the different spatial resolutions. The idea behind varying L in all views (horizontal and vertical) is to statistically measure the largest region in the 4D space that presents enough 4D geometric redundancy. The statistics are computed for K varying from 1 to 16, for the HDCA datasets and for K varying from 1 to 11 for the lenslet dataset. The samples used to obtain the statistics are the valid disparity values from all lenslet and HDCA horizontal and vertical dense disparity maps of all views generated by the procedures described in Section III-B. Unknown disparity values are desconsidered.

For the sake of simplicity, equations (1), (2) and (3) refer to a 4D block of size $L \times L \times K \times K$ only. We computed the statistics, using these equations, for all 4D blocks extracted from the datasets averaging all the obtained P_c values.

Figure 6 displays the GSVR$(P) \times P$ curves of all datasets for P varying from 0.1 to 0.9. The plot shows that the HDCA datasets (Set2, Set6, Set9 and Set10) and the lenslet datasets (I01, I02, I04, I09 and I10) are assembled into two very distinct clusters, showing that GSVR$(P) = \alpha(P)$ could be used to classify lenslet-based and HDCA content. All curves monotonically increase, indicating that 3D points with large disparity values move faster across the different views and need larger spatial block sizes to remain inside a block across the views. Likewise, smaller disparity values move slower across the different views and need smaller spatial block sizes to remain inside a block across the views. In addition, a smaller spatial block size tends to be more stationary, therefore with easier redundancy exploitation. By observing the plots, we

notice that, for the lenslets, the inter-view redundancy of a 4D block can be well exploited using spatial blocks of a much smaller size than in the case of HDCA. Moreover, it is possible to observe that the HDCA datasets present very close GSVR(P) values, leading to a conclusion that the HDCA datasets may present very little diversity of scene geometries.

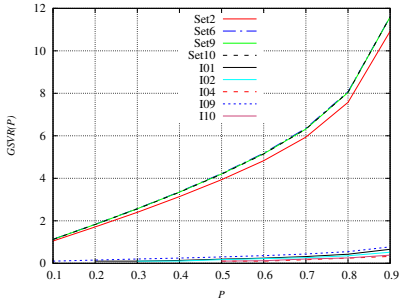


Fig. 6: GSVR(P) $\times$ P plot for all datasets.

Since the curves for the lenslet datasets are barely visible in Figure 6, Figure 7 displays the plot of GSVR(P) $\times$ P, for P varying from 0.78 to 0.98, considering only the lenslet datasets (I01, I02, I04, I09 and I10). From the curves in Figure 7, one can infer that the lenslet contents, considered by the JPEG Pleno CfP [4], have different 3D scene geometries, with different disparity statistics for the objects.

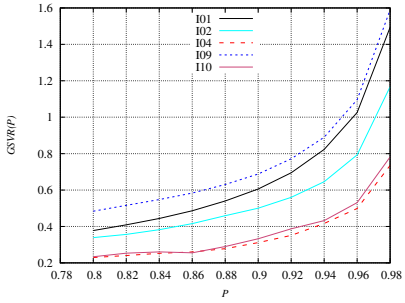


Fig. 7: GSVR(P) $\times$ P plot for the lenslet datasets.

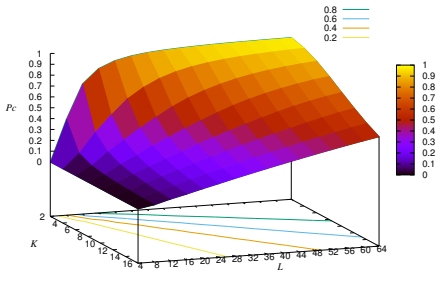


Fig. 8: Probability surface plot for TableTop I (Set 2).

