Accurate Disparity Estimation for Plenoptic Images

Neus Sabater(✉), Mozhdeh Seifi, Valter Drazic,
Gustavo Sandri, and Patrick Pérez

Technicolor, 975 Av. des Champs Blancs, 35576 Cesson-Sévigné, France
Neus.Sabater@technicolor.com

Abstract. In this paper we propose a post-processing pipeline to recover accurately the views (light-field) from the raw data of a plenoptic camera such as Lytro and to estimate disparity maps in a novel way from such a light-field. First, the microlens centers are estimated and then the raw image is demultiplexed without demosaicking it beforehand. Then, we present a new block-matching algorithm to estimate disparities for the mosaicked plenoptic views. Our algorithm exploits at best the configuration given by the plenoptic camera: (i) the views are horizontally and vertically rectified and have the same baseline, and therefore (ii) at each point, the vertical and horizontal disparities are the same. Our strategy of demultiplexing without demosaicking avoids image artifacts due to view cross-talk and helps estimating more accurate disparity maps. Finally, we compare our results with state-of-the-art methods.

Keywords: Plenoptic camera · Raw-data conversion · Disparity estimation

1 Introduction

Plenoptic cameras are gaining a lot of popularity in the field of computational photography because of the additional information they capture compared to traditional cameras. Indeed, they are able to measure the amount of light traveling along each ray bundle that intersects the sensor, thanks to a microlens array placed between the main lens and the sensor. As a result, such cameras have novel post-capture processing capabilities (e.g., depth estimation and refocusing). There are several optical designs for plenoptic cameras depending on the distance between the microlens array and the sensor. If this distance is equal to the microlenses focal length it is called a type 1.0 plenoptic camera [17]; and type 2.0 (or focused) plenoptic camera [16] otherwise. In the first case the number of pixels per rendered view[1] is equal to the number of microlenses (only one pixel per microlens is rendered on each view). In the second case, the rendered views have a higher spatial resolution, but that comes at the cost of decreasing the angular resolution. Depending on the application, one camera or another would be preferred. In this paper we focus on type 1.0 plenoptic cameras.

[1] The terms *view* and *sub-aperture image* are equally used in the literature.

L. Agapito et al. (Eds.): ECCV 2014 Workshops, Part II, LNCS 8926, pp. 548–560, 2015.
DOI: 10.1007/978-3-319-16181-5_42

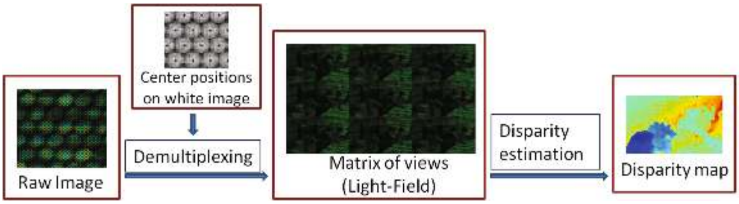

Raw Image
Center positions on white image
Demultiplexing
Matrix of views (Light-Field)
Disparity estimation
Disparity map

Fig. 1. Pipeline of our method. For visualization purposes only a part of the subimages and the views are shown. The LF is obtained by demultiplexing mosaicked data using the center subimage positions. Then the accurate disparity map for a reference view is estimated from the LF.

The concept of *integral photography*, which is the underlying technology in plenoptic cameras was introduced in [15] and then brought up to computer vision in [3], but it has recently become practical with the hand-held cameras that Lytro[2] and Raytrix[3] have put on the market for the mass market and professionals respectively. Since then, the scientific community has taken an interest in the LF (Light-Field) technology. Recent studies in the field address the bottleneck of the plenoptic cameras, namely the resolution problem ([10], [5], [18] and [24]). Besides super-resolution, depth estimation has also been investigated as a natural application of plenoptic images ([5], [24] and [22]). Indeed, the intrinsic information of the LF has the advantage to allow disparity computation without the image calibration and rectification steps required in classic binocular stereo or multi-view algorithms, making it an enormous advantage for 3D applications. However, the last cited works consider the sampled LF (the set of demultiplexed views) as input for their disparity estimation methods, meaning that they do not study the process that converts the raw data acquired by the plenoptic camera into the set of demultiplexed views. In this paper we show that such processing, called *demultiplexing*, is of paramount importance for depth estimation.

The contributions of this paper are twofold. First, we model the demultiplexing process of images acquired with a Lytro camera and then we present a novel algorithm for disparity estimation specially designed for the singular qualities of plenoptic data. In particular, we show that estimating disparities from mosaicked views is preferred to using views obtained through conventional linear demosaicking on the raw data. Therefore, for the sake of accurate disparity estimation, demosaicking is not performed in our method (see our pipeline in Fig. 1). To the best of our knowledge this approach has never been proposed before.

