HAL
open science

# White Lenslet Image Guided Demosaicing for Plenoptic Cameras

Pierre David, Mikaël Le Pendu, Christine Guillemot

## To cite this version:

Pierre David, Mikaël Le Pendu, Christine Guillemot. White Lenslet Image Guided Demosaicing for Plenoptic Cameras. MMSP 2017 - IEEE 19th International Workshop on Multimedia Signal Processing, Oct 2017, Luton, United Kingdom. hal-01590345

**HAL Id: hal-01590345**

**https://hal.science/hal-01590345v1**

Submitted on 19 Sep 2017

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# White Lenslet Image Guided Demosaicing for Plenoptic Cameras

Pierre David, Mikaël Le Pendu, Christine Guillemot
INRIA
Campus de Beaulieu, 35042 Rennes Cedex, France
firstname.lastname@inria.fr

***Abstract*—Most modern cameras use a color filter array on their sensor in order to capture color images. This array is composed of red, green and blue filters and so, each pixel on the sensor lacks two color channels which can be retrieved by a process called demosaicing. In this paper, we propose a new demosaicing method for plenoptic cameras. This type of cameras has become a growing trend and their captured raw images have a particular lenslet structure which must be taken into account to retrieve the sub-aperture images which compose the light field. First, we analyze and describe the flaws of the state-of-the-art light field decoding pipeline. To better identify the different sources of artifacts, our analysis is performed by generating ideal lenslet images from synthetic light fields and use them as input of the decoding pipeline. Then, we detail a new method of demosaicing based on the provided white lenslet images serving as guide. Furthermore, we show that this kind of guided interpolation can be useful on other steps of the decoding pipeline. Finally, the quality of the resulting sub-aperture images is assessed for both synthetic and real light fields using visual comparisons as well as objective metrics.**

## I. Introduction

The recent developement of plenoptic cameras enables the instantaneous capture of light fields with both spatial and angular information of the scene, as opposed to traditional images with only 2D spatial coordinates. Among plenoptic cameras, two optical designs can be distinguished, both based on the addition of a lenslet array between the sensor and the main lens. The first model, originally described by Lippmann [1] and modernized by Ng et al. [2], was popularized by its implemenation in the commercially available Lytro cameras. In this design (also called "plenoptic 1.0"), the main lens focuses the subject on the lenslet array which separates the converging rays on the sensor (see (a) in Figure 1). The second designed, called "focused plenoptic cameras" or "plenoptic 2.0", was described in [3]. In this configuration, the image plane of the main lens is the object plane of the lenslet array. So, the lenslet images on the sensor are in focus (see (b) in Figure 1). This design is used in Raytrix cameras to control the tradeoff between spatial and angular resolution.

Thanks to the lenslet array, we are able to know the contribution of each ray in every direction. However, the image captured by a plenoptic camera is not directly interpretable and needs to be decoded. The main focus of the article is the light field demosaicing for "plenoptic 1.0 cameras". Three approaches have been studied in the litterature. First, we can directly demosaic the raw sensor image (also called lenslet image). The authors in [4] use a 4D kernel regression method exploiting the lenslet structure. However, it performs independent interpolation of the color channels, which results in a loss of detail compared to a classical 2D demosaicing such as [5]. The second approach is demosaicing the views after having demultiplexed them. The authors in [6] chose a dictionary learning based method to retrieve the missing colors, while [7] used a disparity map to find the missing colors in other decoded views. This second type of demosaicing presents a major drawback: every interpolation before extracting the views must be done in a nearest neighbor way. The resulting views have therefore some strong aliasing. Finally, the third approach consists in demosaicing the focal stack as in [8]. Whereas it is easy to compute a focal stack from views, it is much harder to retrieve views from a focal stack [9]. For these reasons, we believe that the first approach, taking into account the lenslet structure of the images, could lead to a better decoding of the light field.

