Disparity compensation of light fields for improved efficiency in 4D transform-based encoders

João M. Santos[⋆,†], Lucas A. Thomaz[⋆,‡], Pedro A. A. Assuncao[⋆,‡],
Luis A. da Silva Cruz[⋆,†], Luis M. N. Tavora[‡], and Sergio M. M. Faria[⋆,‡]
[⋆]*Instituto de Telecomunicações, Portugal*
[†]*University of Coimbra, Coimbra, Portugal*
[‡]*ESTG, Polytechnic of Leiria, Leiria, Portugal*
e-mails: {joao.santos, lucas.thomaz, amado, luis.cruz, sergio.faria}@co.it.pt, luis.tavora@ipleiria.pt

***Abstract*—Efficient light field encoders take advantage of the inherent 4D data structures to achieve high compression performance. This is accomplished by exploiting the redundancy of co-located pixels in different sub-aperture images (SAIs) through prediction and/or transform schemes to find a more compact representation of the signal. However, in image regions with higher disparity between SAIs, such scheme's performance tends to decrease, thus reducing the compression efficiency. This paper introduces a reversible pre-processing algorithm for disparity compensation that operates on the SAI domain of light field data. The proposed method contributes to improve the transform efficiency of the encoder, since the disparity-compensated data presents higher correlation between co-located image blocks. The experimental results show significant improvements in the compression performance of 4D light fields, achieving Bjontegaard delta rate gains of about 44% on average for MuLE codec using the 4D discrete cosine transform, when encoding High Density Camera Arrays (HDCA) light field images.**

***Index Terms*—Light field compression, disparity compensation, 4D light field codecs**

I. Introduction

Light field (LF) imaging has been a research field since the early 1900s. Recently, especially since the emergence of handheld LF cameras, the subject of LF compression has been receiving increasing attention from the research community and also driving the standardisation efforts led by the JPEG group, through their action JPEG Pleno Light Field [1].

Light field images use 4D structures to represent more visual information than traditional 2D images, by capturing the directionality of the light rays that reach the camera sensor. This extra information is useful for post-processing operations, such as the extraction of depth maps [2], image rendering with different focal planes, depth-of-field, or viewing perspectives [3]–[5]. The usefulness of light field imaging, however, is somewhat overshadowed by the huge amount of storage capacity and bandwidth required to handle such extra information. These combined factors curtail the adoption of LF imaging for entertainment and industrial applications, and should be alleviated by using adequate compression techniques.

This work was supported by the Fundação para a Ciência e a Tecnologia (FCT), Portugal, under PhD Grant SFRH/BD/114894/2016, Programa Operacional Regional do Centro, project PlenoISLA POCI-01-0145-FEDER-028325 and by FCT/MCTES through national funds and when applicable co-funded by EU funds under the project UIDB/EEA/50008/2020.

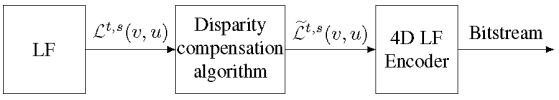

Fig. 1. Workflow of the proposed method.

Since common image/video encoders are not specifically tailored to exploit the data structure of 4D LF, several proposals arose recently for compression of LF images, both for lossless and lossy coding. Some methods adapt conventional image and video coding algorithms to exploit the 4D LF structure, such as [6], [7] based on HEVC and MRP, respectively. Others were designed specifically with the 4D LF structure in mind, such as WaSP [8] and MuLE [9], which were selected to be included in the verification model [10] of the JPEG Pleno Light Field standardisation activities. A comprehensive survey of dense light fields coding methods can be seen in [11].

This work proposes to improve the coding efficiency of 4D transform-based LF encoders, namely MuLE, by using a novel disparity compensation method to enhance the matching between the various sub-aperture images (SAI) of the 4D LF. Figure 1 presents the workflow of the proposed approach, where disparity compensation operates as a pre-processing of the 4D LF on the SAI domain ($\mathcal{L}^{t,s}(v,u)$), and the resulting image $\widetilde{\mathcal{L}}^{t,s}(v,u)$ is then encoded as it is, with no alterations to the encoder algorithm. The objective is to obtain a disparity-compensated LF ($\widetilde{\mathcal{L}}^{t,s}(v,u)$) where, ideally, there is no disparity between the resulting SAIs. As a consequence, the size of the original SAIs, in number of pixels, is increased due to algorithmic requirements to assure its reversibility. The new disparity compensated LF is then encoded using the 4D LF encoder MuLE. The experimental results show that, despite the increase on the data size due to a larger number of pixels to encode, the proposed approach is able to improve the compression efficiency of MuLE, in terms of Bjontegaard delta rate gains, for High Density Camera Array (HDCA) LFs.