2 Related Work

The closest works to our demultiplexing method have been published recently. In [7] a demultiplexing algorithm followed by a rectification step where lens

[2] http://www.lytro.com

[3] http://www.raytrix.de

distortions are corrected using a 15-parameter camera model is proposed. In [6], the authors also proposed a demultiplexing algorithm for the Lytro camera and studied several interpolation methods to superresolve the reconstructed images. On the contrary, [9] recovers the refocused Lytro images via splatting without demultiplexing the views.

Considering disparity estimation for plenoptic images, several works have proposed a variational method ([24], [4], [5] , [13] and [23]). In particular, [24] uses the epipolar plane image (EPI), [4] and [5] propose an antialiasing filtering to avoid cross-talk image artifacts and [13] combines the idea of Active Wavefront Sampling (AWS) with the LF technique. In fact, variational methods better deal with the noise in the images but they are computationally expensive. Given the large number of views on the LF, such approaches are not suitable for many of applications. In addition to variational approaches, other methods have been proposed for disparity estimation. [14] estimates disparity maps from high spatio-angular LF with a fine-to-coarse algorithm where disparities around object boundaries are first estimated using an EPI-based method and then propagated. [22] proposes an interesting approach that combines defocus and correspondence to estimate the scene depth. Finally, [25] presents a Line-Assisted Graph-Cut method in which line segments with known disparities are used as hard constraints in the graph-cut algorithm.

In each section we shall discuss the differences between our method and the most related works on demultiplexing and disparity estimation methods on Lytro data. While demosaicking is not the goal of this paper, note that [10] already pointed out artifacts due to raw plenoptic data demosaicking and that a practical solution was proposed by [26] for type 2.0 plenoptic data.

3 Demultiplexing RAW Data

Demultiplexing (also called ”decoding” [7] or ”calibration and decoding” [6]) is data conversion from the 2D raw image to the 4D LF, usually represented by the two-plane parametrization [12]. In particular, demultiplexing consists in reorganizing the pixels of the raw image[4] in such a way that all pixels capturing the light rays with a certain angle of incidence are stored in the same image creating the so-called *views*. Each view is a projection of the scene under a different angle. The set of views create a block matrix where the central view stores the pixels capturing the light rays perpendicular to the sensor. In fact, in plenoptic type 1.0, the angular information of the light rays is given by the relative pixel positions in the *subimages*[5] with respect to the subimage centers. After demultiplexing, the number of restored views (entries of the block matrix) corresponds to the number of pixels covered by one microlens and each restored view has as many pixels as the number of microlenses.

Estimating Subimage Centers: In a plenoptic camera such as Lytro the microlens centers are not necessarily well aligned with the pixels of the sensor.

[4] We use the tool in [1] to access the raw data from Lytro.

[5] The image that is formed under a microlens and on the sensor is called a subimage.

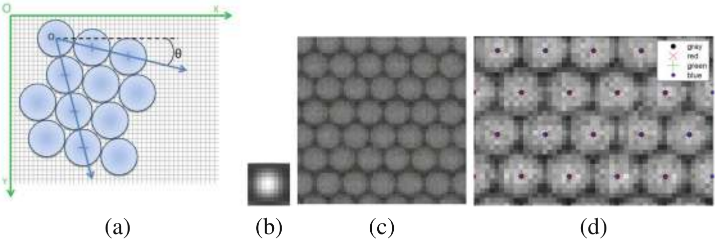


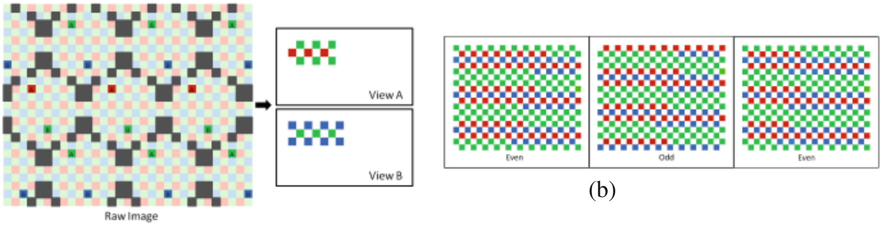
Fig. 2. (a) Microlenses projected on the sensor plane in a hexagonal arrangement. The green and blue axes represent the two CSs. There is a rotational offset θ and a translational offset $O - o$. (b) Mask used to locally estimate subimage center positions. (c) Lytro raw image of a white scene. (d) Estimated center positions. They coincide when estimated from one color channel only or from all the pixels in the raw image (gray).