For plenoptic images, the demosaicing step is just a part of a whole decoding pipeline which extracts views from a lenslet image. So, when proposing a new demosaicing method, it seems relevant to analyze the full decoding pipeline. There are several pipelines [10] [11] [4]. For our paper, we choose the pipeline described by Dansereau et al. in [11] as it is widely used, it has been chosen by JPEG-Pleno and was imposed for ICME 2016 Grand Challenge on Light-Field Image Compression. Lastly, the code in [11] is available. We choose to evaluate our demosaicing method within Dansereau pipeline but the proposed method remains valid for other pipelines as long as the demosaicing step is done on the lenslet image. The contributions of the paper are as follows: first we

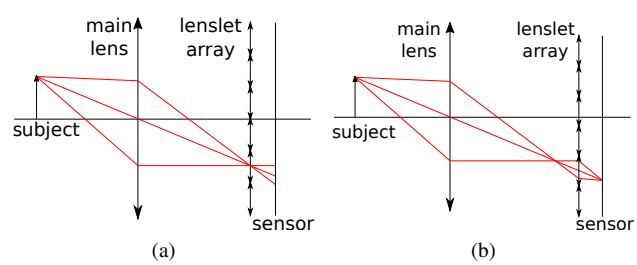



Fig. 1. Optical configurations for plenoptic cameras: (a) plenoptic 1.0, (b) plenoptic 2.0

analyze the flaws of the pipeline and identify the link between the lenslet structure of the image and bad demosaicing, then we propose a demosaicing method guided by a white lenslet image, finally we propose an application of the same principle for the other interpolations in the decoding pipeline.

## II. Description of the decoding pipeline

This section explains how the pipeline [11] retrieves the light field sub-aperture images (or views) from the raw image of a plenoptic camera and the associated metadata. This algorithm is called "demultiplexing" or "decoding". It proceeds by the following steps (illustrated in Figure 2):

*Devignetting:* The optical design of a plenoptic camera (see Figure 1) creates images with a particular structure. These images have the same pattern as the lenslet array: each lenslet creates a small image on the sensor. This image suffers from heavy vignetting which means that the brightness at the periphery is reduced compared to the lenslet center. The devignetting step then consists in dividing the sensor data by a calibration image (or white lenslet image) that is chosen, depending on the zoom and focus settings, among a set of RAW pictures of a uniformly illuminated lambertian white surface. These calibration images exhibit the vignetting pattern of the lenslet array. Note that they are also used in the calibration phase to determine the positions of the lenslet centers on the sensor.

*Demosaicing:* After correcting the vignetting of the raw image, the pipeline proceeds by demosaicing the lenslet images. It uses the gradient corrected interpolation method proposed in [5]. This method computes the values of the missing RGB channels by bilinearly interpolating the neighbor values and correcting the output with the computed gradient of the present channel. The image is convolved with various kernels depending on the color filters of the pixels. The kernels are given in Figure 3. The coefficients $\alpha$, $\beta$ and $\gamma$ represent the strength of the gradient correction. They are chosen to be respectively equal to 1/2, 5/8 and 3/4 which are the integer multiples of small powers of 1/2 which minimize the mean square error of a data set.

*Aligning the sensor with the lenslets:* Knowing the positions of the lenslet centers on the sensor (determined in the calibration phase), rotation, translation and scaling are applied to the image to compensate for the misalignements between the lenslet array and the pixel grid (illustrated in Figure 4).

*Slicing the lenslet image:* The next step is to slice the lenslet image i.e. demultiplex the lenslet image to extract the views. Now that the diameter D of each lenslet image is scaled to be an integer, we just have to pick a pixel every D pixels to form a view (see Figure 5).

*Resampling the views:* As the lenslet array is hexagonal, the views suffer from an hexagonal aliasing (see the two missing pixels in black in the extracted view in Figure 5). The pipeline removes this aliasing by resampling the views into a rectangular grid.

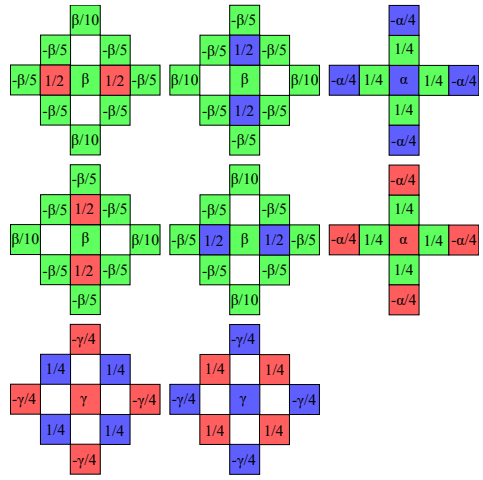



Fig. 3. Kernels in [5]

