Journal of Visual Communication and Image Representation 54 (2018) 21–30

Contents lists available at ScienceDirect

Journal of Visual Communication and Image Representation

journal homepage: www.elsevier.com/locate/jvci

Lossless coding of light field images based on minimum-rate predictors☆

João M. Santos[a,b,*], Pedro A.A. Assuncao[a,c], Luis A. da Silva Cruz[a,b], Luis M.N. Tavora[c], Rui Fonseca-Pinto[a,c], Sergio M.M. Faria[a,c]

[a] Instituto de Telecomunicações, Portugal
[b] Department of Electrical and Computer Engineering, Faculty of Sciences and Technology, University of Coimbra, Coimbra 3030-290, Portugal
[c] Escola Superior de Tecnologia e Gestão (ESTG), Instituto Politécnico de Leiria, Leiria 2411-901, Portugal

ARTICLE INFO

Keywords:
Image coding
Lossless compression
Light field coding

ABSTRACT

Recent developments in light field acquisition and computational photography are driving new research efforts on light field encoding methods, capable of exploiting the specific features of this type of visual data. This paper presents a research study of lossless light field image compression, using Minimum-Rate Predictors (MRP) and mainstream image and video encoders. The research is focused on three light field representation formats: lenslet images, stack of sub-aperture images and epipolar images. The main contributions of this work are the 'Spiral-blackend' serialization method and the use of MRP for the lossless compression of light fields with joint encoding of RGB data. The results show that the lenslet format yields lower compression efficiencies than other formats. Furthermore, it is demonstrated that the MRP algorithm consistently outperforms HEVC-RExt, JPEG2000, JPEG-LS and CALIC when light fields are represented by either a stack of sub-aperture or epipolar images.

1. Introduction

Light field (LF) imaging, also known as plenoptic, holoscopic or integral imaging, a concept first introduced by Lippman in 1908 [1], has started to receive more attention once consumer-grade devices, such as those from Lytro (https://www.lytro.com/) and Raytrix (https://www.raytrix.de/) appeared in the market. These devices are capable of recording information about both the intensity and direction of the light rays reaching the camera, through the use of a specially designed optical system placed right in front of the image sensor. Other LF acquisition set-ups based on arrays of cameras, and therefore with different sizes, geometries and imaging characteristics, have also been developed mainly for research purposes, such as the camera array used to capture part of the LF data available at The (New) Stanford Light Field Archive [2].

LF cameras capture significantly more visual information from scenes than regular cameras, since besides light intensity from each point in the scene, they also record information related to the direction of the light rays travelling towards the sensor. One type of LF camera is built around an optical system that comprises a main lens and a micro-lenses array (MLA), placed in the optical path between the main lens and the light sensor [3], as illustrated in Fig. 1. In such arrangement, each micro-lens of the MLA captures the light intensity from each point in the scene, coming from different perspectives. Accordingly, as represented in Fig. 1, each sample acquired by the image sensor corresponds to the intensity of a light-ray coming from a point in the visual scene with a certain direction. This results in a 4D representation of the sampled LF in the form of a matrix of micro-images, each one corresponding to the light samples captured through a single micro-lens, where the spatial position of each sample implicitly encodes directional information. After adequate calibration and rectification, different views of the visual scene can be obtained from the 4D LF by extracting pixels located at the same spatial position in each micro-image. The whole set of views may comprise a stack of sub-aperture images, providing another representation format for the LF. A more detailed description and analysis of LF acquisition systems can be found in [4–6].

LF imaging is a primary enabler of computational photography, since the captured raw visual data can be processed *a posteriori*, in order to extract other image related data and render images with different characteristics. Amongst the operations that can be performed on the raw data, one can quote extraction of depth maps [7] and rendering of images with different focal planes, depth-of-field or viewing perspectives [3,8,9]. These technical possibilities make LF images useful in many fields of application, from medical imaging to the volumetric recording of prized sculptures and other cultural heritage artefacts [10]. However, this flexibility and the new functionalities come with a

☆ This paper has been recommended for acceptance by Zicheng Liu.

* Corresponding author at: Instituto de Telecomunicações, IPLeiria Campus 2, Morro do Lena - Alto do Vieiro, Leiria 2411-901, Portugal.
E-mail addresses: joao.santos@co.it.pt (J.M. Santos), amado@co.it.pt (P.A.A. Assuncao), luis.cruz@co.it.pt (L.A. da Silva Cruz), luis.tavora@ipleiria.pt (L.M.N. Tavora), rui.pinto@ipleiria.pt (R. Fonseca-Pinto), sergio.faria@co.it.pt (S.M.M. Faria).

https://doi.org/10.1016/j.jvcir.2018.03.003
Received 31 August 2017; Received in revised form 30 December 2017; Accepted 3 March 2018
Available online 22 April 2018
1047-3203/

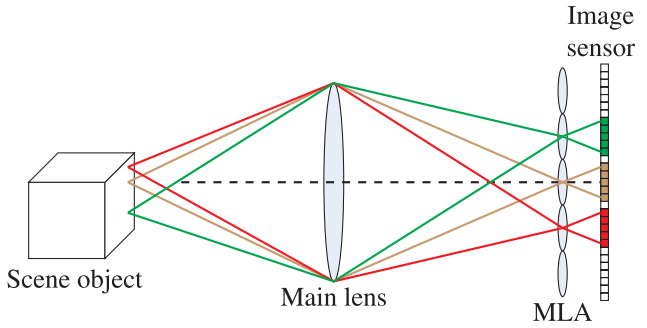


Fig. 1. Light field imaging acquisition system, example of an unfocused camera.

cost: the need to process and store the very large amount of data that are required for such representation of spatial and directional light colour intensity. Much like what is done with regular 2D images, data compression can be used to ease this problem.

In application fields with stringent accuracy requirements, high-precision depth measurements may be necessary and all relevant features captured in LF data must be preserved, which can only be achieved using lossless coding. This is the case, for instance, of some industrial computer vision applications and medical imaging (e.g., LF microscopy). The new and rather unique challenges posed by LF image coding are reflected in the recent standardization initiative of the ISO JPEG, named JPEG-Pleno [11], which aims at studying and proposing a new standard for compressed representation of LF imagery. This paper follows the same trend by presenting a contribution for performance evaluation of LF lossless coding.

The paper is organised as follows: Section 2 reviews the state-of-the-art methods for LF coding, Section 3 describes a lossless compression method based on minimum-rate predictors (MRP) and in Section 4 efficient LF image data arrangement and representation formats are described. In Section 5 the experimental set-up used in this work is presented, followed by the results obtained, as well their analysis and discussion. Finally, Section 6 presents the overall conclusions and suggestions for future work.

2. Related work

Several articles have been published reporting on methods for LF image coding, with most of them focused on lossy coding techniques. Some proposals exploit the inherent correlation between the neighbouring micro-images which, due to the partially overlapping field of view, share a significant amount of information. For instance, in [12], a three-dimensional discrete cosine transform (3D-DCT) is applied to a stack of micro-images, to exploit the correlation between adjacent micro-images. Others, exploit the correlation between sub-aperture images, using a pseudo-video approach where such images are encoded as frames of a video sequence using standard video coding methods. In [13], the lenslet image is decomposed into various sub-aperture images, which are assembled into a cube in order to apply a three-dimensional wavelet or DCT transform, followed by quantisation and entropy encoding. In [14], lenslet images are compressed by extending H.264/AVC spatial prediction to take advantage of the similarity between the micro-images, by estimating and compensating the micro-images disparities. In the case of holoscopic video, this compensation is combined with motion-compensated temporal prediction, as followed in [15] by using HEVC as the core information compressor. In [16], a scheme based on Multiview Video Coding is proposed. However, these solutions do not exploit depth information, leaving room for more efficient prediction schemes that take advantage of multidimensional coding, as was done recently by the authors in [17,18].