There is a rotation offset between the sensor and the microlens plane. Also, the microlens diameter does not cover an integer number of pixels. Finally, the microlenses are arranged on a hexagonal grid to efficiently sample the space. Thus, in order to robustly estimate the microlens centers, we estimate the transformation between two coordinate systems (CS), the Cartesian CS given by the sensor pixels and K, the microlens center CS. K is defined as follows: the origin is the center of the topmost and leftmost microlens and the basis vectors are the two vectors from the origin to the adjacent microlens centers (see Fig.2-(a)). Formally, if $\mathbf{x}$ and $\mathbf{k}$ are respectively the coordinates on the sensor and microlens CSs, then, we estimate the system transformation matrix $\mathbf{T}$ and the offset vector between the origins $\mathbf{c}$ such that $\mathbf{x} = \mathbf{T}\mathbf{k} + \mathbf{c}$, and

$$\mathbf{T} = \begin{pmatrix} 1 & 1/2 \\ 0 & \sqrt{3}/2 \end{pmatrix} \begin{pmatrix} d_h & 0 \\ 0 & d_v \end{pmatrix} \begin{pmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{pmatrix} , \tag{1}$$

where the first matrix accounts for the orthogonal to hexagonal grid conversion due to the microlens arrangement, the second matrix deals with the vertical and horizontal shears and the third matrix is the rotation matrix. Thus, estimating the microlens model parameters $\{\mathbf{c}, d_h, d_v, \theta\}$ gives the microlenses center positions.

In practice, the subimage centers are computed from a *white image* depicted in Fig. 2-(c), that is an image taken through a white Lambertian diffuser. Actually, the subimage centers $\mathbf{x}_i$ of the i-th microlens in the raw image are computed as the local maximum positions of the convolution between the *white image* and the mask shown in Fig. 2-(b). Then, given $\mathbf{x}_i$ and the integer positions $\mathbf{k}_i$ in the K CS, the model parameters (and consequently $\mathbf{T}$ and $\mathbf{c}$) are estimated as the solution of a least square error problem from the equations $\mathbf{x}_i = \mathbf{T}\mathbf{k}_i + \mathbf{c}$. Thus, in this paper, the final center positions used in the demultiplexing step are the pixel positions given by $\mathbf{c}_i := \text{round}(\mathbf{T}\mathbf{k}_i + \mathbf{c})$. However, more advanced approaches



(a)

(b)

Fig. 3. (a) Demultiplexing. Pixels with the same relative position w.r.t. the subimage centers are stored in the same view. Only two views are illustrated for visualization. Color corresponds to sensor color on original Bayer pattern, and is carried over to assembled raw views. (b) Color patterns of three consecutive mosaicked views (even, odd and even positions of a line of the matrix of views) for a Lytro camera (~10 × 10 pix. per microlens). Color patterns from the views at even positions are very similar while the color pattern at the odd position is significantly different although there are horizontal color stripes too. White (empty) pixels are left to avoid aliasing.

can take into account the sub-pixel accuracy of the estimated centers and re-grid the data on integer spatial coordinates of the Cartesian CS. Fig. 2-(d) shows the subimage center estimation obtained with the method described above. Since the raw white image has a Bayer pattern, we have verified that the center positions estimated by considering only red, green or blue channel, or alternatively considering all color channels, are essentially the same. Indeed, demosaicking the raw *white image* does not create image cross-talk since the three color channels are the same for all pixels in the center of the subimages.

Reordering Pixels: In the following, we assume that the raw image has been divided pixel-wise by the white image. This division considerably corrects the vignetting [6] which is enough for our purposes. We refer to [7] for a precise vignetting modeling in plenoptic images. Now, in order to recover the different views, pixels are organized as illustrated in Fig. 3-(a). In order to preserve the pixel arrangement in the raw image (hexagonal pixel grid), empty spaces are left between pixels on the views as shown in Fig. 3-(b). Respecting the sampling grid avoids creating aliasing on the views. Notice that, since the raw image has not been demosaicked, the views inherit new color patterns. Because of the shift and rotation of the microlenses w.r.t. the sensor, the microlens centers (as well as other relative positions) do not always correspond to the same color. As a consequence, each view has its own color pattern (mainly horizontal monochrome lines in Lytro).

After demultiplexing, the views could be demosaicked without risking to fuse pixel information from different angular light rays. However, classic demosaicking algorithms are not well adapted to these new color patterns, specially on

[6] Light rays hitting the sensor at an oblique angle produce a weaker signal than other light rays.