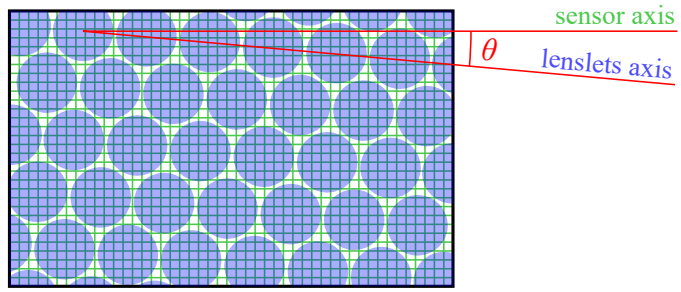



Fig. 4. Misalignment between the sensor and the lenslets

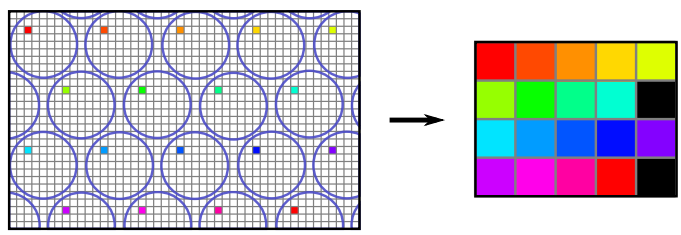


Fig. 5. Extraction of a view from the lenslet images

## III. Flaws of the pipeline and some improvements

### A. Synthetic lenslet image generation

In order to better analyze the pipeline, we developed a method to make a synthetic lenslet image from a synthetic light field [12] (see Figure 6). We first remove some peripheral views (in the corners), in order to have circular angular patches (as the captured plenoptic light field are). Then we put the different angular patches near one another along a hexagonal grid. This operation is the exact contrary of the slicing or demultiplexing step in the decoding pipeline. We also add vignetting to the lenslets and generate a white image that holds the exact vignetting profile of the lenslet image (dividing the lenslet image by the white image will give us perfect devignetting). Note that the diameter of the generated lenslet image is only 9 (as we originally have 9x9 synthetic views) and that for a real plenoptic camera, the diameter is likely

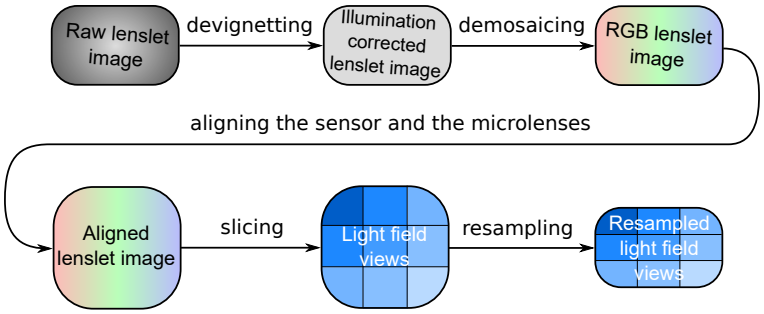



Fig. 2. Dansereau's pipeline

to be higher (11 for the Lytro 1, 15 for the Lytro Illum for example).