This paper is organised as follows, Section II describes the proposed disparity compensation method, Section III presents an analysis on the experimental results, and the conclusions

are presented in Section IV.

II. Proposed LF disparity compensation

A LF is a 7D multi-dimensional vector function that samples the light flow in every point of space, called the plenoptic function [12]. To cope with the problems derived from its high-dimensionality, the plenoptic function can be simplified into a 4D function, under certain assumptions [12]:

$$\mathcal{L}(t,s,v,u) : \mathbb{R}^4 \longmapsto \mathbb{R}^3, \tag{1}$$

where (t,s) represents the angular and (v,u) the spatial coordinates. In this coordinate system a given SAI, *i.e.* a scene view or perspective captured from a certain angular position, can then be represented as:

$$\mathcal{L}^{t,s}(v,u) : \mathbb{R}^2 \longmapsto \mathbb{R}^3. \tag{2}$$

A 4D LF is usually acquired by an array of cameras or by a specialised camera which uses an array of micro-lenses positioned in front of the camera main sensor to capture different perspectives of the scene. Examples of the former are the images in the HCI dataset [13] and of the latter those in the EPFL dataset [14]. Due to these acquisition processes, each perspective is slightly different from its neighbours due to the baseline distance between the cameras or micro lenses.

Therefore, each SAI in a row or column after the disparity compensation can be represented by:

$$\widetilde{\mathcal{L}}^{t,s}(v,u) = \mathcal{L}^{t,s}(v - \mathcal{D}_{\mathcal{V}}(v,u), u - \mathcal{D}_{\mathcal{H}}(v,u)), \tag{3}$$

where $\mathcal{D}_{\mathcal{H}}$ represents the horizontal disparity between matching pixels in two neighbouring sub-aperture images, and $\mathcal{D}_{\mathcal{V}}$ the vertical disparity. For most of the JPEG Pleno datasets [15] it is reasonable to assume that the cameras (or micro lenses) are equally spaced, both horizontally and vertically. Under these conditions, the disparity of any two consecutive SAIs should be the same, such that $\mathcal{D}_{\mathcal{H}} = \mathcal{D}_{\mathcal{V}} = \mathcal{D}$.

With a proper compensation of this disparity between neighbouring SAIs, the compression efficiency for 4D encoders, such as MuLE, is expected to increase due to the higher data correlation resulting from better pixel alignment. Considering that a scene is composed of several objects at various depths, each SAI pixel can have a disparity value different from those of its neighbours, which should be compensated independently in order to obtain a better compression ratio. Since the determination of a disparity value for each pixel would entail a large increase in the amount of coded data, as well as a huge computational complexity, in this work the image is divided into 4D non-overlapping blocks and a block-level disparity $\mathcal{D}_b$ is estimated according to

$$\mathcal{D}_b = \underset{\mathcal{D}_b}{\arg\min}\, \mathrm{H}\left(\mathcal{Q}_p * \left\lfloor \frac{\mathrm{DCT}\left(\widetilde{\mathcal{L}}_b^{t,s}(v,u)\right)}{\mathcal{Q}_p} \right\rceil \right), \tag{4}$$

where $\mathrm{H}(\Omega) = -\sum_i P_i \log P_i$ represents the entropy, with $P_i = P(\Omega_i)$ the probability of the i^{th} symbol in the set Ω, $\lfloor \cdot \rceil$ is the rounding to the nearest integer operation, $\mathrm{DCT}(\cdot)$ denotes the coefficients of the 4D discrete cosine transform (DCT) of size equal to the block $\widetilde{\mathcal{L}}_b^{t,s}(v,u)$, and $\mathcal{Q}_p$ is the step uniform quantisation step, which was chosen to be 16.

The 4D disparity compensated blocks ($\widetilde{\mathcal{L}}_b^{t,s}(v,u)$) are computed by shifting each 2D SAI block (defined in (v,u)) by an amount proportional to the disparity and angular position t or s. After the optimal disparity compensation is found, the relative positions of all SAI blocks, originally co-located in the 4D space (*i.e.* with the same (v,u) and different (t,s)), are shifted accordingly, thus minimising Eq. (4).