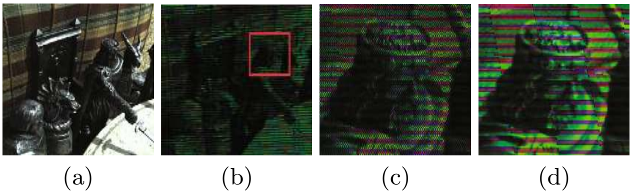


Fig. 4. (a) Lytro image (for visualization purposes). (b) One mosaicked view. (c) Zoomed red rectangle in view (b). (d) Same zoom with horizontal interpolation of empty (black) pixels, when possible. This simple interpolation does not create artifacts since all the pixels in a view contain same angular information.

high frequencies. For the sake of disparity estimation, we simply fill the empty pixels in a color chanel (white pixels in Fig. 3) when the neighboring pixels have the color information for this chanel (see Fig. 4). For example, if an empty pixel of the raw data has a green pixel on the right and on the left, then the empty pixel is filled with a green value by interpolation (1D Piecewise Cubic Hermite interpolation). Other empty pixels are left as such.

Differences with State-of-the-Art: The main difference with the demultiplexing method in [7] is the fact that in their method the raw data of a scene is demosaicked before being demultiplexed. This approach mixes information from different views and, as we will show in the next section, it has dramatic consequences on the disparity estimation. Besides, the method in [7] estimates the microlenses centers similarly to us but it does not force the center positions to be integer as we do in our optimization step. Instead, the raw image is interpolated to satisfy this constraint. Even if theoretically this solution should provide a more accurate LF, interpolating the raw data implies again mixing information from different views which creates image cross-talk artifacts. The method for estimating the center positions in [6] differs considerably from ours since the centers are found via local maxima estimation in the frequency domain. First, the raw image is demosaicked and converted to gray and the final center positions are the result of fitting the local estimation on a Delaunay triangular grid. Moreover, the second step to render the views is coupled with super-resolution providing views of size 1080 × 1080 (instead of 328 × 328, which is the number of microlenses).

The goal of this paper is to estimate accurately the disparity on plenoptic images, but we have observed that the processing needed before doing that is of foremost importance. So, even if the works in [7] and [6] are an important step forward for LF processing, we propose an alternative processing of the views which is better suited to subsequent disparity estimation.

4 Disparity Estimation

In this section, we present a new block-matching disparity estimation algorithm adapted to plenoptic images. We assume that a matrix of views is available (obtained as explained in the previous section) such that the views are horizontally and vertically rectified, i.e., satisfying the epipolar constraint. Therefore, given a pixel in a reference view, its corresponding pixels from the same row of the matrix are only shifted horizontally. Similar reasoning is valid for the vertical pixel shifts among views from the same column of the matrix. Furthermore, consecutive views have always the same baseline a (horizontally and vertically). As a consequence, for each point, its horizontal and vertical disparities with respect to nearest views are equal provided the point is not occluded. In other words, given a point in the reference view, the corresponding point in its consecutive right view is displaced horizontally by the same distance than the corresponding point in its consecutive bottom view is displaced vertically. By construction, the plenoptic camera provides a matrix of views with small baselines, which means that the possible occlusions are small. In fact, each point of the scene is seen from different points of views (even if it is occluded for some of them). Thus, the horizontal and vertical disparity equality is true for almost all the points of the scene. To the best of our knowledge, this particular property of plenoptic data has not been exploited before.

Since the available views have color patterns as in Fig. 3, we propose a block matching method in which only pixels in the block having the same color information are compared. We propose to use a similarity measure between blocks based on the ZSSD (Zero-Mean Sum of Squared Differences). Formally, let I^p be a reference view of the matrix of views and I^q be a view belonging to the same matrix row as I^p. Let $a_{p,q}$ be the respective baseline (a multiple of a). Then, the cost function between I^p and I^q at the center (x_0, y_0) of a block B_0 in I^p is defined as a function of the disparity d:

$$CF_0^{p,q}(d) = \frac{1}{\sum\limits_{(x,y)\in B_0} W(x,x',y)} \sum_{(x,y)\in B_0} W(x,x',y)\Big(I^p(x,y) - \overline{I_0^p} - I^q(x',y) + \overline{I_0^q}\Big)^2, \tag{2}$$

where $x' := x + a_{p,q}\, d$, $\overline{I_0^p}$ and $\overline{I_0^q}$ are the average values of I^p and I^q over the block centered at (x_0, y_0) and $(x_0 + a_{p,q}\, d, y_0)$ respectively and W is the window function

$$W(x,x',y) = G_0(x,y) \cdot S(x,x',y)\,,$$

where G_0 is a Gaussian function centered at (x_0, y_0) and supported in B_0 and S is the characteristic function controlling that only pixels in the block with same color information are compared in the cost function: $S(x,x',y) = 1$ if $I^p(x,y)$ and $I^q(x',y)$ have the same color information, and 0 otherwise. Note that the cost function is similarly defined when I^p and I^q are views from the same matrix column. In practice, we consider blocks of size 13×13.

