SPARSE MODELLING AND PREDICTIVE CODING OF SUBAPERTURE IMAGES FOR LOSSLESS PLENOPTIC IMAGE COMPRESSION

Petri Helin[⋆], *Pekka Astola*[⋆], *Bhaskar Rao*[†], *Ioan Tabus*[⋆]

[⋆]Department of Signal Processing, Tampere University of Technology, Tampere, FINLAND
[†]Department of Electrical and Computer Engineering, University of California, San Diego, CA, USA

ABSTRACT

This paper studies the lossless compression of rectified light-field images captured by plenoptic cameras, exploiting the high similarity existing between the subaperture images, or views, composing the light-field image. The encoding is predictive, where one sparse predictor is designed for every region of a view, using as regressors the pixels from the already transmitted views. As a first step, consistent segmentations for all subaperture images are constructed, defining the regions as connected components in the quantized depth map of the central view, and then propagating them to all side views. The sparse predictors are able to take into account the small horizontal and vertical disparities between regions in corresponding close-by views and perform optimal least squares interpolation accounting implicitly for fractional disparities. The optimal structure of the sparse predictor is selected for each region based on an implementable description length. The encoding of the views is done sequentially starting from the central view and the scheme produces results better than standard lossless compression methods utilized directly on the full light-field image or applied to the views in a similar sequential order as our method.

Index Terms — lossless compression, light-field coding, depth map warping, plenoptics, sparse prediction

1. INTRODUCTION

Plenoptic cameras have received recently wide attention as an alternative to e.g. stereo camera setups in capturing 3D information from a scene. Plenoptic imaging, also refered to as light-field or holoscopic imaging, was first introduced as a concept by Lippmann in [1]. The plenoptic camera differs from a conventional camera by having an array of microlenses interposed in between the main lens and the sensor, enabling the sensing of angular information which has proven useful in numerous applications such as 3D optical microscopy, computer vision applications, 3D underwater imaging and medical imaging [2].

Recently there is a wide interest in studying the compressibility of the captured light-field, e.g. a new study action known as JPEG PLENO [3] was started by the JPEG group and a related challenge for lossy compression of plenoptic measurements was organized at ICME 2016. The lossless compression case was not extensively studied, while the lossy case is already carefully investigated see e.g., [4] and [5] and the references therein.

In plenoptic applications, the sensed image captured by the photosensor of a plenoptic camera is processed in a chain consisting of demosaicing, devignetting, alignment to a square grid and rectification, and the result is rearranged into a 5D light-field structure L_F, where an element $L_F(i, j, x, y, c)$ has the following indices: $i, j \in \{1, \ldots, N\}$ are indices for a certain pixel under a microlens (hence addressing a particular subaperture image) and x, y correspond to the spatial pixel location in the subaperture image. Hence, there are $N \times N$ subaperture images (views) of size $\xi \times \eta$ where the $(i, j)th$ view is A_{ij} with elements $A_{ij}(x, y, c)$ at the pixel (x, y) and color component $c \in \{0, 1, 2\}$ denoting hereafter a component in RGB space. For example we experiment with plenoptic images from Lytro Illum, where $N = 15, \xi = 434$ and $\eta = 625$. We refer to the view with index $(i_c, j_c) = (8, 8)$ as the center view and to the other 224 views as side views. The task considered in this paper is to compress simultaneously $N \times N = 225$ views losslessly, where the close-by views are extremely similar.

We utilize the redundancy between one side view and its neighbours by proposing a predictive scheme where one optimal sparse predictor is designed for each region from a segmentation of the side view. The segmentations of all side views are based on the quantized depth map and are obtained in the first stage, by propagating the regions using optimal disparities along x and y coordinates and suitably filling the missing regions that may result after warping. In the encoding stage, done sequentially over all side views, one region of a side view is predicted using an optimal sparse predictor having as regressors pixels from several side views, accounting for small integer or fractional disparities.

For obtaining the light-field from the camera measurements we use the rectification tools described in [6], but any other rectification method can be used and the decoder does not need to known which method was used and does not need to repeat the rectification processing. If the task would be to encode the *initial sensor image* (instead of the rectified light-field image as we do here) the decoder has to re-run the whole rectification process, which is not possible when some of the processing steps from sensor image to rectified light-field image are not known at the decoder.