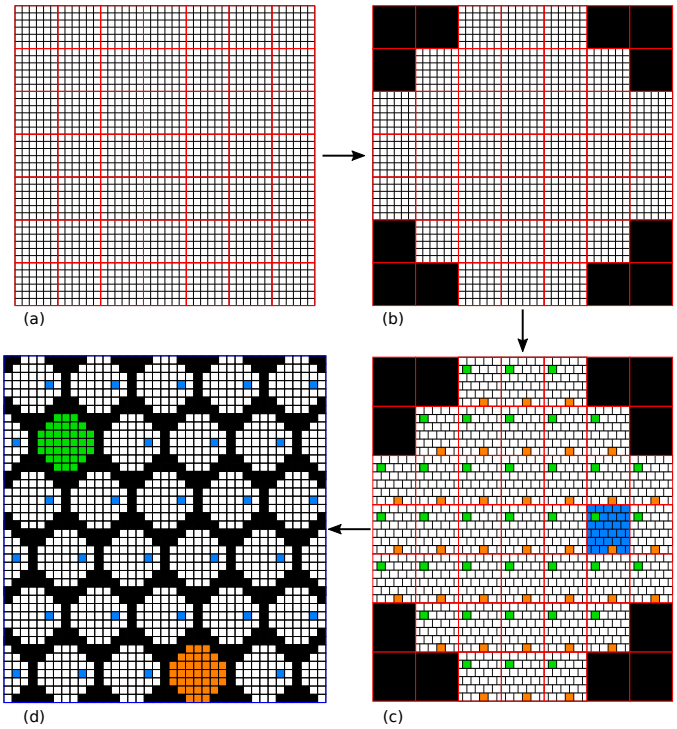



Fig. 6. From a synthetic light field to a lenslet image: **(a)** is a 7x7 synthetic light field, the red squares are representing the views and the black squares the pixels of a view; in **(b)** the views in the corners are removed; in **(c)** the light field is resampled in order for each view to have an hexagonal grid of pixels; **(d)** is the final lenslet image.

To test how well the gradient corrected interpolation demosaicing method works on lenslet images, we use synthetic lenslet images and we remove two channels from each pixels according to a Bayer pattern to generate ground truth demosaiced lenslet images. Figure 7 shows the ground truth lenslet image and (a) in Figure 8 shows the same image demosaiced with Malvar's method [5]. We can see that this method is not ideal as it creates color fringes on the borders of the lenslets. The fringes are due to the pixels that are out of the lenslet. As they get no signal, their value remains the same (if we ignore noise) between the raw image and a white image. So when we divide the raw image by the white image, these pixel values

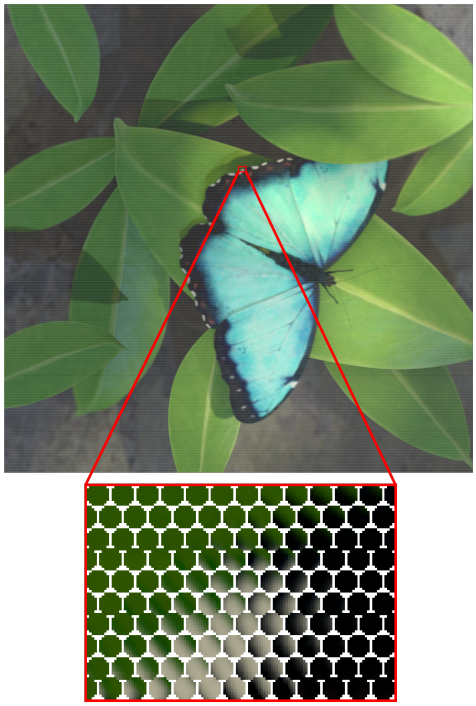


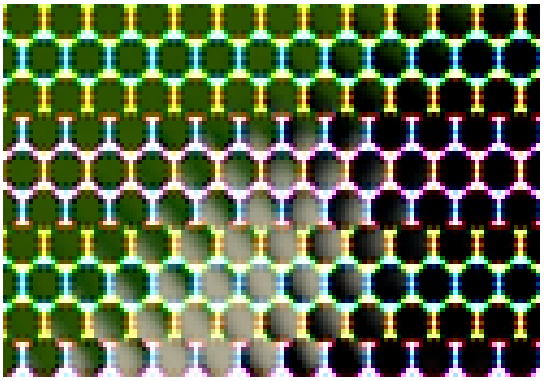
Fig. 7. Lenslet image of Butterfly [12]

are 1. When interpolating the missing channels for a pixel at the border of a lenslet, the bilinear interpolation and the gradient corrections are both disturbed by these white pixels.