Now, our algorithm takes advantage of the multitude of views given by the LF and estimates the disparity through all the rows and columns of the matrix. Let Θ be the set of index-view pairs such that the disparity can be computed horizontally or vertically w.r.t. the reference view I^p. In other words, Θ is the set of index-view pairs of the form (I^p, I^q), where I^q is from the same row or the same column as I^p. In fact, consecutive views are not considered in Θ

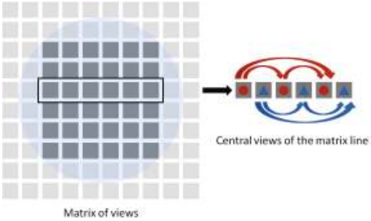


Fig. 5. On the left: LF (matrix of views). Views in the center get more radiance than views of the border of the matrix (pixels coming from the border of the microlenses). The 6×6 central views among the 10×10 are used. On the right: 6 central views from the same row of the matrix. Odd and even views have different color patterns between them (but very similar patterns between odd views and even views). This is represented with a red circle and a blue triangle. The index-view pairs in Θ corresponding to this matrix row are represented with the red and blue arrows.

since consecutive color patterns are essentially different because of the sampling period of sensor's Bayer pattern. Besides, views on the borders of the matrix are strongly degraded by the vignetting effect of the main lens. So, it is reasonable to only consider the 8×8 or 6×6 matrix of views placed in the center for the Lytro camera. Fig. 5 depicts the pairs of considered images for disparity estimation in a matrix row. Finally, given a reference view I^p, the disparity at (x_0, y_0) is given by

$$d(x_0, y_0) = \text{Med}_{(p,q)\in\Theta}\big\{\arg\min_{d} CF^{p,q}_{B_0}(d)\big\}\,, \tag{3}$$

where Med stands for the 1D median filter. This median filter is used to remove outliers that may appear on a disparity map computed for a single pair of views, specially in low-textured areas. It should be noted that through this median filtering, all the horizontally and vertically estimated disparities are considered to select a robust estimation of disparity which is possible thanks to the horizontal and vertical disparity equality mentioned beforehand.

Removing Outliers: Block-matching methods tend to provide noisy disparity maps when there is a matching ambiguity, e.g., for repeated structures in the images or on poorly textured areas. Inspired by the well-known cross-checking in binocular stereovision [20] (i.e., comparing left-to-right and right-to-left disparity maps), our method can also remove unreliable estimations comparing all possible estimations. Since a large amount of views are available from a LF, it is straightforward to rule out inconsistent disparities. More precisely, points (x_0, y_0) are considered unreliable if

$$\text{Std}_{(p,q)\in\Theta}\big\{\arg\min_{d} CF^{p,q}_{x_0,y_0}(d)\big\} > \varepsilon\,, \tag{4}$$

where Std stands for standard deviation and ε is the accuracy in pixels. In practice, we consider an accuracy of an eight of a pixel, $\varepsilon = \frac{1}{8}$.

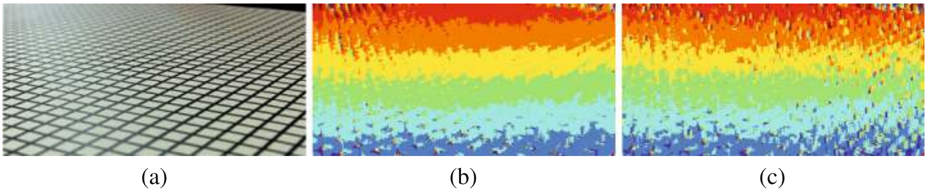

(a) (b) (c)

Fig. 6. (a) Lytro Image of the scene. (b) Disparity estimation without raw image demosaicking. (c) Disparity estimation with raw image demosaicking. The cost function is the same but the characteristic function is equal to one for all the points since the views are in full RGB. For the sake of accurate analysis no sub-pixel refinement has been performed. Errors due to image cross-talk artifacts are tremendous on disparity maps.