To the best of the authors' knowledge, so far only two methods for lossless LF image coding have been investigated and published, namely one proposed by Perra in [19] and another proposed by Helgin in [20]. In [19], Perra proposes a lossless coding architecture for plenoptic images that processes the raw information provided by the image sensor, rather than using rectified images. Initially the image is separated into colour components, and then the resulting colour planes are independently processed by an entropy analysis module, which calculates an optimal displacement that achieves the best matching between a pixel and its replicas. This information is then used in a block DPCM encoder, that partitions each colour component into blocks which are predicted using the previous horizontal blocks as reference. Finally, the resulting information is encoded with a Lempel-Ziv-Markov chain algorithm (LZMA). Compared to JPEG2000 (compressing lenslet non-rectified) and using a particular set of images (non-available), this method reduces the bitrate by 56% and requires about 32.5% of the uncompressed bitrate.

In [20], Helin et al. encode rectified lenslet images in a (multi) view format, by taking each sub-aperture image as a different view, and exploiting the inter-view redundancy. An optimal sparse predictor was designed, using samples from the current view and from up to five reference views, scanned along a spiral path, similarly to what is done in [21]. The views are segmented into several regions of constant depth, and each region is encoded using a different predictor, specifically designed for that region. The region's boundaries and locations are propagated for each additional view after disparity compensation using optimal disparities, ensuring that the same regions exist in the different views. The central view is encoded independently using lossless JPEG2000. This method achieves an average bit rate of 91.1% in comparison with HEVC.

The Minimum-Rate Prediction (MRP) algorithm [22] has proven to be one of the most efficient state-of-the-art algorithms for lossless coding. Due to its compression properties, this work further explores its application to LF image compression, combined with the study of the LF data representation format that most benefits the MRP algorithm. Since the algorithm is natively single component, i.e. it only operates on single-colour components or grey-scale images, a solution to jointly encode the RGB components of LF images was also devised.

3. Lossless coding using Minimum Rate Predictors

Minimum rate prediction coders, like other lossless coding algorithms, try to minimise the coding rate of the prediction error [22], rather than minimising the mean square prediction error for a given rate constraint, as occurs in lossy encoders. The approach used by the MRP algorithm to achieve reduced coding ratios is to estimate the amount of information conveyed by prediction errors, and then design the predictors with the goal of minimising the rate. Adaptive prediction is used to cope with the diversity of visual content represented throughout the images. Efficient entropy coding of the prediction error involves context modelling, classification into a finite number of groups and assigning coders (or coding tables) that correspond to the probability models. In each of these groups the variance of the prediction error should be roughly constant, and the prediction errors are then modelled with a generalised Gaussian distribution [23] within the given group variance.

To minimise the encoding rate of prediction errors, the design of MRP is mathematically formulated as the search for linear predictors, seeking to minimise the average number of bits required to encode the prediction residues and auxiliary information, such as the prediction coefficients. The amount of information conveyed by the prediction errors e in a given image region R is given by [22],

$$I(R) = \sum_{n=0}^{N-1} \left\{ - \sum_{p_0 \in g_n} \log_2 \alpha_n + \frac{\log_2 \epsilon}{2} \cdot \sum_{p_0 \in g_n} \frac{e^2}{\sigma_n^2} \right\}, \quad (1)$$

where $N$is the number of groups associated with the context modelling, g_n are the pixels associated with the n^{th} group, σ_n is the variance of the

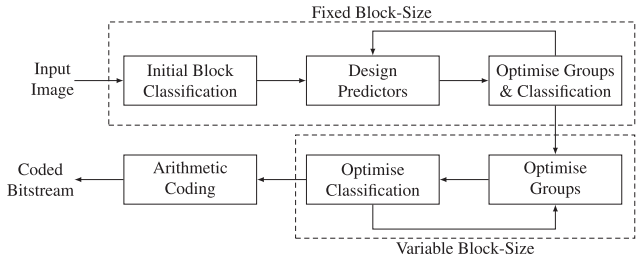


Fig. 2. Functional diagram of the MRP algorithm.

prediction error e in the n^{th} group, and $\alpha_n = \Delta e/\sqrt{2\pi\sigma_n^2}$, with Δe representing a sufficiently small quantization step-size of e. Since this amount of information provides an indication about the length of the lossless binary representation of pixels in region R, the objective of the encoding procedure is to minimise $I(R)$, such that low values of e and $I(R)$ are obtained through a proper choice of linear prediction coefficients for the groups. The linear prediction models, used by MRP are adapted to the image characteristics, through the use of a variable block size partitioning scheme combined with the iterative minimization of a cost function.

3.1. The MRP algorithm

In general, the MRP algorithm can be described by the processing modules shown in Fig. 2, including the three main stages named *Fixed Block-Size*, *Variable Block-Size* and *Arithmetic Coding*. In the *Fixed Block-Size* stage, the predictors are designed using blocks of 8×8 pixels. In the *Variable Block-Size* stage, with the already designed predictors, the blocks classification is optimised using blocks with variable size. Finally, all the resulting information is encoded using an arithmetic coder [22]. The stages named as *Fixed Block-Size* and *Variable Block-Size* include, each one, an optimisation loop. The first loop comprises the modules *Design Predictors* and *Optimise Groups* & *Classification*, while the second loop comprises the modules *Optimise Groups* and *Optimise Classification*.

Initially, before determining the prediction, the algorithm starts by dividing the image into blocks of 8×8 samples (e.g. luminance values), in the *Initial Block Classification* of Fig. 2. Each of these blocks is assigned to one of M classes, according to the block variance. All the blocks in a given class are encoded using a linear predictor designed for that class.

The first optimisation loop starts in module *Design Predictors*, where a set of parameters, defining the prediction models, is determined for each class by solving a set of Yule-Walker equations involving the underlying correlations. These parameters are computed on all pixels of the blocks associated with the class. In this work, two further linear combination terms were used in addition to those already included in the original MRP algorithm. These terms provide new prediction contributions from two additional sets of pixels with the objective of better exploiting the correlation between adjacent light field views. A similar approach was used in our previous work on coding of medical images [24].

These two sets of pixels belong to two distinct images, akin to past and future frames in video, as represented in Fig. 3, allowing for bi-directional inter-frame prediction. The resulting prediction function for pixel p_0 of the current frame, belonging to a block from the m-th class, is given by,

$$\hat{s}(p_0) = \sum_{k=1}^{K_c} a_{m_c}(k)\cdot s_c(p_k) + \sum_{k=1}^{K_b} a_{m_b}(k)\cdot s_b(p_k) + \sum_{k=1}^{K_f} a_{m_f}(k)\cdot s_f(p_k), \quad (2)$$

where $a_{m_c}(k)$, $a_{m_b}(k)$ and $a_{m_f}(k)$ represent the prediction model weights associated to pixels in the current and reference frames (forward and backward), respectively, for the m-th class. In Eq. (2), $s_c(p_k)$, $s_b(p_k)$ and $s_f(p_k)$ are the pixel values in the current and reference frames at position $k$of the prediction support region (see Fig. 3), and K_c, K_b and K_f are the sizes of the prediction support regions. In this work, the prediction support templates were defined to be centred at the position of the pixel to be predicted, indicated by the bold circles in Fig. 3. As described in Section 4, light fields can be structured as pseudo-video sequences of slightly shifted views (a stack of sub-aperture images), which is a particularly efficient representation for such prediction supports.