Our method makes use of the depth information of the scene, which is encoded along in the bitstream efficiently using [7]. The depth map for plenoptic images can be estimated with numerous methods, see e.g. [8]. In this paper we use the depth map provided by the camera manufacturer's software, namely Lytro Desktop.

Plenoptic images captured with a focused plenoptic camera were compressed losslessly by Perra [9], where the compression of the non-rectified image obtained directly at the sensor is studied. As opposed to [9], we investigate the compressibility of light-field after the rectification of the sensor measurements and we exploit the redundancy of the subaperture images.

In Section 2 the general structure of the proposed method is given. In Section 3 the predictive scheme is discussed in detail. Section 4 illustrates the performance of the method and Section 5 concludes the paper.

The work was supported by Academy of Finland, project "Integrated multimodal sensing of 3D environment for intelligent manipulators".

2. COMPRESSION ARCHITECTURE

2.1. Definitions and notations

For the central view (i_c, j_c) we are given an estimation of the depth map which we quantize on K levels resulting in a $\xi \times \eta$ quantized depth map $Z_{i_c j_c}(x, y) \in \{0, 1, \ldots, K-1\}$ where 0 is the closest depth and $K-1$ is the farthest. Similarly Z_{ij} is the quantized depth map associated to the view (i, j) and is obtained with Alg. 1, see below. The set of pixels of any view, is denoted $\Omega = \{0, \ldots, \xi - 1\} \times \{0, \ldots, \eta - 1\}$. The depth map Z_{ij} induces a partition $\{\Omega^0_{ij}, \ldots, \Omega^{K-1}_{ij}\}$ of Ω, where $\Omega^k_{ij} = \{(x, y)|Z_{ij}(x, y) = k\}$ (hence Z_{ij} is a label image for the partition $\{\Omega^0_{ij}, \ldots, \Omega^{K-1}_{ij}\}$).

We find useful to work also with a finer partition, where each set Ω^k_{ij} is split into its connected components $\{R^{kl}_{ij}\}, l = 0, \ldots, \lambda^k_{ij}$.

We introduce the horizontal and vertical disparity of a region Ω^k_{ij} in the following way. The horizontal disparity δ^k_{ij} and the vertical disparity μ^k_{ij} warp a region $\Omega^k_{i_c j_c}$ of the view $A_{i_c j_c}$ into a region Ω^k_{ij} of the view A_{ij} such that for any non-occluded pixel $(x, y) \in \Omega^k_{i_c j_c}$ ideally $A_{i_c j_c}(x, y) \approx A_{ij}(x + \delta^k_{ij}, y + \mu^k_{ij})$.

2.2. Encoding scheme

The encoder processes the views sequentially, starting with the central view $A_{i_c j_c}$ which is encoded independently using the lossless mode of JPEG 2000. Also the quantized depth map $Z_{i_c j_c}$ is encoded, using CERV algorithm [7]. Then each side view A_{ij} is encoded conditionally on the already transmitted views, as described in the following.

First, we construct the partition of a side view into regions $\{\Omega^0_{ij}, \ldots, \Omega^{K-1}_{ij}\}$ by using the partition of the central view $\{\Omega^0_{i_c j_c}, \ldots, \Omega^{K-1}_{i_c j_c}\}$ (having the label image $Z_{i_c j_c}$) and warping each region with disparities found by minimizing the Mean Square Error (MSE) in the color views:

$$(\delta^k_{ij}, \mu^k_{ij}) = \arg\min_{\delta,\mu} \sum_{x,y \in \Omega^k_{i_c j_c}} \sum_{c=0}^{2} (A_{i_c j_c}(x, y, c) - A_{ij}(x + \delta, y + \mu, c))^2 . \quad (1)$$

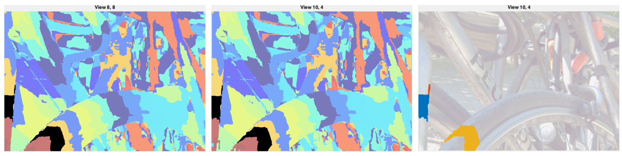
Using these disparities, the constant depth regions of the center view are warped to the side view as described in Alg. 1. The warping operation may create a set of missing pixels due to disocclusions, denoted by Φ. We found an easy and effective way of treating missing pixels by allocating them to the nearby region (with respect to the coordinates x and y) located at the farthest depth, as described in steps 2.2–2.4 in Alg. 1. The regions in a partition and the partition propagation are illustrated in Fig. 1.