Sub-Pixel Disparity Estimation: By construction, the baseline between the views is small, specially between views with close positions in the matrix. So the disparity estimation for plenoptic images must achieve sub-pixel accuracy. Such precision can be achieved in two different ways: either by upsampling the views or by interpolating the cost function. Usually the first method achieves better accuracy but at a higher computational burden, unless GPU implementations are used [8]. For this reason, the second method (cost function interpolation) is usually used. However, it has been proved [19] that block-matching algorithms with a quadratic cost function as in Eq. (2) achieve the best trade-off between complexity and accuracy only by first upsampling the images by a factor of 2 and then interpolating the cost function. We follow this rule in our disparity estimation algorithm.

Differences with State-of-the-Art: The closest disparity estimation method for plenoptic images compared to ours is the method presented in [5] but there are several differences between both methods. First, our method properly demultiplexes the views before estimating the disparity, whereas the method in [5] considers full RGB views and proposes an antialiasing filter to cope with the weak prefilter in plenoptic type 2.0. Then, the energy defined in [5] (compare Eq. 3 of this paper with Eq. 3 in [5]) considers all the possible pairs of views even if in practice, for complexity reasons, only a subset of view pairs can be considered. In [5], no criteria is given to define such subset of view pairs while a reasonable subset is given with respect to the color pattern in our views. Finally, the proposed energy in [5] considers a regularization term in addition to the data term and the energy is minimized iteratively using conjugate gradients. In another state-of-the-art method, [22] combines spatial correspondence with defocus. More precisely, the algorithm uses the 4D EPI and estimates correspondence cues by computing angular variance, and defocus cues by computing spatial variance after angular integration. Both cues are combined in an MRF global optimization process. Nevertheless, their disparity estimation method does not take care of the demultiplexing step accurately. Their algorithm not only demosaicks the raw image, but it stores it using JPEG compression. So, the resulting LF is affected by image cross-talk artifacts and compression artifacts. In next section, we shall compare our results with this method. Unfortunately, a qualitative comparison with [5] is not possible since the authors work with different data: mosaicked views from a focused or type 2.0 plenoptic camera.

5 Experimental Results

In this section we show the results obtained with our algorithm. First of all, we have compared the disparity maps obtained with and without demosaicking the raw image. Intuitively one can think that demosaicking the raw image will get better results since more information is available on the views. However this intuition is rejected in practice (see for instance Fig. 6). Therefore, we claim that accurate disparity estimation should consider only the raw data on the views. Unfortunately, experimental evaluation with available benchmarks with ground-truth [24] as in [13] is not possible because all LF in the benchmark are already demosaicked.

Fig. 7 compares our disparity maps from Lytro using [2] and the disparity map from [22] using the code provided by the authors and the corresponding microlenses center positions for each experiment. The algorithms have been tested with images from [22] and images obtained with our Lytro camera. The poor results from [22] with our data show a strong sensitivity to parameters of their algorithm. Also, their algorithm demosaicks and compresses (JPEG) the raw image before depth is estimated. On the other hand, Lytro disparity maps are more robust but they are strongly quantized which may not be sufficiently accurate for some applications. All in all, our method has been tested on a large number of images from Lytro with different conditions and it provides robust and accurate results compared to state-of-the-art disparity estimation method for plenoptic images.

Obviously, other approaches could be considered for disparity estimation. For instance, our cost function can be regarded as the data term in a global energy minimization approach as in [25]. However, for the sake of computational speed we have preferred a local method. Specially, because a multitude of disparity estimations can be performed at each pixel. Moreover, other approaches using EPI's as in [24] could be used but we have observed that EPI's from Lytro are highly noisy and only disparities on object edges are reliable (EPI from Lytro is only ~ 10 pixels width).

In this paper we propose to not perform demosaicking on the raw image to avoid artifacts but full RGB images are needed for some applications (i.e., refocusing). In that case we suggest to recover the lacking colors by bringing the color information from all the corresponding points in all views using the estimated disparity information as in [21]. Indeed, one point in the reference view seen with one color channel is seen in the other views with another color. Fig. 8 shows disparity-guided demosaicking results. We show that our approach avoids color artifacts compared with the method in [22] that demosaicks raw images. So,