After defining the prediction models, the groups thresholds are optimised in modules *Optimise Groups* & *Classification*, using a quantised estimation of the residuals variance, and a reclassification of the blocks based on the following cost function, expressed in number of bits:

$$J = \sum_{p_0} L(e|\hat{s}(p_0),n) + B_a + B_m + B_t, \quad (3)$$

where $L(e|\hat{s}(p_0),n)$ is the cost associated with the prediction errors given by the probability density function model pertaining to the n-th group, and B_a, B_m and B_t are the costs resulting from binary encoding the prediction coefficients, class selection and context modelling threshold values, respectively. In the optimisation of the groups context modelling, the prediction error is used to find a probability model for each group, where the signal may be considered stationary with constant variance and with a generalised Gaussian probability density function. The context for a pixel is determined by finding an estimate of the variance of the prediction residuals at that pixel, given by,

$$U = \sum_{k=1}^{K_c} \frac{1}{\delta_{c,k}} |s_c(p_k) - \hat{s}_c(p_k)| + \sum_{k=1}^{K_b} \frac{1}{\delta_{b,k}} |s_b(p_k) - \hat{s}_b(p_k)| + \sum_{k=1}^{K_f} \frac{1}{\delta_{f,k}} |s_f(p_k) - \hat{s}_f(p_k)|. \quad (4)$$

The weighting factors $\delta_{j,k}$, $j = c,b,f$, are proportional to the Euclidean distance between the current pixel, p_0, and the reference ones, p_k, as given by Eq. (5), where D represents the inter-frame distance (note that $D = 0$ for $j = c$). The parameter U is then quantised using threshold values and each pixel is classified into one of the groups corresponding to a constant variance.

$$\delta_{j,k} = \frac{\sqrt{d_x(k)^2 + d_y(k)^2 + D^2}}{64}, \quad j = c,b,f \quad (5)$$

This first loop stops either after 10 iterations without cost reduction or after reaching a predetermined number of iterations (e.g., 100). After determining the initial design of the predictors, the algorithm enters the second optimisation loop, where variable block size (VBS) prediction is used. At this stage the image can be partitioned into 32×32 pixel blocks, which in turn can be sub-partitioned according to a quadtree structure, down to 2×2 pixel blocks. In this second loop, the stopping criterion is either one iteration without cost reduction or a predetermined number of iterations (e.g., 100).

The quantisation thresholds, which define the groups for probability distribution estimation and block classification, are iteratively optimised at this stage. In this context, block optimisation refers to the class selection and block partitioning that maximises the overall compression ratio. For optimising the block classification, the blocks at each quadtree level are classified into the existing classes, according to their cost value. The optimal quadtree partition is selected by using, at each level, *i.e.* block size, the current block cost and the sum of the costs of the resulting sub-blocks. The partitioning only occurs if the sum of the costs of the four resulting sub-blocks is lower than the cost of the current block. The final arithmetic encoding of all the parameters and prediction residues is done with a range coder [25].

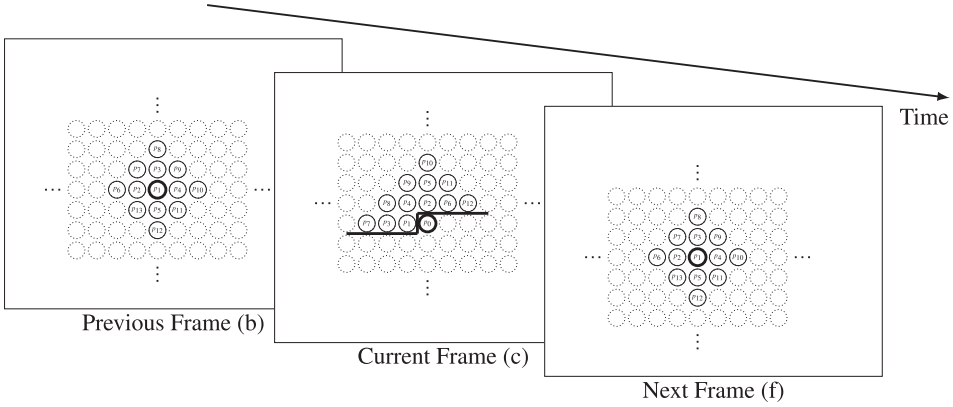


Fig. 3. Support templates used for intra and bidirectional prediction.

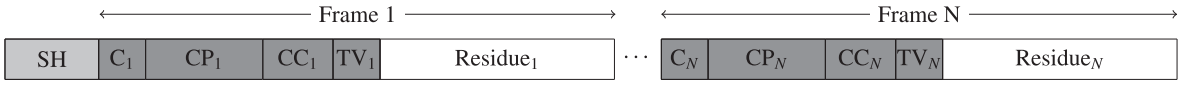


Fig. 4. MRP stream frame-based structure.

3.2. Coded stream syntax and overhead

The syntax of MRP coded streams includes a sequence header, followed by the information at frame level, which consists in the number of classes, quadtree division and classification, predictors coefficients and prediction error. This is organised as shown in Fig. 4, representing the sequence of the syntax elements: Sequence Header (SH), Number of Classes (C), Class Partition (CP), Class Coefficients (CC) and Threshold Value (TV). The shaded blocks, in light grey and grey, represent the side information at sequence and frame level, respectively.

Table 1 describes the side information for the different data arrangements described in Section 4. The last three columns show the number of bits spent in each syntax element shown in Fig. 4, accounting for the three colour components of the *Vespa* LF image.

As it can be observed in Table 1, the side information represents a small percentage of the total file size, ranging from 3.05% for Pseudo-Video Sequence (PVS) to 9.61% for lenslet. The lenslet format incurs in the highest overhead, which mostly results from its data structure with a more detailed quadtree and smaller blocks. The amount of side information required for the PVS and epipolar formats is higher for C, CC and TV due to the larger number of frames.

4. Data formats for LF coding

In general, there are several factors with impact on the performance of a video or image encoder, like input content, quality and representation, coding architecture and algorithm, and system parameters. As some of these factors are constrained by the available data content (LF) and coding algorithms, the structure arrangement of the input images might be a key factor to boost the compression performance. Thus, in this work LF data is arranged in various formats prior to be compressed by a video encoder, namely lenslet, pseudo-video sequence (stack of sub-aperture images) and epipolar arrangements.

Lenslet is the primary format to represent a 4D LF. This type of LF image is composed of a two-dimensional array of micro-images. As it can be seen in the rectified lenslet image shown in Fig. 5, the micro-lens pattern is clearly discernible.

The size of each micro-image in these images is 15×15 pixels, where the black pixels in the corner of each micro-image correspond to values computed with low reliability.[1] Lenslet images are compressed using intra-frame algorithms, which exploit the spatial redundancy within a 2D set of data.