The chosen effective processing order for the views, also mentioned in e.g. [10], that discusses the compression of similar plenoptic images, is a spiral sequence depicted in Fig. 2. We denote Γ_{ij} the set of pairs (i', j') indexing the views that are available when designing a predictor for the view A_{ij}; e.g. when predicting the view (9,7), we have $\Gamma_{9,7} = \{(8, 8), (7, 8), (7, 7), (8, 7)\}$, see Fig. 2. Since the number of available regressors grows as we move along the spiral, we limit the cardinality Γ so that $|\Gamma| \leq 5$ by choosing the nearest views i.e. the views (i', j') for which $|i - i'| + |j - j'|$ is smallest.

The processing flow for encoding one region Ω^k_{ij} is depicted in Fig. 3 where the best disparities given by (1) first are transmitted for the whole region Ω^k_{ij}. Then the region is split into its connected components (see Fig. 1c) and each of them is encoded in a predictive way, described in Fig. 4. All regions smaller than 50 pixels are merged into a single region (one for each side view), treated as any other region R^{kl}_{ij}.

Algorithm 1 Partition propagation from central view (i_c, j_c) to side view (i, j).

Input: Depth map for the center view $Z_{i_c j_c}$ which labels the partition $\{\Omega^0_{i_c j_c}, \ldots, \Omega^{K-1}_{i_c j_c}\}$; disparities δ^k_{ij} and μ^k_{ij}, for $k = 0, \ldots, K-1$.
1. For $k = K-1, \ldots, 0$
  For each $(x, y) \in \Omega^k_{i_c, j_c}$
    1.1 Set $x' = x + \delta^k_{ij}$ and $y' = y + \mu^k_{ij}$.
    1.2 Insert (x', y') into Ω^k_{ij}.
2. Fill missing pixels $\Phi = \Omega \setminus \{\Omega^0_{ij}, \ldots, \Omega^{K-1}_{ij}\}$ to form partition:
  For $k = K-1, \ldots, 0$:
    2.1 Dilate the set Ω^k_{ij} with a disk of radius r to obtain region Φ_1.
    2.2 $\Phi_2 = \Phi \cap \Phi_1$ (pick from dilated region only missing pixels).
    2.3 $\Omega^k_{ij} = \Omega^k_{ij} \cup \Phi_2$ (allocate to set Ω^k_{ij}).
    2.4 $\Phi = \Phi \setminus \Phi_2$ (discard Φ_2 from missing pixels).
Output: $\{\Omega^0_{ij}, \ldots, \Omega^{K-1}_{ij}\}$.



(a) Regions in center view. (b) Propagated regions in the side view. (c) Regions at depth $Z = 3$.

Figure 1: a) Segmentation of the depth map of the center view in $K = 32$ quantization levels. The three connected components (regions) forming the region $\Omega^3_{8,8}$ at depth $Z = 3$ are shown in black. b) Propagation of the depth segments to the side view $(10, 4)$. Region $\Omega^3_{10,4}$ is shown in black. c) Location of the three connected component regions: $R^{3,0}_{10,4}$(orange), $R^{3,1}_{10,4}$ (blue), and $R^{3,2}_{10,4}$ (red) overlaid over the color side view image.