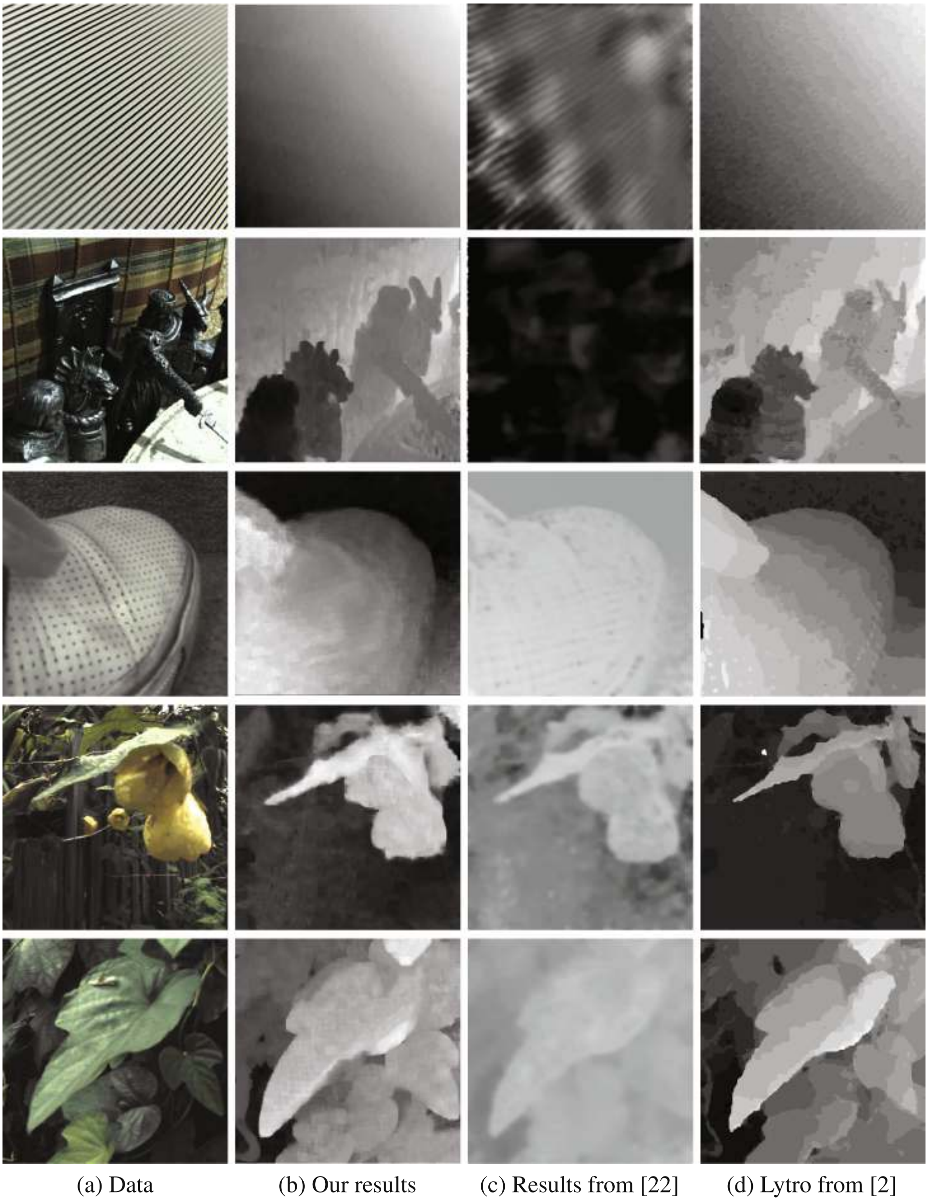


Fig. 7. (a) Original data. The three last images are published in [22]. (b) Our disparity map results. (c) Results from [22]. The authors have found a good set of parameters for their data but we have found poor results using their algorithm with our data. (d) Depth map used by Lytro, obtained with a third party toolbox [2].

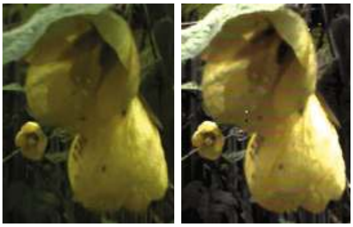


Fig. 8. Comparison of RGB views. Left: Our result. Right: Result of demosaicking the raw data as in [22]. Besides of a different dynamic range certainly due to a different color balance, notice the reddish and greenish bands on the right flower (best seen on PDF).

our demultiplexing mosaicked data strategy not only avoids artifacts on disparity maps but also on full RGB view rendering.

It shall be pointed out that we assume the Lytro camera to be a plenoptic type 1.0. Although not much is officially available about its internal structure, our observation of the captured data and the study in [11] support this assumption. However, the assumption on the camera type only changes the pixel reordering in the demultiplexing step, and the proposed method can be easily generalized to the case of plenoptic type 2.0.

Finally, even if our method only considers central views of the matrix of views, we have observed slightly bigger errors on the borders of the image. Pushing further the correction of vignetting and of other chromatic aberrations could be profitable to accurate disparity estimation. This is one of our perspectives for future work.