Nevertheless, as mentioned before, in the lenslet image the spatial position of each pixel in a micro-image represents the direction of a light ray. Thus, an alternative data format reinterprets the multiple views (i.e., sub-aperture images) derived from the LF micro-images as a short video sequence, which is called Pseudo-Video Sequence (PSV), where the sub-aperture images assume the role of video frames. In spite of the existing redundancy between adjacent LF micro-images, higher compression performance is expected when pseudo-video is used instead of lenslet images. Furthermore, the black pixels in the corner of each micro-image generate black sub-aperture images, which are referred to as Low Reliability Black Frames (LRBF).

The compression performance of pseudo-video sequences for standard lossy video encoders has been addressed in [21], using the state-of-the-art HEVC video encoder with two different arrangement schemes for the stack of sub-aperture images, namely 'Raster' and 'Spiral' scan. However, to the authors' best knowledge, the impact of such different data formats for lossless encoding has never been investigated. Additionally, due to its inherent characteristics, the LRBF images may be properly exploited to improve the compression efficiency. Thus, this work investigates the performance of lossless coding using various schemes to generate pseudo-video sequences extracted from the stack of sub-aperture images, namely a new scheme named as 'Spiral-blackend'.

As can be seen in Fig. 6(a) and (b), 'Raster' and 'Spiral' scans position LRBF images between other sub-aperture images with significant information, which may be prejudicial for the inter-frame prediction. In the proposed 'Spiral-blackend' scan (SBE), the pseudo-video sequence is built on spiral order, but packing together all LRBF images at the end of the sequence, rather than including them in the middle of the sequence. In Fig. 6(c), the scan starts at the central sub-aperture image proceeding in the green[2] spiral, in order to minimise the disparity between consecutive scanned views, thus displacing the corner frames to the end of pseudo-video sequence. The green path is followed by the yellow path (outer path) and then by the red path (external path), in a clock-wise fashion.

The Epipolar representation uses Epipolar Plane Images (EPI), which intersects the image plane of each view where it forms lines. The depth of the corresponding scene is closely related to the slope of these lines [27]. For instance, a vertical line indicates that an object is located

[1] These black pixels may have different brightness. As they may be used in LF reconstruction, they are also encoded in a lossless manner.

[2] For interpretation of color in Fig. 6, the reader is referred to the web version of this article.

Table 1
Side information for different data arrangements (*Vespa* image, results in bits).

Type of Information	Description	Size (bits)	Frequency	Lenslet (R + G + B)	PVS (R + G + B)	EPI (R + G + B)
Sequence Header	Magic number, software version, width, height and number of frames, pixel bit depth, number of B type frames, bi-directional prediction type flag, number of components, number of groups, prediction order values and histogram packing information.	200	Once	600	600	600
Number of classes (C)	Number of classes used for a given frame.	8	Per frame	24	5400	10,416
Class partition (CP)	Quadtree partition and block classification information.	Variable	Per frame	56,259,936	7,617,400	13,953,448
Class coefficients (CC)	Coefficients values for each class	Variable	Per frame	68,784	3,612,248	5,998,792
Threshold values (TV)	Encoding of the probability distribution thresholds values.	Variable	Per frame	14,328	1,253,616	1,977,072
Total Side Information				56,343,672 (9.61%)	12,489,264 (3.05%)	21,940,328 (4.94%)
Prediction residuals	Encoding of the prediction residuals.	Variable	Per frame	529,828,920 (90.39%)	397,219,056 (96.95%)	422,222,840 (95.06%)

Fig. 5. Example of a rectified lenslet image and a zoomed detail taken from *Friends_1* image [26].

at an infinite distance from the camera, and therefore such object has no disparity between the sub-aperture images.

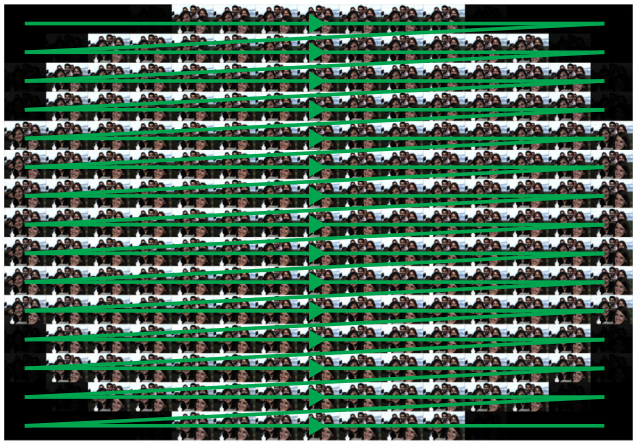
An example of generating an EPI slice from the pseudo-video sequence *Friends_1* is shown in Fig. 7. The EPI image illustrated in Fig. 7(b) corresponds to the plane $(x,\ s)$, represented by the white lines in the planes $(x,\ y)$ and $(y,\ s)$. This type of image presents very specific characteristics, as it comprises smooth areas and sharp edges. Therefore, it is expected that methods based on linear predictors, like MRP, perform efficiently on predicting the sharp edges, thus improving the compression.

5. Experimental performance analysis and comparisons

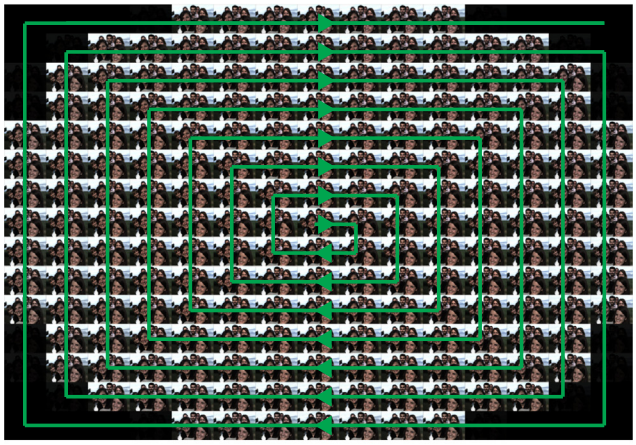
In this section the LF image coding efficiency of the MRP algorithm is evaluated and compared to several competing lossless image coding algorithms. The evaluations were carried out by encoding an ensemble of images that were recently used in a LF image coding competition [26], thus allowing direct comparison with results published in other research works. As mentioned in the previous section, since LF imagery data can be represented in several different formats, namely as lenslet, sub-aperture and epipolar, each one impacting the compression performance in a different manner, the various test cases include combinations of coding techniques and LF data formats. The results are then analysed and discussed, providing evidence about consistent behaviour of LF lossless encoding using specific LF data arrangements.