Figures 8 and 9 show the probability surface plots (Eq. (3)) for Table Top I (Set 2) and Bikes (I01) images, from datasets HDCA and lenslet, respectively. Figures 10 and 11 show the iso probability contours $C(L, K, P)$ of P_c (L, K) for the same HDCA dataset Table Top I (Set 2) and the lenslet dataset Bikes (I01), respectively. It is clear that, from Figures 10 and 11, that the HDCA and the lenslet images have very different

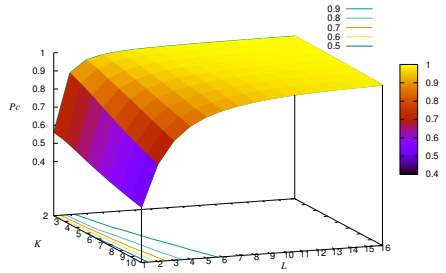


Fig. 9: Probability surface plot for Bikes (I01).

contours, which could be used to discriminate between lenslet and HDCA content.

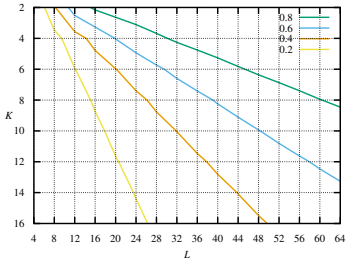


Fig. 10: Iso probability contours for TableTop I (Set 2).

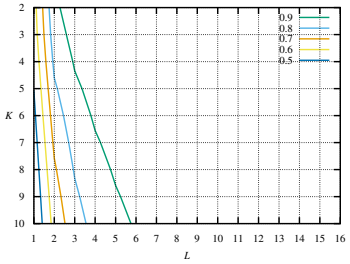
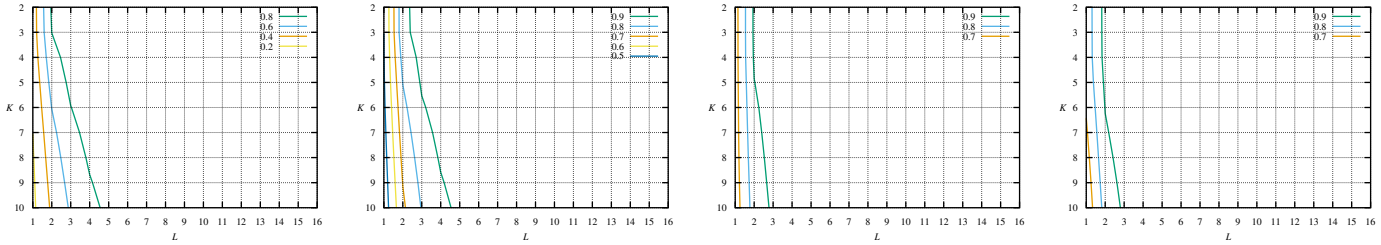


Fig. 11: Iso probability contours for Bikes (I01).

Figure 12 shows the iso probability contours for the remaining lenslet datasets (I02, I04, I09 and I10). Due to space constraints, we show only the iso probability contours of Sets 2 (TableTop I) and 10 (Workshop), as the HDCA datasets are very similar and their iso probability contours look like the ones depicted in Figures 10 and 13.

The GSVR can be used by light field images encoding schemes to establish the number of views worthy to be considered for different sizes of spatial blocks. For example, when analyzing Figure 13, that displays the probabilities for the HDCA Set 10, for a block size of 32×32 (L =32), there is a probability of 60% (blue contour) that correspondent points to be present in the other views for a block of 6 views (K =6). If one increases the block size from 32×32 to 52×52, the curve tells that for the same probability of 60%, one can to use a block as large as 10 views (K =10).

The larger the number of views with a good amount of geometric redundancy, the more the inter-view redundancy can be exploited. However, if this requires a block size that is too large, then the amount of spatial redundancy inside it may not be too large, which may impair the coding efficiency.



(a) Fountain&Vincent 2 (I09). (b) Danger de Mort (I02). (c) Friends 1 (I10). (d) Stone Pillars Outside (I04).

Fig. 12: Iso probability contours for lenslet-based images.