6 Conclusion

Plenoptic cameras are promising tools to expand the capabilities of conventional cameras, for they capture the 4D LF of a scene. However, specific image processing algorithms should be developed to make the most of this new technology. There has been tremendous effort on disparity estimation for binocular stereovision [20], but very little has been done for the case of plenoptic data. In this paper, we have addressed the disparity estimation problem in plenoptic data and we have seen that it should be studied together with demultiplexing. In fact, the proposed demultiplexing step on mosaicked data is a simple pre-processing that has clear benefits for disparity estimation and full RGB view rendering since they do not suffer from view cross-talk artifacts.

References

1. http://code.behnam.es/python-lfp-reader/
2. http://optics.miloush.net/lytro/

3. Adelson, E., Wang, J.: Single lens stereo with a plenoptic camera. TPAMI **14**(2), 99–106 (1992)
4. Bishop, T.E., Favaro, P.: Full-Resolution Depth Map Estimation from an Aliased Plenoptic Light Field. In: Kimmel, R., Klette, R., Sugimoto, A. (eds.) ACCV 2010, Part II. LNCS, vol. 6493, pp. 186–200. Springer, Heidelberg (2011)
5. Bishop, T.E., Favaro, P.: The light field camera: Extended depth of field, aliasing, and superresolution. TPAMI **34**(5), 972–986 (2012)
6. Cho, D., Lee, M., Kim, S., Tai, Y.W.: Modeling the calibration pipeline of the lytro camera for high quality light-field image reconstruction. In: ICCV (2013)
7. Dansereau, D.G., Pizarro, O., Williams, S.B.: Decoding, calibration and rectification for lenselet-based plenoptic cameras. In: CVPR (2013)
8. Drazic, V., Sabater, N.: A precise real-time stereo algorithm. In: ACM Conf. on Image and Vision Computing New Zealand, pp. 138–143 (2012)
9. Fiss, J., Curless, B., Szeliski, R.: Refocusing plenoptic images using depth-adaptive splatting. In: ICCP (2014)
10. Georgiev, T., Chunev, G., Lumsdaine, A.: Superresolution with the focused plenoptic camera. In: SPIE Electronic Imaging (2011)
11. Georgiev, T., Yu, Z., Lumsdaine, A., Goma, S.: Lytro camera technology: theory, algorithms, performance analysis. In: SPIE Electronic Imaging, pp. 86671J–86671J (2013)
12. Gortler, S.J., Grzeszczuk, R., Szeliski, R., Cohen, M.F.: The lumigraph. In: Conf. on Computer Graphics and Interactive Techniques (1996)
13. Heber, S., Ranftl, R., Pock, T.: Variational Shape from Light Field. In: Heyden, A., Kahl, F., Olsson, C., Oskarsson, M., Tai, X.-C. (eds.) EMMCVPR 2013. LNCS, vol. 8081, pp. 66–79. Springer, Heidelberg (2013)
14. Kim, C., Zimmer, H., Pritch, Y., Sorkine-Hornung, A., Gross, M.: Scene reconstruction from high spatio-angular resolution light fields. ACM Trans. Graph. **32**(4), 73 (2013)
15. Lippmann, G.: Epreuves reversibles donnant la sensation du relief. J. Phys. Theor. Appl. **7**(1), 821–825 (1908)
16. Lumsdaine, A., Georgiev, T.: The focused plenoptic camera. In: ICCP (2009)
17. Ng, R.: Digital light field photography. Ph.D. thesis, Stanford University (2006)
18. Perez, F., Perez, A., Rodriguez, M., Magdaleno, E.: Fourier slice super-resolution in plenoptic cameras. In: ICCP (2012)
19. Sabater, N., Morel, J.M., Almansa, A.: How accurate can block matches be in stereo vision? SIAM Journal on Imaging Sciences **4**(1), 472–500 (2011)
20. Scharstein, D., Szeliski, R.: A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. IJCV **47**(1–3), 7–42 (2002)
21. Seifi, M., Sabater, N., Drazic, V., Perez, P.: Disparity-guided demosaicing of light-field images. In: ICIP (2014)
22. Tao, M., Hadap, S., Malik, J., Ramamoorthi, R.: Depth from combining defocus and correspondence using light-field cameras. In: ICCV (2013)
23. Tulyakov, S., Lee, T., H., H.: Quadratic formulation of disparity estimation problem for light-field camera. In: ICIP (2013)
24. Wanner, S., Goldluecke, B.: Variational light field analysis for disparity estimation and super-resolution. TPAMI (2014) (to appear)
25. Yu, Z., Guo, X., Ling, H., Lumsdaine, A., Yu, J.: Line assisted light field triangulation and stereo matching. In: ICCV (2013)
26. Yu, Z., Yu, J., Lumsdaine, A., Georgiev, T.: An analysis of color demosaicing in plenoptic cameras. In: CVPR (2012)