5.1. Lossless coding algorithms

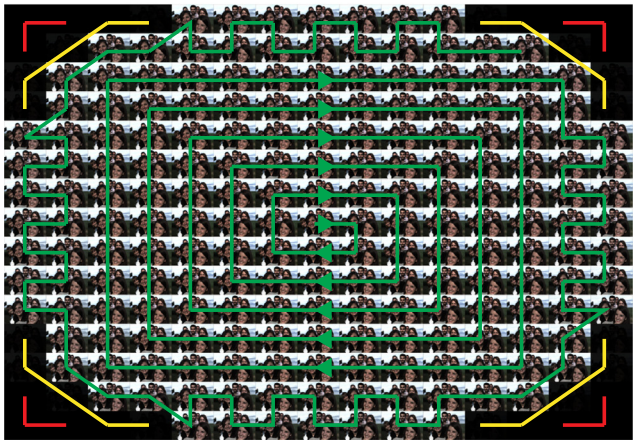
In order to obtain comparable results, these experiments used not only the same image dataset and formats, but also perform similar compression evaluation tests for various state-of-the-art high performance lossless encoders, namely JPEG-LS [28], JPEG2000 [29], CALIC [30] and the High Efficiency Video Coder (HEVC) [31]. In order to establish a common ground to enable fair comparisons between the results obtained in this research study and other works, the test conditions are explicitly defined below. The encoding configurations were



(a)



(b)



(c)

Fig. 6. Representation of the used scans: (a): 'Raster', (b): 'Spiral' and (c) 'Spiral-blackend'.

defined as follows[3]:

- JPEG-LS [32]: JPEG-LS reference software with standard configuration (*-ls 1*) and YCbCr colour transformation bypass active (*-c*).

[3] Only the main parametrisations are presented, everything else was left with the default values.

(a)

(b)

Fig. 7. Example of a subset of EPIs extracted from the pseudo-video sequence *Friends_1* using the 'Raster' scan [26].

- JPEG2000 [33]: OpenJPEG implementation, lossless, using default 6 resolutions, 64 × 64 code-blocks, reversible DWT 5-3, with multi-component transform disabled (*-mct 0*).
- CALIC [34]: Martin Briano's implementation, in lossless mode (*-d = 0*) and 2D CALIC (*-m 1*).
- HEVC [35]: HM reference software, v.16.12, with the Format Range Extension (REext) [31] and main REext profile (*Profile = main-REext*). Additional configurations/modes: lossless cost mode, QP zero, transform quantization bypass (*TransquantBypassEnable = 1*), 32 intra period (*IntraPeriod = 32*), 16 GOP size (*GOPSize = 16*), CRA intra random access point (open GOP) (*DecodingRefreshType = 1*) and TZ Search motion estimation (*FastSearch = 1*).
- MRP: maximum number of classes dependent on image size, 63 for lenslet, 42 for PVS and 29 for EPI. Bidirectional prediction with GOPsize = 8 (*-B -G 7*), histogram packing enabled, variable block size prediction and number of reference pixels according to image format:
  - Lenslet: 72 pixels.
  - Others: 56 or 30 pixels, depending of the data arrangement (PVS and EPI respectively) on the intra frame, 20 pixels on the current frame and 5 pixels on the reference frame for unidirectional prediction frames and 12 pixels on the current frame and 13 pixels on the reference frames for bidirectional prediction frames.

All the results report the encoding performance measured in bits-per-pixel (bpp), calculated according to:

$$bpp = \frac{B_{C_R} + B_{C_G} + B_{C_B}}{W \times H}, \tag{6}$$

where $B_{C_i}, i = R,G,B$, is the size (#bits) of the compressed image component C_i and $W$and $H$are the image width and height, measured in pixels.

5.2. Test image data

The coding experiments were performed on the LF Image Dataset used in the 2016 IEEE International Conference on Multimedia and Expo (ICME 2016) Grand Challenge: Light-Field Image Compression [26]. This dataset contains images acquired with a LF Lytro Illum camera and stored in the proprietary Lytro format called LFR. Before compression, these images were pre-processed using Dansereau's LF image processing toolbox version 0.4 (LF Toolbox v0.4) [36]. The pre-processing stage includes a LF image rectification step, relying on the camera calibration data, as described in [37]. The resulting data were then converted to images in lenslet format with a spatial resolution of 9375×6510 pixels, and the corresponding sub-aperture images, with a spatial resolution of 625×434 pixels. These images were then scaled to a 0–255 range fitting an 8-bit representation. The RGB format was used as input for encoding all LF images.

Since MRP, as previously mentioned, is a single-component encoder, without support for RGB images, two schemes were used to encode the three components of RGB LF images: (i) encoding each RGB component individually and accounting for all of them as whole or (ii) concatenating the RGB components prior to encoding into a single bitstream. Moreover, the MRP encoder requires the input image dimensions to be multiple of 8 pixels. Therefore, whenever necessary the images were extended by replicating the last column or row in order to match this requirement.

5.3. Experimental results and comparisons

To simplify the compression efficiency analysis for the different lossless encoders, the results are presented according to the LF image representations used, namely lenslet, sub-aperture and epipolar. While the lenslet representation can only be encoded as a 2D image, sub-aperture and EPI images are stacked to form a pseudo-video signal, allowing to exploit the correlation along the third dimension inherent to a sequence of sub-images. As it will become clear from the results, some encoders have better compression performance when using specific LF image formats. As an example, the results showed that encoders like HEVC and a modified version of MRP [24], which are able to exploit correlation in a third dimension (akin to the temporal dimension in video), are more adequate to compress the pseudo-video image stacks. Note that in these tests, independent MRP encoding was used for each colour component and the sum of the coded bits from the three components was accounted for measuring the efficiency.

5.3.1. Coding of lenslet LF images

After rectification and conversion to the lenslet format as described above, all images in the dataset from [26] were compressed using the lossless encoders listed previously. Table 2 lists the compression ratios achieved on each image, as well as the average rate per encoder, all expressed in bits-per-pixel. In the following tables the best result for each row is highlighted in bold.

Table 2
Coding performance for lenslet format (bpp).

Sequences	JPEG2000	JPEG-LS	CALIC	HEVC	MRP
Ankylosaurus_Diplodocus_1	14.24	**10.76**	12.45	11.18	11.26
Bikes	13.12	10.46	11.71	**9.91**	10.71
Color_Chart_1	13.51	**10.75**	12.24	10.89	11.09
Danger_de_Mort	11.99	9.52	10.69	**9.30**	9.75
Desktop	13.32	8.35	10.68	**8.28**	9.13
Flowers	12.34	9.80	11.03	**9.41**	10.08
Fountain_Vincent_2	13.91	11.26	12.59	**10.79**	11.58
Friends_1	12.62	**9.10**	11.09	9.44	9.57
ISO_Chart_12	14.02	10.88	12.34	**10.32**	11.22
Magnets_1	14.20	**10.65**	12.22	11.09	11.05
Stone_Pillars_Outside	12.67	10.07	11.35	**9.91**	10.22
Vespa	11.74	9.39	10.55	**9.15**	9.60
Average	13.14	10.08	11.58	**9.96**	10.44

It can be observed that for lenslet images HEVC and JPEG-LS achieve the highest average coding performance among the tested algorithms, far better than the remaining ones. HEVC performs slightly better (0.12 bpp lower) than JPEG-LS.

5.3.2. Coding of sub-aperture stacked images (pseudo-video)

In a second set of experiments the LF data, represented by sub-aperture images stacked to form pseudo-video sequences, were compressed using the same encoders, as shown in Table 3. Prior work [21] showed that the sequence order that was followed to select the sub-aperture images for stacking or serializing the data into pseudo-video is an important factor that affects the resulting compression. To characterise this impact, three scanning orders described earlier were used for encoding with HEVC and MRP, and the results segmented accordingly.