To perform the minimisation, two parameters are needed: the size of the 4D blocks B_S and the maximum allowed disparity ($\mathcal{D}_M$), both expressed in pixels. Given B_S and $\mathcal{D}_M$ values as input, all disparity values belonging to $[-\mathcal{D}_M, \mathcal{D}_M]$ are tested for each block, in steps of 0.1 pixels. Assuming the horizontal (t) and vertical (s) disparities are the same ($\mathcal{D}_b$), due to the camera geometry, this search can be performed only along the s dimension. For each s the respective SAI block is shifted by an amount that depends on $\mathcal{D}_b$ and the position s itself, as given by,

$$\mathcal{D}_b^s = \lfloor \mathcal{D}_b \cdot (s-1) \rceil, \tag{5}$$

where, again, $\lfloor \cdot \rceil$ represents the rounding to the nearest integer operation. Eq. (5) shows that the further away a view is from the reference, *i.e.* $s = 1$, the further it needs to be moved. As sub pixel interpolations are not performed to keep the method fully reversible, the previously described displacement treats cases of $\mathcal{D} < 1$, by only moving a block $\mathcal{L}_b^{t,S_c}(v,u)$ when the accumulated disparity $D_b^{S_c}$ reaches the next integer. S_c represents the current SAI position. This process is shown in Figure 2a for the cases where D_b equals one.

The proposed disparity compensation adds new undetermined areas (hashed regions in Figure 2a) when each SAI block $\mathcal{L}_b^{t,s}(v,u)$ is shifted. Figure 2b shows the filling of the undetermined areas, using the neighbouring SAI information, *i.e.* the respective number of lines or columns is copied from the co-located position in the previous SAI, so that these areas are the same in all SAI, producing a disparity compensated image with increased 4D redundancy. For each new 4D block, Eq. (4) provides the optimal disparity value $\mathcal{D}_b$.

The optimal disparity for each $32 \times 32 \times 9 \times 9$ pixels 4D block in the *Greek* LF, from [15], is shown in Figure 3. It can easily be seen that regions at different depths are represented by different ranges of disparities.

From the algorithm description, it can be inferred that, the higher the value of $\mathcal{D}_M$, the higher the potential increase in the number of pixels in the final disparity compensated image. In order to minimise the number of extra pixels in the final image, blocks of equal disparity are grouped in super blocks, either in v or u direction, which is chosen to minimise the final number of pixels. The algorithm is applied with the calculated disparity to each super block as a whole. The super block partition is represented in red in Figure 3 (for the v direction). Such operation allows to reduce the amount of extra information needed to fill the undetermined areas presented in Figure 2.

It is important to balance the trade-off between the increased inter-SAI redundancy and the enlargement of the original im-

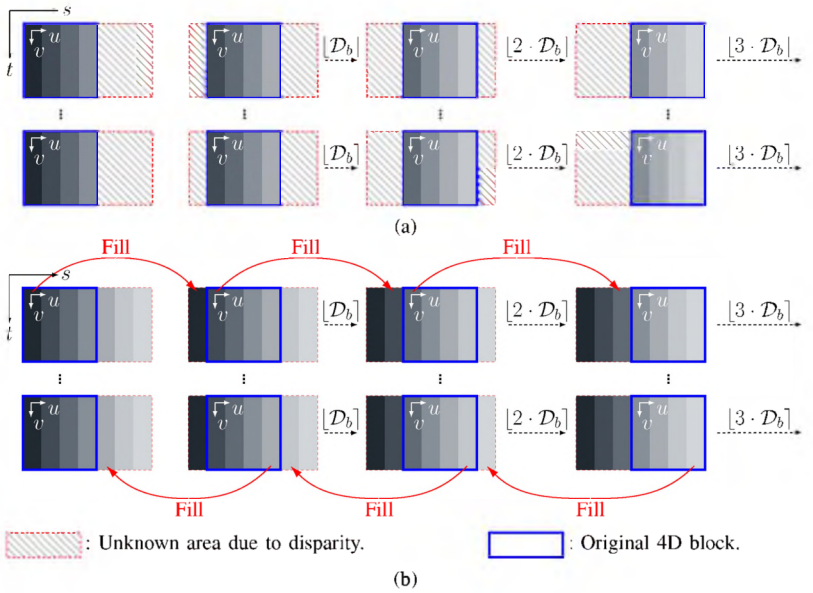

Fig. 2. Disparity compensation algorithm diagram: (a) new undetermined areas, (b) filling of the undetermined areas.

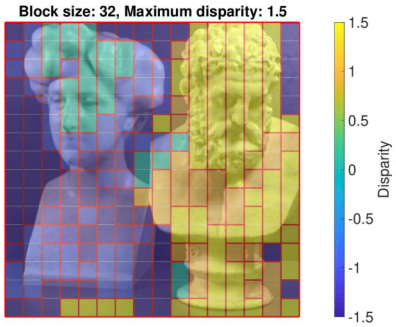

Fig. 3. Optimal disparity for each 32×32 pixels block of image Greek.