3. SPARSE DICTIONARY DESIGN AND SPARSE MODELLING

A specific sparse predictor is designed for the pixels in the finest partition, e.g. for a region R^{kl}_{ij}, where prediction of the color components at (x, y) is done conditionally on the causal neighborhood of this pixel and also on the neighborhoods of the pixel in the available side views $(i', j') \in \Gamma_{ij}$. The prediction is done independently for each color component. For the region R^{kl}_{ij} we introduce the sparse modeling problem as

$$\text{minimize}_{\boldsymbol{\theta}} \quad \|\boldsymbol{y} - \boldsymbol{D}\boldsymbol{\theta}\|_2 \quad \text{s.t.} \quad \|\boldsymbol{\theta}\|_0 \leq \kappa \quad (2)$$

and describe the elements in dictionary $\boldsymbol{D}$ and in the desired vector $\boldsymbol{y}$ one row at a time where each row corresponds to a pixel $(x, y) \in R^{kl}_{ij}$.

Let r be the row in $\boldsymbol{D}$ and $\boldsymbol{y}$ corresponding to the pixel $(x, y) \in R^{kl}_{ij}$. The element r of the desired vector is

$$y_r = A_{ij}(x, y, c). \quad (3)$$

The elements of the r-th row of the matrix $\boldsymbol{D}$ are in order

1. The constant element 1 compensating for the bias.
2. The N,W,NW and NE causal neighbors of (x, y) from the current view (e.g. the north element is $A_{ij}(x, y-1, c)$).
3. For all views $(i', j') \in \Gamma_{ij}$:
   A row vector consisting of the elements of $A_{i'j'}$ from the full 3×3 neighborhood of (x, y) (e.g. if we denote $C = A_{i'j'}$, the row vector is
   $$[C_{x,y}\ C_{x-1,y}\ C_{x-1,y-1}\ C_{x-1,y+1}\ C_{x,y-1}\ C_{x,y+1}\ C_{x+1,y-1}\ C_{x+1,y}\ C_{x+1,y+1}]$$
   where we omitted the index for color and denote as subscripts the pixel indices).

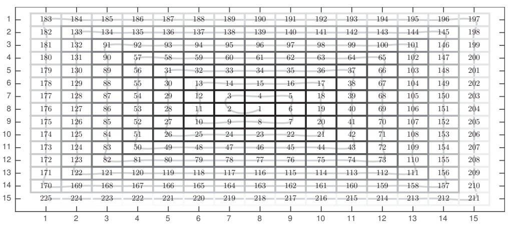


Figure 2: The processing order of the views. The index inside the rectangles gives the processing order number while the vertical axis represents i and horizontal axis represents j.

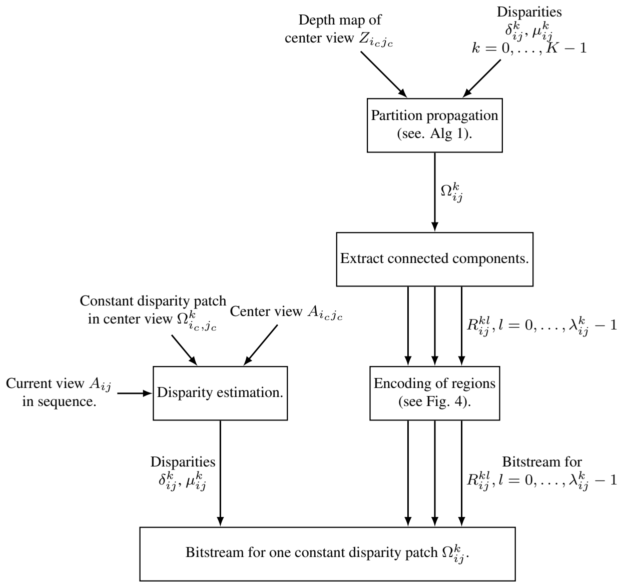


Figure 3: Encoding a constant disparity patch Ω^k_{ij} in sideview (i, j).

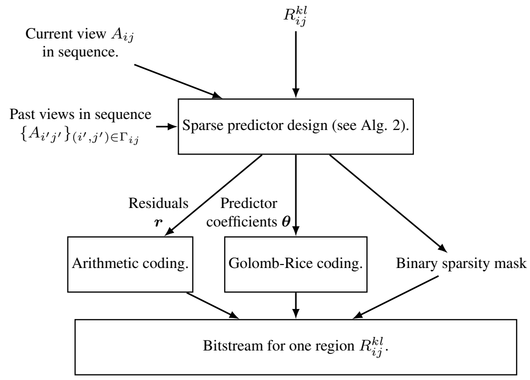


Figure 4: Encoding the region R^{kl}_{ij} in side view (i, j).