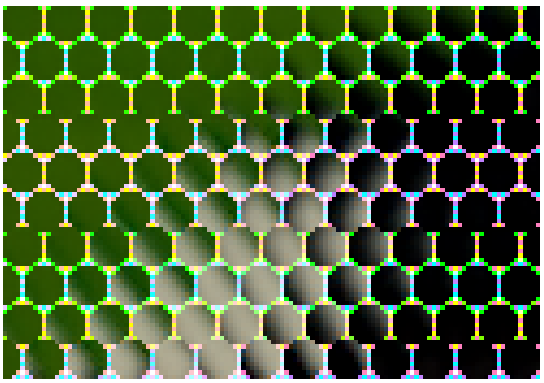
### B. Proposed demosaicing method

Following this observation, we propose to discard the pixels that are out of the lenslets from the demosaicing step. To do this, we adapted the gradient corrected interpolation method by weighting the bilinear interpolation and gradient correction:

- First with a mask $b$ : we do not want to interpolate data from different lenslets as this creates crosstalk artifacts. Knowing the lenslet grid parameters, we know the position of the lenslets centers and we can identify the pixels which belong to the same lenslets.
- Then with a white image $c$ : we have full confidence in the pixels that have the maximum values on the white image whereas we have less confidence in the pixels that are darker on the white image, as they are noisier.

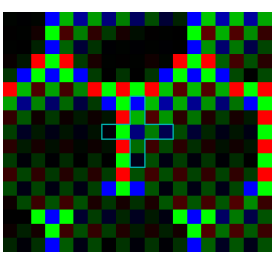


(a) PSNR = 19.17

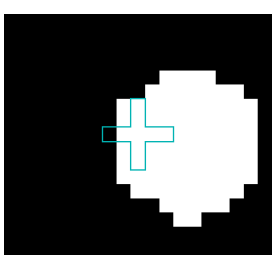


(b) PSNR = 42.94

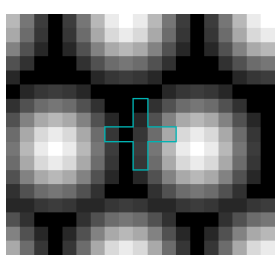
Fig. 8. Detail of a demosaiced synthetic lenslet image: (a) uses the gradient corrected interpolation method [5] and (b) uses our method. The pixels out of the lenslets are ignored in the PSNR computation.



(a)



(b)



(c)

Fig. 9. (a) is the raw image we want to demosaic, (b) is a mask which holds every pixel belonging to the same lenslet, (c) . We use the combination of (b) and (c) as additional weights in the bilinear interpolation and gradient correction described in [5].

Let us take the example shown in Figure 9 where we want to compute the green value of the blue pixel located in the center of the cross (Image **(a)**). Let $(i, j)$ be the coordinates of the blue pixel, $G_{ij}$ the green value to compute, $B_{ij}$ the measured blue value, $(b_{ij})$ the mask, $(c_{ij})$ the white image and $(d_{ij})$ the coefficients of the kernels mentioned in [5] (see Figure 3).

The green value is computed as follows:

$$G_{ij} = \sum_{(k,l)\in N} w_{kl}^{bil} G_{kl} + \alpha\Big(\sum_{\substack{(k,l)\in M\\ d_{kl}>0}} w_{kl}^{grad+} B_{kl} - \sum_{\substack{(k,l)\in M\\ d_{kl}<0}} w_{kl}^{grad-} B_{kl}\Big) \quad (1)$$

where:

$$N = \{(i-1,j),(i+1,j),(i,j-1),(i,j+1)\} \quad (2)$$
$$M = \{(i,j),(i-2,j),(i+2,j),(i,j-2),(i,j+2)\}$$

and:

$$w_{kl}^{bil} = \frac{e_{kl}}{\sum_{(m,n)\in N} e_{kl}}$$
$$w_{kl}^{grad+} = \frac{f_{kl}}{\sum_{\substack{(m,n)\in M\\ d_{mn}>0}} f_{mn}} \quad , \quad w_{kl}^{grad-} = \frac{f_{kl}}{\sum_{\substack{(m,n)\in M\\ d_{mn}<0}} f_{mn}} \quad (3)$$
$$e_{kl} = b_{kl} \times c_{kl}$$
$$f_{kl} = b_{kl} \times c_{kl} \times d_{kl}$$