As expected, encoders using inter-frame prediction tools, which can explore redundancy between consecutive sub-images (*i.e.* views), exhibit the best average performance: MRP achieves the best efficiency, followed by HEVC (0.34 bpp higher than MRP), both outperforming intra-frame encoders. From the results it is also clear that MRP and HEVC show better performance when encoding the LF images represented as pseudo-video as compared to the lenslet representation, by 3.30 and 2.48 bpp, respectively. This gain is notably smaller for the intra-frame based encoders. In this latter case, using pseudo-video rather than lenslet format, reduces the coding rate by the same order of magnitude, with the exception of JPEG-LS, where the reduction is only a few hundredths of bpp.

Possibly, the gains shown by HEVC and MRP described in Section 3 can be associated to the efficiency of the encoding tools implemented by these two encoders, which help to exploit inter-sub-aperture image correlation. The first of these tools is bi-directional prediction which, in the case at hand, encoding neighbouring sub-aperture images, most likely succeeds in overcoming the problem of predicting pixels of one sub-aperture image which are occluded in neighbouring images taken from slightly different perspectives. The second feature of both encoders which also helps improving the compression ratio is sub-pixel accurate prediction. Since it is possible that for a given baseline distance the disparity observed in adjacent sub-aperture images have fractional values, depending on the depth of the scene objects, being able to form inter-sub-aperture image predictions with fractional-pixel accuracy, is likely to improve the prediction quality, driving the coding rate down. It is therefore expectable that the sub-pixel accurate motion estimation/compensation of HEVC contributes to improve the encoding performance of LF pseudo-video sequences. The MRP algorithm, on the other hand, does not use motion compensation, or apparently sub-pixel accurate prediction, as predictors involve pixels located at integer-grid positions. It can however achieve sub-pixel accurate prediction, without motion compensation, because the linear predictors can use fractional coefficients, thus defining prediction values which are interpolations between pixels of the prediction support region. As an example, consider a case where only two reference neighbouring pixels of prediction support region are used; if the linear prediction coefficients associated with these pixels are 0.5, for instance, then the predictor is essentially computing an interpolation of these two pixels, *i.e.* an half-pel position reference value. Linear prediction in MRP can thus produce a result similar to sub-pixel accuracy, with the ensuing advantages for the coding of the sequence of sub-aperture images.

Another variable with impact on the compression ratio, and important to analyse, is the type of scan order used to create the pseudo-video sequence from the stack of sub-aperture images. As can be seen in Table 3, the scanning order used has only a small effect on the coding performance. In the HEVC case, the 'Spiral-blackend' method provides the most efficient results, with a difference of up to 0.1 bpp with respect to the other scan orders. For MRP, the best results are obtained with the

Table 3
Coding performance for pseudo-video LF format (bpp).

Sequences	JPEG2000	JPEG-LS	CALIC	HEVC			MRP			Ref. [20]
				Raster	Spiral	SBE	Raster	Spiral	SBE	
Ankylosaurus_Diplodocus_1	9.79	9.60	9.33	7.94	7.86	7.85	**7.51**	**7.51**	7.52	8.10
Bikes	11.59	11.02	9.45	7.83	7.75	7.72	7.45	**7.41**	7.42	9.57
Color_Chart_1	9.62	9.29	10.44	8.06	7.94	7.94	7.30	**7.28**	7.30	9.79
Danger_de_Mort	11.49	10.94	9.64	7.36	7.25	7.23	7.07	**7.01**	7.02	10.59
Desktop	8.65	7.86	10.83	6.01	5.97	5.96	**5.68**	5.69	5.69	6.98
Flowers	12.12	11.49	10.92	7.46	7.43	7.41	**7.17**	7.20	7.21	10.74
Fountain_Vincent_2	11.38	10.95	8.18	8.39	8.32	8.30	**7.81**	**7.81**	7.83	9.64
Friends_1	10.41	10.00	11.47	6.88	6.80	6.78	6.45	6.42	**6.41**	8.27
ISO_Chart_12	11.05	10.45	9.59	8.23	8.11	8.09	7.73	**7.71**	**7.71**	9.81
Magnets_1	9.76	9.52	10.06	7.93	7.84	7.84	**7.53**	**7.53**	7.55	8.18
Stone_Pillars_Outside	12.01	11.56	11.36	7.61	7.58	7.54	**7.40**	7.44	7.41	9.24
Vespa	9.80	9.25	10.88	7.21	7.13	7.12	6.71	**6.68**	6.71	9.88
Average	10.64	10.16	10.18	7.58	7.50	7.48	7.15	**7.14**	7.15	9.23

'Spiral' scan, but the encoding performance differences observed when using different scans are minimal 0.01 bpp at most. The 'Raster' scan order produces the worst results for both encoders, with a more pronounced negative effect in HEVC. Most likely, this worse performance is due to the fact that, unlike the other scan orders, this arrangement does not keep neighbouring sub-aperture images close to each other in the serialized sequence, breaking the high inter-sub-aperture image correlation structure that exists in the original 2D tiling. Additionally, unlike what happens in the 'Spiral-blackend' arrangement, where the similar LRBF images are packed together at the end of the pseudo-video sequence, slightly improving the performance of HEVC, in the case of the 'Raster' scan order the LRBF images are distributed along the pseudo-video sequence, further deteriorating the inter-frame prediction performance. It is also noticeable that for MRP the 'Spiral' and 'Spiral-blackend' orders do not improve the coding results obtained with the 'Raster' order very much. Perhaps this is due to the very good adaptation mechanisms used in the MRP prediction step, which make the method very robust.

One alternative way of handling the LRBF images is to simply not encode them. This is possible because usually the structure of the microlenses images is known, and so the low reliability pixels can be identified and removed. This solution was also tested, where the pseudo-video sequences obtained using the Spiral scan order and truncated to ignore the LRBF images were encoded using HEVC and MRP, achieving on average coding ratios of 7.30 bpp and 6.99 bpp, respectively, which is an improvement of 0.18 bpp for HEVC and 0.16 bpp for MRP. However, not coding these images results in a compression method which is no longer lossless, inducing slight deteriorations to the performance of post-processing operations like refocusing that use these low reliability corner pixels, resulting however in imperceptible degradations on the refocused images.

The rightmost column of Table 3 presents results of a recent light field image encoding method [20]. Despite the fact that such algorithm was specifically designed for LF images, the results show that the compression efficiency of both HEVC and MRP is consistently higher for all data arrangements (i.e. lower bpp). On average the method presented in [20] requires about 29% more bpp than MRP and 23% more than HEVC.

5.3.3. Coding of epipolar images

The coding of LF images represented in the third format mentioned before, epipolar format, and using the encoders indicated in the previous sections was also evaluated. The experimental results obtained using the same encoders as in the previous sections, and the same dataset, are shown in Table 4. These results show that MRP is again the most efficient encoder for this type of images, followed by the HEVC (0.54 bpp higher). However, the MRP and HEVC results obtained with the epipolar representation are worse than those achieved by encoding the pseudo-video sequence format, but still better than the results achieved for the lenslet image representation format.

In comparison with the case where pseudo-video is used, MRP presents a lower performance decrease (0.35 bpp) than HEVC (0.55 bpp). The best results are obtained for both encoders using the 'Spiral-blackend' scan. This result may occur because the LRBF images are grouped together at the end of the pseudo-video sequence, *i.e.*, they are grouped in the epipolar images, thus improving the prediction.