Fig. 4. Example of a prediction SAI computed with proposed algorithm.

age. This is done by guaranteeing that, although the codec has to encode more pixels than originally, the resulting compressed bit-stream is smaller due to the increased 4D redundancy.

Finally, after applying the disparity compensation to each super block, they are concatenated in order to assemble the final image. As each of these blocks has a different number of pixels, a padding operation, performed by repeating the last pixels, is performed to fill the gaps in the final disparity compensated image, as can be seen in Figure 4. For instance, in this example the algorithm increased the size of the SAI from 512×512 to 704×607 pixels.

III. Experimental Results

For these experiments the adopted procedure followed the coding common test conditions of JPEG Pleno Light Field (JPEG-CTC) [16]. Regarding the disparity compensation, for each of the images, multiple block sizes and maximum disparities were used, and the resulting images from the preprocessing step were encoded with MuLE without changes to the encoder. In order to determine the configuration that results in the best coding efficiency, the Bjontegaard metric was employed, taking as reference the HEVC anchor provided in [16].

The compression efficiency obtained by the proposed algorithm (MuLE with Disparity Compensation) in comparison with WaSP and MuLE, is presented in Table I. The last column of the table also shows, within parentheses, the best configuration for each image, in terms of block size and maximum disparity. For a better perception of the relative coding performance, the rate-distortion curves of *Greek* LF are shown in Figure 5, for the various encoders. All the results include the cost of transmitting the block size and the disparity for each block. This cost was calculated as: $\lceil \log_2(B) \rceil + \#D \times \left\lceil \log_2 \left(\frac{2 \times \max(|D|)}{0.1} \right) \right\rceil$, with B the block size, D the array of disparities to transmit, $\#D$ the number of elements in the array, and 0.1 the step used for the range of disparities.

As discussed in [17], MuLE achieves the highest efficiency for the lenslet type LFs, outperforming HEVC and WaSP. However, due to the higher disparity of the HDCA LF images, the 4D-DCT employed by MuLE is not very efficient, presenting a coding Bjontegaard delta (BD) rate up to 155.6% higher

TABLE I
BD-RATE (%) USING HEVC AS REFERENCE.

Type	LF	WaSP	MuLE	Proposed
Lenslet	I01	3.35	-40.42	-40.42 (128, 2.5)
	I02	-10.85	-38.79	-38.79 (128, 2.5)
	I04	-2.34	-32.93	-32.93 (64, 2.0)
	I09	13.09	-15.79	-15.79 (64, 2.0)
HDCA	Greek	-34.67	67.51	-9.88 (128, 2.5)
	Sideboard	-28.01	34.76	-13.15 (32, 1.0)
	Tarot	24.54	155.62	30.70 (128, 2.5)

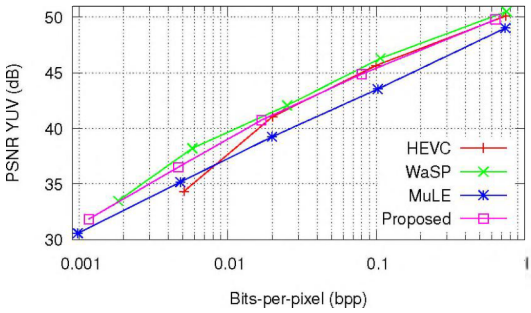

Fig. 5. Rate distortion curves for *Greek*.

than HEVC. For lenslet LF images, the proposed disparity compensation algorithm achieves the same results as those obtained by the MuLE algorithm. This was expected due to the small disparities of lenslet LF images and the high compression efficiency of MuLE in this case (the maximum disparity of lenslet LF images is in general less than one pixel).

Regarding the HDCA images, the results show that MuLE benefits from the use of the disparity compensation algorithm, for instance for *Greek* and *Sideboard* images the method outperforms HEVC. Although, *Tarot* BD rate is still higher than HEVC, the surplus was reduced from 155.5% to 30.7%.

While for lenslet images the proposed method produces no BD rate changes, for HDCA images the proposed method consistently improves the compression efficiency of MuLE, achieving Bjontegaard delta (BD) rate gains of 44.44%, on average, as can be seen in Table II.

TABLE II
BJONTEGAARD Δ RATE (%) USING MULE AS REFERENCE.

Type	LF	Proposed
HDCA	Greek	-43.12
	Sideboard	-35.85
	Tarot	-54.36
	Average	**-44.44**

IV. CONCLUSIONS AND FUTURE WORK

This work proposes a fully reversible pre-processing disparity compensation tool for light field image compression, which reduces the disparity between sub-aperture images with no changes to the codec operations. When using 4D light field codecs, images generated by the proposed algorithm present a higher compression ratio than the original images.