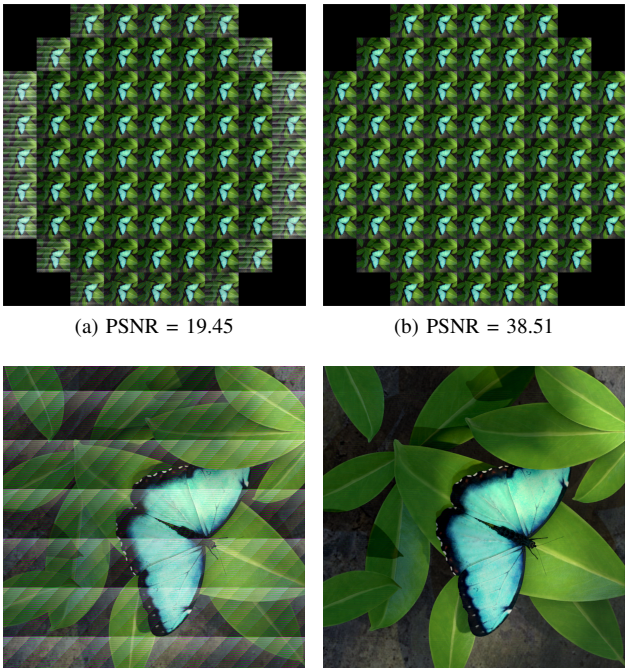
Every other missing values of each pixel is computed in the same way, using $(b_{ij})$ and $(c_{ij})$ as weights (respectively illustrated by (b) and (c) in Figure 9). Figure 8 (a) and (b) show the results respectively with and without the contribution of the weights $(b_{ij})$ and $(c_{ij})$. In this experiment, the PSNR can be computed since the ground truth lenslet image is known. Note that the pixels out of the lenslets are ignored in the PSNR computation since they do not hold any signal and they are not necessary for the view reconstruction. In these conditions, a large PSNR gain of 23.77 dB is observed with our method.

### C. Proposed alignment method

Like the demosaicing step, applying a rotation, translation and scaling involves some interpolations which mix pixels with signal with pixels without signal. Moreover, it can also mix pixels from different lenslets, creating crosstalk artifacts. This leads to a strong deterioration of the lenslet borders.

To assess the deterioration induced by this scaled rotation, we first generate a synthetic lenslet image where the lenslets are not perfectly aligned with the pixels. The grid defined by the lenslets is slightly tilted with respect to the pixel grid (the angle is approximately the same as the Lytro 1, 0.05°). Then, as it is done in the pipeline, we demosaic and apply an affine transformation to the lenslet image in order for the lenslets to be aligned with the pixels.

As the slicing step is just a reorganization of the data to have light field views, it introduces no further error. So, we apply



(a) PSNR = 19.45 (b) PSNR = 38.51

(c) PSNR = 15.29 (d) PSNR = 36.13

Fig. 10. Extracted views from a synthetic lenslet image: (a) uses a simple rotation to align the sensor and the lenslets, (b) uses our method. Images (c) and (d) show the same view (2,2) respectively taken from (a) and (b).

the slicing step to better visualize the impact of the previous rotation and to compare it with the ground truth light field.

We propose a new way to compute the affine transformation which is quite similar to the demosaicing method: we compute the new aligned lenslet image with a bilinear interpolation weighted by the white image and a mask which eliminates the pixels which do not belong to the same lenslets.

As we can see in (a) and (c) in Figure 10, the peripheral views are greatly degraded by the affine transformation. On the other hand, when we use our white image guided interpolation method, we see that there are no line anymore on the views and that the global PSNR has doubled, going from 19.45 to 38.51 dB (see (b) and (c) in Figure 10).

If we use a perfectly demosaiced lenslet image (using the ground truth lenslet image), Dansereau's alignment method gives us a global PSNR of 23.12 dB whereas our method gives a PSNR of 39.84 dB. So, our interpolation method is shown to significantly improve the decoding pipeline for the synthetic lenslet images.

### D. Real lenslet images

We now test our method on real lenslet images. We use Lytro 1 [13] and Illum [14] images. For a real lenslet image, the pixels at the border of the lenslets are not enough penalized when using the pixel values $(c_{ij})$ of the white image as coefficients. So we use the pixel values of the white image raised to the power of ten instead, which reduces the weights of the pixels at the border of the lenslets relatively to those at the centers. In Figures 11 and 12, we can see that the peripheral views are sharper with the proposed demosaicing and alignment methods. Our proposed methods are also efficient to remove some of the crosstalk artifacts. Indeed, we can see on the bottle in the foreground in Figure 11(a) and on the rose petals in Figure 12(a) that there are two views mixed together. In Figures 11(b) and 12(b), this ghost effect disappears. Finally, the colors of the peripheral views are closer to the colors of the central view, as pixels at the center of the lenslets have more weights than those at the border which are grayer in real lenslet images.