Algorithm 2 Sparse prediction.

Input: desired vector $\boldsymbol{y}$, regressors $\boldsymbol{D}$, maximum order of prediction $\kappa_{\max}$.
Initialize $\kappa = 1$, the set of selected indices $\mathcal{I}_0 = \emptyset$ and the residuals $\boldsymbol{r} = \boldsymbol{y}$.
1. For each column of $\boldsymbol{D}$, denoted $\boldsymbol{d}_i$, compute the criterion

$$c_i = \frac{|\boldsymbol{r}^T \boldsymbol{d}_i|}{\|\boldsymbol{d}_i\|}$$

2. Append the index i that maximizes the criterion to the set of selected indices $\mathcal{I}_\kappa = \mathcal{I}_{\kappa-1} \cup \arg\max_i c_i$.
3. Compute the nonzero values of the predictor $\boldsymbol{\theta}_\kappa = \boldsymbol{D}^\dagger_{\mathcal{I}_\kappa} \boldsymbol{y}$ where $\boldsymbol{D}^\dagger$ is the Moore-Penrose pseudoinverse of $\boldsymbol{D}$.
4. Quantize $\boldsymbol{\theta}_k$ to 10 bits and update residual $\boldsymbol{r} = \lfloor \boldsymbol{y} - \boldsymbol{D}_{\mathcal{I}_\kappa} \boldsymbol{\theta}_\kappa \rceil$.
5. Compute the code length $\text{CL}_{\boldsymbol{r}} + \text{CL}_{\text{mask}} + \text{CL}_{\boldsymbol{\theta}_\kappa}$.
6. If $\kappa < \kappa_{\max}$, set $\kappa = \kappa + 1$ and go to step 1.
7. Compute $\hat{\kappa}$ as in (4).
Output: The nonzero coefficients $\boldsymbol{\theta}_{\hat{\kappa}}$, and the sparsity mask corresponding to $\mathcal{I}_{\hat{\kappa}}$.

An example of the elements of a row in $\boldsymbol{D}$ is shown in Fig. 5. When a pixel from the 3×3 neighborhood does not belong to the region R^{kl}_{ij} the value at that pixel is replaced with the value of the nearest pixel in the neighborhood that belongs to the region (in the same way as in [11]).

3.1. Sparse prediction with code length based order selection

As is well known, the problem (2) is non-convex and fast algorithmic solutions require a relaxation e.g. by solving the ℓ_1-norm instead of the 0-norm or by using a greedy algorithm to select relevant regressors from the dictionary one at a time. For the compression, it is crucial to have fine granularity when increasing the sparsity degree (or order κ) which is why we opt to use Orthogonal Matching Pursuit (OMP) with a model order selection criterion as follows. Optimal order $\hat{\kappa}$, i.e. the number of nonzero coefficients in $\boldsymbol{\theta}$, for prediction is chosen using code length optimization, i.e.

$$\hat{\kappa} = \arg\min_{\kappa} \quad \text{CL}_{\boldsymbol{r}} + \text{CL}_{\text{mask}} + \text{CL}_{\boldsymbol{\theta}_\kappa}, \tag{4}$$

where $\text{CL}_{\boldsymbol{r}}$ is the number of bits required to code the prediction residuals while CL_{mask} corresponds similarly to the binary sparsity mask and $\text{CL}_{\boldsymbol{\theta}_\kappa}$ to the nonzero predictor values.

The compression scheme uses context modeling and arithmetic coding. Ignoring the effect of contexts, the code length when the residuals $\boldsymbol{r} = \lfloor \boldsymbol{y} - \boldsymbol{D}\boldsymbol{\theta}_\kappa \rceil$ are entropy coded with adaptive arithmetic coding with Laplace's rule of succession is

$$\text{CL}_{\boldsymbol{r}} = -\log_2 \frac{\prod_{i=1}^{n_s} n_i!}{(n + n_s - 1)!/(n_s - 1)!} \tag{5}$$

where n_s is the number of symbols, in this case $n_s = 511$, n_i is the final count of occurrences for symbol i in the residual vector $\boldsymbol{r}$, and n is the size of the region R^{kl}_{ij}.