Table 4
Coding performance for the EPI format (bpp).

Sequences	JPEG2000	JPEG-LS	CALIC	HEVC			MRP		
				Raster	Spiral	SBE	Raster	Spiral	SBE
Ankylosaurus_Diplodocus_1	9.56	8.83	9.35	9.07	8.79	8.43	8.17	8.01	**7.73**
Bikes	10.05	9.50	10.24	8.91	8.68	8.33	8.19	8.11	**7.82**
Color_Chart_1	9.97	8.83	9.62	8.99	8.76	8.41	7.98	7.82	**7.57**
Danger_de_Mort	11.04	9.30	9.96	8.43	8.19	7.84	7.84	7.76	**7.54**
Desktop	9.64	6.91	7.47	6.69	6.55	6.28	6.15	6.08	**5.85**
Flowers	10.69	8.86	9.74	8.60	8.44	8.05	7.93	7.97	**7.74**
Fountain_Vincent_2	11.33	9.82	10.42	9.44	9.32	8.97	8.52	8.55	**8.31**
Friends_1	10.34	8.03	8.92	7.86	7.57	7.21	7.19	7.02	**6.70**
ISO_Chart_12	10.94	9.40	10.05	9.21	8.96	8.60	8.40	8.29	**8.02**
Magnets_1	10.59	8.75	9.23	9.10	8.82	8.47	8.13	7.99	**7.70**
Stone_Pillars_Outside	8.29	9.26	10.03	8.70	8.49	8.06	8.23	8.19	**7.90**
Vespa	10.65	8.32	9.03	8.23	8.00	7.67	7.28	7.21	**7.00**
Average	10.26	8.82	9.51	8.60	8.38	8.03	7.83	7.75	**7.49**

Table 5
Average coding performance for component concatenation (bpp).

PVS			EPI		
Raster	Spiral	SBE	Raster	Spiral	SBE
7.11	**7.10**	7.11	7.75	7.68	7.43

5.3.4. Impact of visual content on coding efficiency

A cursory subjective evaluation was performed to understand if any characteristic of the light field image content has any consistent relationship with the coding performance. Direct observations indicate that the sequence with worse compression results for HEVC and MRP, *Fountain_Vincent_2*, presents many details in the bottom region of the image, making it more difficult to compress. This dependence on spatial frequency content seems to be confirmed by a similar analysis of the easier-to-compress *Desktop* image. This image has large smooth areas, which leads to higher compression ratios. Other than visual content with high spatial coding complexity, a more detailed visual analysis also reveals that the presence of structured texture can influence the coding efficiency when the lenslet and EPI data arrangements are used. That is the case of the *Fountain_Vincent_2* image, where the visible horizontally oriented textures apparently degrades the quality of the prediction and, consequently, decreases the compression efficiency. Further aspects are similar to those already known from traditional image coding methods.

5.4. Colour image coding in MRP

As pointed out earlier, the previous results were obtained for RGB LF images for all encoders. While in standard encoders, such as HEVC, the correlations between the colour components are already exploited, in MRP there is higher flexibility to exploit such similarities as pre-processing because it is a single-component encoder.

This was done by concatenating the three colour components from each frame of the pseudo-video sequence in order to form a composite frame to be encoded as a single plane with three times the number rows. Using a single plane, it is expected that the class block-based linear predictor design and the optimisation procedure of MRP are able to exploit the redundancies between different colour channels. This is mainly achieved by reducing the overhead necessary to repeat encoding of some classes in different colour channels and also by assigning the same class to blocks from different channels but with similar amplitude distribution properties. To test the effectiveness of this approach, the set of LF images represented as pseudo-video format, using the 'Spiral' image scan order, were encoded using MRP. The colour components of each frame were concatenated in two directions, side-by-side (horizontally) and top-bottom (vertically). As expected, due to MRP characteristics, the direction of this concatenation operation had little or no effect on the compression results, thus we present the results only for the vertical concatenation in Table 5.

As it can be seen, the results revealed just a slight increase of the MRP performance by 0.04 bpp, for sub-aperture images, and about 0.07 bpp (on average), for epipolar images, when compared to the independent encoding of the colour components, thus (weakly) confirming the previous intuition behind the idea of concatenating the colour planes. For lenslet images the results are of the same order of magnitude as those in Table 5.

5.5. Encoding and decoding time complexity analysis

The computational complexity was measured as the running time of the encoding and decoding operations for both the MRP and HEVC encoders. A dedicated server running Ubuntu Server 16.04 with an Intel Xeon E3-1240 V2 @ 3.40 GHz CPU and 23.5 GB of RAM was used for these experiments. The relative complexity of MRP is shown in Table 6 and discussed below. In these results, the running time of HEVC is used as reference, which means that HEVC complexity equals 1.0.

Table 6
Relative MRP encoding/ decoding running time.

Process	Lenslet	PVS			EPI		
		Raster	Spiral	SBE	Raster	Spiral	SBE
Encoding	99.1	3.7	3.8	4.1	2.9	3.0	3.4
Decoding	1.5	1.4	1.4	1.4	1.1	1.2	1.3

Due to the encoding algorithm optimisation mode, which runs for all image pixels at once, MRP is much more adversely affected by the increasing size of images. This effect is observed in the lenslet format, which is 99 times slower than HEVC. For the other LF formats, the running time complexity is much lower ranging from 2.9 times for EPI spiral to 4.1 times for PVS SBE. For decoding, the MRP algorithm exhibits a relative complexity very close to that of HEVC decoding. The worst case is again for the lenslet format, where MRP is 1.5 times slower than HEVC.

It should be noted that the MRP encoder is not optimised for speed, which largely justifies the relatively high complexity of MRP encoding in comparison with HEVC. As shown by these experiments, the disparity of processing speed between MRP and HEVC decoding is significantly smaller than for encoding. Nevertheless, an optimisation of the MRP decoder implementation could easily reduce the complexity difference to HEVC.

6. Conclusion

This paper presented a research study about LF image lossless coding using MRP and other state-of-the-art image and video encoders. Several methods for encoding LF image data were investigated, namely lenslet, stack of sub-aperture images and epipolar images, in order to assess and compare the coding performance of quite different encoding algorithms using various LF data formats.

It was found that the lenslet format results in the worst performance for nearly all cases, which leads to the conclusion that, if there is no strong reason for using such format, then it is better to use pseudo-video sequences of either sub-aperture or epipolar images, but preferably the former. The paper also demonstrates that the MRP algorithm with bi-directional predictors consistently achieves higher compression ratios in comparison with the most prominent state-of-the-art lossless image and video encoders.

As future work, the MRP algorithm may be further optimised to better exploit the lenslet structure of LF images, *e.g.* by adding specialized prediction support region shapes or data partitioning schemes beyond the current square blocks, in order to improve its performance beyond the levels already achieved when using the other data structures. Additionally, other colour spaces, rather than RGB, will be exploited to represent the LF images, such that the decorrelated data may be more efficiently compressed by the proposed minimum-rate predictors algorithm.

Acknowledgements

This work was supported in part by the Fundação para a Ciência e a Tecnologia (FCT), Portugal, under PhD Grant SFRH/BD/114894/2016 and by DERMOPLENO project in the scope of R&D Unit 50008, through national funds and when applicable co-funded by European Regional Development Fund (FEDER) – PT2020 partnership agreement.