The experimental results show that the proposed method consistently improves the compression efficiency of MuLE for light fields images with higher disparity, like HDCA dataset, by about 44% on average, and does not affect the coding performance for lenslet images.

In the future, a study will be performed on the configuration parameters of the compensation algorithm with the intent of developing a method to automatically select the block size and the maximum disparity values.

REFERENCES

[1] I. J. 1/SC29/WG1N74014, "JPEG pleno call for proposals on light field coding," JPEG, 2017.

[2] H. G. Jeon, J. Park, G. Choe, J. Park, Y. Bok, Y. W. Tai, and I. S. Kweon, "Accurate depth map estimation from a lenslet light field camera," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2015, pp. 1547–1555.

[3] M. Harris, "Focusing on everything," *IEEE Spectr.*, vol. 49, no. 5, pp. 44–50, May 2012.

[4] D. Donatsch, S. A. Bigdeli, P. Robert, and M. Zwicker, "Hand-held 3D light field photography and applications," *The Visual Computer*, vol. 30, no. 6-8, p. 897, Jun. 2014.

[5] D. G. Dansereau, O. Pizarro, and S. B. Williams, "Linear volumetric focus for light field cameras," *ACM Transactions on Graphics (TOG)*, vol. 34, no. 2, pp. 1–20, Mar. 2015.

[6] Ricardo J. S. Monteiro, Nuno M. M. Rodrigues, Sérgio M. M. Faria, and Paulo J. L. Nunes, "Optimized reference picture selection for light field image coding," in *2019 27th European Signal Processing Conference (EUSIPCO)*, Sep. 2019, pp. 1–5.

[7] J. M. Santos, P. A. A. Assuncao, L. A. da S. Cruz, L. M. N. Tavora, R. Fonseca-Pinto, and S. M. M. Faria, "Lossless compression of light fields using multi-reference minimum rate predictors," in *2019 Data Compression Conference (DCC)*, March 2019, pp. 408–417.

[8] P. Astola and I. Tabus, "WaSP: Hierarchical warping, merging, and sparse prediction for light field image compression," in *European Workshop on Visual Information Processing (EUVIP)*. IEEE, Nov. 2018, pp. 1–6.

[9] M. B. de Carvalho, M. P. Pereira, G. Alves, E. A. B. da Silva, C. L. Pagliari, F. Pereira, and V. Testoni, "A 4D DCT-based lenslet light field codec," in *IEEE International Conference on Image Processing (ICIP)*. IEEE, Oct. 2018, pp. 435–439.

[10] P. Schelkens, P. Astola, E. A. B. da Silva, C. Pagliari, C. Perra, I. Tabus, and O. Watanabe, "JPEG Pleno light field coding technologies," in *Applications of Digital Image Processing XLII*, A. G. Tescher and T. Ebrahimi, Eds., vol. 11137, International Society for Optics and Photonics. SPIE, 2019, pp. 391 – 401.

[11] C. Conti, L. D. Soares, and P. Nunes, "Dense light field coding: A survey," *IEEE Access*, vol. 8, pp. 49 244–49 284, 2020.

[12] C. Zhang and T. Chen, *Light Field Sampling*. Morgan & Claypool, 2006.

[13] K. Honauer, O. Johannsen, D. Kondermann, and B. Goldluecke, "A Dataset and Evaluation Methodology for Depth Estimation on 4D Light Fields," in *Computer Vision – ACCV 2016*, 2017, pp. 19–34.

[14] M. Rerabek and T. Ebrahimi, "New Light Field Image Dataset," in *International Conference on Quality of Multimedia Experience (QoMEX)*, 2016.

[15] JPEG, "JPEG Pleno Light Field Datasets according to common test conditions," https://jpeg.org/plenodb/lf/pleno_lf/, accessed: 2020-02-06.

[16] WG1, "JPEG Pleno Light Field Coding Common Test Conditions V3.3," in *Doc. ISO/IEC JTC 1/SC29/WG1/N84025*, 84th JPEG Meeting, Brussels, Belgium, Jul. 2019.

[17] C. Perra, P. Astola, E. A. B. da Silva, H. Khanmohammad, C. Pagliari, P. Schelkens, and I. Tabus, "Performance analysis of JPEG Pleno light field coding," in *Applications of Digital Image Processing XLII*, A. G. Tescher and T. Ebrahimi, Eds., vol. 11137, International Society for Optics and Photonics. SPIE, 2019, pp. 402 – 413.