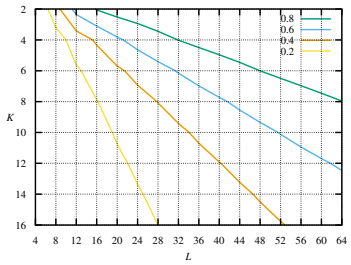


Fig. 13: Iso probability contours for Workshop (Set 10).

The slopes of the iso probability contours (Eq. (4) and Figures 10 to 13) represent the increase in one variable (e.g. L) needed to compensate for an increase in the other (K), in order to keep the probability unchanged. Therefore, the slope of an iso probability contour can be interpreted as the probability trade-off, as from the iso probability contours one can determine the inter-dependence between the two variables (L and K).

V. Conclusion

In this paper we have proposed the GSVR descriptor, that expresses the probability of a point in the 3D space to be present in all views from a given 4D space-view block. It considers all views of a light field, and can be used to characterize different light field datasets. We have computed the GSVR for the datasets of the JPEG Pleno CfP [1], [4]. The GSVR analysis suggest that the lenslet datasets have enough diversity in terms of geometric 4D redundancy, while the HDCA datasets are very similar in terms of the GSVR, indicating that they may not have enough 4D redundancy diversity to properly stress prospective JPEG Pleno encoding schemes candidates. The geometric space-view redundancy (GSVR) descriptor can provide the trade-off between the exploration of spatial and geometric inter-view redundancies. However, it is important to note that a proper assessment of the usefulness of the GSVR will require the analysis of a much more diverse set of light fields, beyond the ones of the JPEG Pleno CfP [1], [4].

Acknowledgment

The authors would like to thank CAPES, FAPERJ, CNPq, TV Globo and Centro de Pesquisa da Samsung no Brasil.

References

[1] I. J. S. N74014, "JPEG Pleno Call for Proposals on Light Field Coding," Geneva–Switzerland, January 2017.

[2] "Overview of JPEG Pleno." [Online]. Available: https://jpeg.org/jpegpleno/index.html

[3] T. Ebrahimi, S. Foessel, F. Pereira, and P. Schelkens, "JPEG Pleno: Toward an Efficient Representation of Visual Reality," *IEEE MultiMedia*, vol. 23, no. 4, pp. 14–20, Oct 2016.

[4] I. J. 1/SC29/WG1N75024, "JPEG Pleno Call for Proposals Submission Process Details," Sydney–Australia, March 2017.

[5] E. H. Adelson and J. R. Bergen, "The plenoptic function and the elements of early vision," *M. Landy and J. A. Movshon, (eds) Computational Models of Visual Processing*, 1991.

[6] M. Levoy and P. Hanrahan, "Light field rendering," in *Proc. of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*. New York, NY, USA: ACM, 1996, pp. 31–42.

[7] R. Ng, "Digital light field photography," Ph.D. dissertation, Stanford, CA, USA, 2006, aAI3219345.

[8] M. Řeřábek and T. Ebrahimi, "New light field image dataset," in *8th International Conference on Quality of Multimedia Experience (QoMEX)*, Lisbon, Portugal, 2016.

[9] "Lytro." [Online]. Available: https://www.lytro.com/

[10] "Light field toolbox." [Online]. Available: http://www.mathworks.com/matlabcentral/fileexchange/49683-light-field-toolbox-v0-4

[11] "The Stanford Multi-Camera Array." [Online]. Available: https://graphics.stanford.edu/projects/array/

[12] *ITU-T & ISO/IEC.*, "High Efficiency Video Coding, Rec. ITU-T H.265 and ISO/IEC 23008-2," 2013.

[13] R. C. Bolles, H. H. Baker, and D. H. Marimont, "Epipolar-plane image analysis: An approach to determining structure from motion," *International Journal of Computer Vision*, vol. 1, no. 1, pp. 7–55, 1987.

[14] J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 8, no. 6, pp. 679–698, Jun. 1986.

[15] P. V. C. Hough, "Machine analysis of bubble chamber pictures," in *Proceedings, HEACC 1959, CERN, Geneva, Switzerland*, vol. C590914, Sep. 1959, pp. 554–558.