The mask is transmitted to the decoder by simply sending a zero if the sparse coefficient is zero and a one otherwise. Thus $\text{CL}_{\text{mask}} = n$ where n is the number of regressors, typically $n = 50$. The cost of coefficients is $\text{CL}_{\boldsymbol{\theta}_\kappa} = 10\kappa$ since the coefficients are quantized to 10 bits.

The sparse prediction is summarized in Alg. 2.

3.2. Entropy coding

As also seen in Fig. 4, after the predictive decorrelation stage, the predictor coefficients are encoded with Golomb-Rice coding and the binary sparsity mask is simply copied as a side information header. The prediction residuals are coded using context modeling and adaptive arithmetic coding similar to [11]. We compute contexts using quantized gradients classified into 16 different contexts.

Overall the information needed to be sent to the decoder is: the color components of the central view $A_{i_c j_c}$ encoded by JPEG 2000; the quantized depth map of the center view $Z_{i_c j_c}$; for each (i, j) the disparities δ^k_{ij} and μ^k_{ij}; for each (i, j) the prediction results i.e., prediction mask, predictor coefficients and residuals.

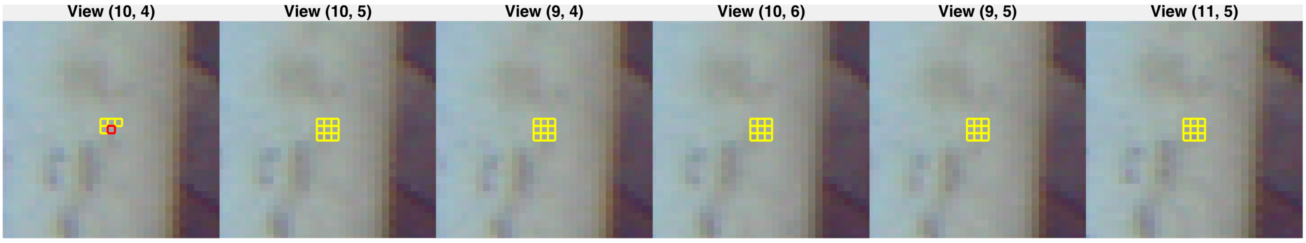


Figure 5: The regressors. The red pixel in view (10,4) is predicted and has a causal mask shown in yellow in the same figure. The five other views are the closest in the spiral setting. The views are available at the decoder so we can use the full template, shown in yellow, in prediction. It can be seen that close-by views are very similar.

Table 1: Compressed file sizes in mega bytes (MB).

Image	HEVC	AVC	CALIC	FP8	proposed
Bikes	82.08	80.81	82.47	82.99	**69.66**
Danger_de_Mort	87.06	86.53	88.63	86.75	**77.06**
Flowers	86.99	86.36	88.65	86.06	**78.14**
Stone_Pillars_Outside	82.02	80.23	81.30	78.76	**67.28**
Vespa	84.29	82.76	87.94	83.23	**71.90**
Ankylosaurus_&_D...	71.32	69.82	75.82	68.40	**58.99**
Desktop	56.61	55.94	63.45	52.86	**50.79**
Magnets_1	70.67	69.16	75.95	67.17	**59.56**
Fountain_&_Vincent_2	85.42	84.29	85.75	85.67	**70.20**
Friends_1	68.99	67.70	75.20	66.92	**60.19**
Color_Chart_1	83.31	81.92	93.36	78.92	**71.29**
ISO_Chart_12	78.38	78.48	76.85	75.04	**71.40**

4. RESULTS

For experimental results, data from Light-Field Image Dataset[1], containing raw data captured with a Lytro Illum camera, is used. We select the same subset of images as in the recent challenge at ICME 2016.