More results and an available Matlab code can be found on the web page https://www.irisa.fr/temics/demos/lightField/Demosaicing/LensletDemosaicing.html

## IV. Conclusion

We developed a method to create lenslet images from synthetic light fields. With this synthetic lenslet images, we were able to analyze the light field decoding pipeline chosen by JPEG-Pleno. We particularly evaluated the demosaicing and alignment steps. We noticed that the used methods were not adapted to the lenslet images and that it created crosstalk artifacts and color patterns. In order to improve these steps, we developed methods of demosaicing and alignment guided by a white lenslet image. These proposed methods showed improvements on the synthetic and on the real lenslet images. The results are shown to reduce cross talks artifacts and produce better colors on the decoded light fields views.

## Acknowledgment

This project has been supported by the EU H2020 Research and Innovation Programme under grant agreement No 694122 (ERC advanced grant CLIM).

## References

[1] G. Lippmann, "Epreuves reversibles donnant la sensation du relief," *J. Phys. Theor. Appl.*, vol. 7, no. 1, pp. 821–825, 1908.

[2] R. Ng, M. Levoy, M. Brédif, G. Duval, M. Horowitz, and P. Hanrahan, "Light field photography with a hand-held plenoptic camera."

[3] A. Lumsdaine and T. Georgiev, "The focused plenoptic camera," in *Computational Photography (ICCP), 2009 IEEE International Conference on*. IEEE, 2009, pp. 1–8.

[4] S. Xu, Z.-L. Zhou, and N. Devaney, "Multi-view image restoration from plenoptic raw images," in *Asian Conference on Computer Vision*. Springer, 2014, pp. 3–15.

[5] H. S. Malvar, L.-w. He, and R. Cutler, "High-quality linear interpolation for demosaicing of bayer-patterned color images," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 3. IEEE, 2004, pp. iii–485.

[6] X. Huang and O. Cossairt, "Dictionary learning based color demosaicing for plenoptic cameras," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2014, pp. 449–454.

[7] M. Seifi, N. Sabater, V. Drazic, and P. Perez, "Disparity-guided demosaicking of light field images," in *Image Processing (ICIP), 2014 IEEE International Conference on*. IEEE, 2014, pp. 5482–5486.

[8] Z. Yu, J. Yu, A. Lumsdaine, and T. Georgiev, "An analysis of color demosaicing in plenoptic cameras," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 901–908.

(a) (b)

Fig. 11. Extracted peripheral view from a real lenslet image taken with a Lytro 1: (a) is decoded as described in [11], (b) is decoded with our proposed demosaicing and alignment methods.

(a) (b)

Fig. 12. Extracted peripheral view from a real lenslet image taken with a Lytro Illum: (a) is decoded as described in [11], (b) is decoded with our proposed demosaicing and alignment methods.


[9] A. Levin and F. Durand, "Linear view synthesis using a dimensionality gap light field prior," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE, 2010, pp. 1831–1838.

[10] D. Cho, M. Lee, S. Kim, and Y.-W. Tai, "Modeling the calibration pipeline of the lytro camera for high quality light-field image reconstruction," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 3280–3287.

[11] D. G. Dansereau, O. Pizarro, and S. B. Williams, "Decoding, calibration and rectification for lenselet-based plenoptic cameras," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2013, pp. 1027–1034.

[12] S. Wanner, S. Meister, and B. Goldluecke, "Datasets and benchmarks for densely sampled 4d light fields." in *VMV*. Citeseer, 2013, pp. 225–226.

[13] "Lytro 1 image dataset," http://www.irisa.fr/temics/demos/lightField/index.html, accessed: 05-03-2017.

[14] "Lytro Illum image dataset," http://www.irisa.fr/temics/demos/IllumDatasetLF/index.html, accessed: 05-03-2017.