References

[1] G. Lippmann, Epreuves reversibles donnant la sensation du relief, J. Phys. Theor.

Appl. 7 (1) (1908) 821–825.
[2] The (New) Stanford Light Field Archive. < http://lightfield.stanford.edu/ > (accessed 15 December 2017).
[3] M. Harris, Focusing on everything, IEEE Spectr. 49 (5) (2012) 44–50.
[4] M. Levoy, P. Hanrahan, Light field rendering, Conference on Computer Graphics and Interactive Techniques (SIGGRAPH), ACM, New York, NY, USA, 1996, pp. 31–42.
[5] C. Zhou, S.K. Nayar, Computational cameras: convergence of optics and processing, IEEE Trans. Image Process. 20 (12) (2011) 3322–3340.
[6] M. Levoy, Light fields and computational imaging, Computer 39 (8) (2006) 46–55.
[7] H.G. Jeon, J. Park, G. Choe, J. Park, Y. Bok, Y.W. Tai, I.S. Kweon, Accurate depth map estimation from a lenslet light field camera, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 1547–1555.
[8] D. Donatsch, S.A. Bigdeli, P. Robert, M. Zwicker, Hand-held 3D light field photography and applications, Visual Comput. 30 (6–8) (2014) 897.
[9] D.G. Dansereau, O. Pizarro, S.B. Williams, Linear volumetric focus for light field cameras, ACM Trans. Graph. (TOG) 34 (2) (2015) 1–20.
[10] X. Huang, E. Uffelman, O. Cossairt, M. Walton, A.K. Katsaggelos, Computational imaging for cultural heritage: recent developments in spectral imaging, 3-D surface measurement, image relighting, and X-ray mapping, IEEE Signal Process. Mag. 33 (5) (2016) 130–138.
[11] JPEG Pleno Call for Proposals on Light Field Coding, Tech. rep., 2016.
[12] A. Aggoun, A 3D DCT compression algorithm for omnidirectional integral images, in: IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP), vol. 2, 2006, pp. II.
[13] A. Aggoun, Compression of 3D integral images using 3D wavelet transform, J. Display Technol. 7 (11) (2011) 586–592.
[14] C. Conti, J. Lino, P. Nunes, L.D. Soares, P.L. Correia, Spatial prediction based on self-similarity compensation for 3D holoscopic image and video coding, in: IEEE International Conference on Image Processing (ICIP), 2011, pp. 961–964.
[15] C. Conti, P. Nunes, L.D. Soares, Inter-layer prediction scheme for scalable 3-D holoscopic video coding, IEEE Signal Process. Lett. 20 (8) (2013) 819–822.
[16] S. Shi, P. Gioia, G. Madec, Efficient compression method for integral images using multi-view video coding, in: IEEE International Conference on Image Processing (ICIP), 2011, pp. 137–140.
[17] N.C. Francisco, N.M.M. Rodrigues, E.A.B. da Silva, M.B. de Carvalho, S.M.M. de Faria, Video compression using 3D multiscale recurrent patterns, in: IEEE International Symposium on Circuits and Systems (ISCAS), 2013, pp. 1412–1415.
[18] L.F.R. Lucas, C. Conti, P. Nunes, L.D. Soares, N.M.M. Rodrigues, C.L. Pagliari, E.A.B. d. Silva, S.M.M. de Faria, Locally linear embedding-based prediction for 3D holoscopic image coding using HEVC, in: European Signal Processing Conference (EUSIPCO), 2014, pp. 11–15.
[19] C. Perra, Lossless plenoptic image compression using adaptive block differential prediction, in: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2015, pp. 1231–1234.
[20] P. Helin, P. Astola, B. Rao, I. Tabus, Sparse modelling and predictive coding of subaperture images for lossless plenoptic image compression, in: 3DTV-Conference: The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON), 2016, pp. 1–4. http://dx.doi.org/10.1109/3DTV.2016.7548953.
[21] A. Vieira, H. Duarte, C. Perra, L. Tavora, P. Assuncao, Data formats for high efficiency coding of Lytro-Illum light fields, in: International Conference on Image Processing Theory, Tools and Applications (IPTA), 2015, pp. 494–497.
[22] I. Matsuda, H. Mori, J. Maeda, S. Itoh, Design and evaluation of minimum-rate predictors for lossless image coding, Syst. Comput. Jpn. 38 (5) (2007) 90–98.
[23] I. Matsuda, Y. Umezu, N. Ozaki, J. Maeda, S. Itoh, A lossless coding scheme using adaptive predictors and arithmetic code optimized for each image, Syst. Comput. Jpn. 38 (4) (2007) 1–11.
[24] J.M. Santos, A.F.R. Guarda, L.A. da Silva Cruz, N.M.M. Rodrigues, S.M.M. de Faria, Compression of medical images using MRP with bi-directional prediction and histogram packing, in: Picture Coding Symposium (PCS), 2016.
[25] G.N.N. Martin, Range encoding: an algorithm for removing redundancy from a digitized message, in: Video and Data Recording Conference, Southampton, UK, 1979.
[26] M. Rerabek, T. Ebrahimi, New Light Field Image Dataset, in: International Conference on Quality of Multimedia Experience (QoMEX), 2016.
[27] S. Wanner, J. Fehr, B. Jhne, Generating EPI Representations of 4D Light Fields with a Single Lens Focused Plenoptic Camera, Springer, Berlin, Heidelberg, 2011.
[28] M.J. Weinberger, G. Seroussi, G. Sapiro, The LOCO-I lossless image compression algorithm: principles and standardization into JPEG-LS, IEEE Trans. Image Process. 9 (8) (2000) 1309–1324.
[29] ITU, ITU-T Recommendation T.812: Information technology JPEG 2000 image coding system: An entry level JPEG 2000 encoder, Itu, International Telecommunication Union, Geneva, Switzerland, 2007.
[30] X. Wu, N. Memon, Context-based, adaptive, lossless image coding, IEEE Trans. Commun. 45 (4) (1997) 437–444.
[31] G.J. Sullivan, J.M. Boyce, Y. Chen, J.R. Ohm, C.A. Segall, A. Vetro, Standardized extensions of high efficiency video coding (HEVC), IEEE J. Sel. Topics Signal Process. 7 (6) (2013) 1001–1016.
[32] JPEG-LS encoder implementation. < https://github.com/thorfdbg/libjpeg > (accessed 15 December 2017).
[33] Openjpeg JPEG2000 encoder v. 2.1.2. < http://www.openjpeg.org/ > (accessed 15 December 2017).
[34] M. Briano, CALIC, 3D CALIC and M-CALIC customizable software implementation. < https://prezi.com/j2ichhip4ls7/calic-implementation/ > (accessed 15 December 2017).
[35] HEVC reference software encoder HM-16.12 + RExt. < https://hevc.hhi.fraunhofer.de/ > (accessed 15 December 2017).
[36] D. Dansereau, Light Field Toolbox v0.4. < https://www.mathworks.com/matlabcentral/fileexchange/49683-light-field-toolbox-v0-4 > (accessed 15 December 2017).
[37] D.G. Dansereau, O. Pizarro, S.B. Williams, Decoding, calibration and rectification for lenselet-based plenoptic cameras, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2013.