We compare the performance of the proposed compressor to existing generic tools for compressing L_F losslessly. The views can be treated as a pseudo-video-sequence and compressed with HEVC or AVC using their lossless modes. For maximizing their performance we used the preset 'veryslow' and the same spiral (see Fig. 2) sequence as for our method.

Another approach is to rearrange the views in a single image I such that $I((x-1)N+i,(y-1)N+j,c) = A_{ij}(x,y,c)$, where $i = 1,\ldots,N$, $j = 1,\ldots,N$, $x = 0,\ldots,\xi-1$, $y = 0,\ldots,\eta-1$ and $c = 0,1,2$ [10]. This structure can be efficiently compressed with tools such as JPEG 2000, CALIC [12] or FP8 compressor[2].

Our encoder and decoder were implemented with a combination of Matlab and C. The reported file sizes are directory sizes containing all the neccessary files for decoding and lossless decoding was checked. The size of the raw 24 bit data is 183 MB and the compressed sizes are given in Table 1. Results for JPEG2000 or JPEG-LS are not shown because CALIC outperforms them for all the images. It can be seen that our coding scheme achieves for every image the best results, improving significantly the compression rate over the generic compressors.

5. CONCLUSIONS

We conclude that partitioning the side views according to the quantized depth map provides essential regions, for which optimal LS sparse predictors can be designed to capture very efficiently the similarities between the current view and its neighbor views, resulting in a very efficient overall predictive coding. Being able to handle regions with arbitrary shapes, our method is more flexible and outperforms the methods operating over rectangular blocks.

[1] Available at http://mmspg.epfl.ch/EPFL-light-field-image-dataset.
[2] Available at http://mattmahoney.net/dc/text.html#1532.

6. REFERENCES

[1] G. Lippmann, "Epreuves reversibles donnant la sensation du relief," *Journal of Physics*, vol. 7, no. 4, pp. 821–825, 1908.

[2] X. Xiao, B. Javidi, M. Martinez-Corral, and A. Stern, "Advances in three-dimensional integral imaging: sensing, display, and applications [invited]," *Appl. Opt.*, vol. 52, no. 4, pp. 546–560, Feb 2013.

[3] T. Ebrahimi, "Jpeg pleno abstract and executive summary," in *ISO/IEC JTC 1/SC 29/WG1 N6922*, Sydney, Australia, 2015.

[4] C. Conti, L. Soares, and P. Nunes, "HEVC-based 3D holoscopic video coding using self-similarity compensated prediction," *Signal Processing: Image Communication*, vol. 42, pp. 59 – 78, 2016.

[5] Y. Li, M. Sjöström, R. Olsson, and U. Jennehag, "Scalable coding of plenoptic images by using a sparse set and disparities," *IEEE Trans. Image Process.*, vol. 25, no. 1, pp. 80–91, Jan 2016.

[6] D. G. Dansereau, O. Pizarro, and S. B. Williams, "Decoding, calibration and rectification for lenselet-based plenoptic cameras," in *CVPR*, June 2013, pp. 1027–1034.

[7] I. Tabus, I. Schiopu, and J. Astola, "Context coding of depth map images under the piecewise-constant image model representation," *IEEE Trans. Image Process.*, vol. 22, no. 11, pp. 4195–4210, Nov 2013.

[8] T. C. Wang, A. A. Efros, and R. Ramamoorthi, "Occlusion-aware depth estimation using light-field cameras," in *ICCV*, Dec 2015, pp. 3487–3495.

[9] C. Perra, "Lossless plenoptic image compression using adaptive block differential prediction," in *ICASSP*, April 2015, pp. 1231–1234.

[10] A. Vieira, H. Duarte, C. Perra, L. Tavora, and P. Assuncão, "Data formats for high efficiency coding of lytro-illum light fields," in *IPTA*, Nov 2015, pp. 494–497.

[11] I. Tabus and P. Astola, "Sparse prediction for compression of stereo color images conditional on constant disparity patches," in *3DTV-CON*, July 2014, pp. 1–4.

[12] X. Wu and N. Memon, "Context-based lossless interband compression-extending calic," *IEEE Trans. Image Process.*, vol. 9, no. 6, pp. 994–1001, Jun 2000.