Capturing the plenoptic function in a swipe

Michael Lawson, Mike Brookes, and Pier Luigi Dragotti

Imperial College, London, UK

ABSTRACT

Blur in images, caused by camera motion, is typically thought of as a problem. The approach described in this paper shows instead that it is possible to use the blur caused by the integration of light rays at different positions along a moving camera trajectory to extract information about the light rays present within the scene. Retrieving the light rays of a scene from different viewpoints is equivalent to retrieving the plenoptic function of the scene. In this paper, we focus on a specific case in which the blurred image of a scene, containing a flat plane with a texture signal that is a sum of sine waves, is analysed to recreate the plenoptic function. The image is captured by a single lens camera with shutter open, moving in a straight line between two points, resulting in a swiped image. It is shown that finite rate of innovation sampling theory can be used to recover the scene geometry and therefore the epipolar plane image from the single swiped image. This epipolar plane image can be used to generate unblurred images for a given camera location.

Keywords: Annihilating filter method, Piecewise sinusoidal signals, Plenoptic function, Plenoptic sampling, Sampling methods

1. INTRODUCTION

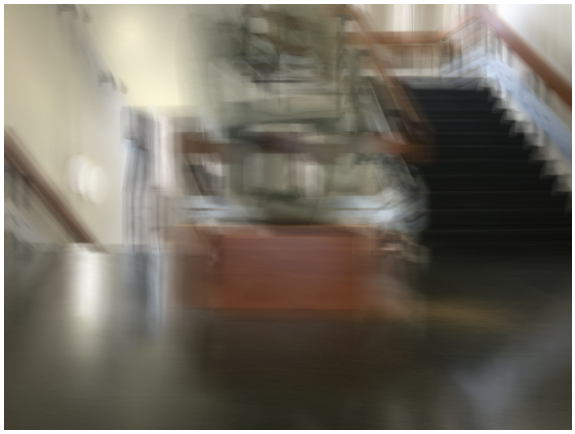

Figure 1. A blurred image of a scene. The motion blur is due to the camera moving in a straight line with the shutter open. In this paper, this type of images are called 'swiped images' and the aim is to to retrieve the entire EPI from them.

The plenoptic function, first introduced by Adelson and Bergen in[1], describes a scene in terms of light rays observed by a camera at an arbitrary location. The full plenoptic function is a 7 dimensional function, with the intensity of the light rays given by $I(x, y, z, v, w, \lambda, \tau)$, where (x, y, z) is the camera location, (v, w) specifies the direction of the light ray, λ is the wavelength, and τ is time[1–6]. Figure 1 shows the blur caused by the integration of light rays at different positions along a moving camera trajectory, which is equivalent to integrating the light rays in a plenoptic function. The full plenoptic function is complicated to analyse due to its high dimensionality. A simplification to the full plenoptic function is the light field model[7], in which time and wavelength are fixed. The scene or camera view can be contained within a bounding box or convex hull, limiting the light field plenoptic

Further author information: (Send correspondence to Michael Lawson)
Michael Lawson: E-mail: M.lawson14@imperial.ac.uk

function to 4 parameters $I(u, v, s, t)$. Each ray of light passes through two parallel planes. The plane with the coordinates (u, v) is the focal plane, and the plane with the coordinates (s, t) is the camera plane.

A further simplification to the plenoptic function can be made by constraining the camera movement to a horizontal line perpendicular to the horizontal axis; this leads to the Epipolar Plane Image (EPI)[8]. In the EPI model, wavelength and time are also omitted, which results in a 3 parameter function $I(x, v, w)$. In the EPI model it is assumed that the surfaces in the scene are Lambertian, therefore the light intensity does not vary with viewing angle[5]. The plenoptic function is important as it represents an entire scene without the limitations of a single image taken from a single viewpoint.

The plenoptic function is a powerful tool to model and describe many computer vision and image processing problems that deal with the processing of multi-view images. For example the problem of synthesising a novel view of a real-word scene given a set of images from different viewpoints, known as Image Based Rendering (IBR)[9–15], can be seen as the problem of sampling and interpolating the plenoptic function. In order to perform this sampling and interpolation effectively many researchers have studied the spectral properties of the plenoptic function (e.g.[2,3,5]), or have used a layer-based model for the scene of interest[15–18].

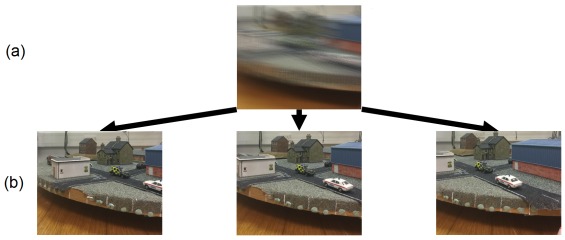

Figure 2. (a) shows a swiped image created by opening the camera shutter as the camera moves. (b) shows the images reconstructed using the plenoptic function recovered from (a).

Obtaining the plenoptic function of a scene is difficult in practice and requires either an array of cameras or a multi lens camera[19]. Both of these options significantly increase the cost, and so are rarely used in consumer electronics. The outcome of the technique described in this paper is to capture the plenoptic function (i.e. the EPI) with a single image taken with a conventional consumer camera or a smartphone. The entire information of the EPI, within the camera swipe start and end locations, is encapsulated by capturing a single swiped image using a camera moving along a straight line with the shutter open, with an example swiped image being shown in Figure 1. The goal is, given the swiped image shown in Figure 2(a), to generate new images, shown in Figure 2(b), that match images of the scene from different viewpoints.

This novel technique is different from the traditional deblurring methods that are employed in many consumer electronics, as the exact reconstructed pictures of the scene can be created, within the range of camera movement. This contrasts with the non-exact sharp image from a single viewpoint that results from a mathematical model used for the well-studied process of deblurring[20,21]. Traditionally, blur in an image is thought of as a problem to be removed from the image, however our technique instead makes use of blur as an asset to extract information about the light rays in the scene.

In our formulation, the problem of recovering the EPI from a single swiped image is mapped to that of reconstructing signals with a finite rate of innovation (FRI)[22–26]. We demonstrate that the exact recovery of the EPI is possible when the scene is made of a single fronto-parallel plane with a texture pasted onto it that is a sum of sinusoids.

The paper is organised as follows: in Section 2 we describe the problem formulation and the process of capturing the swiped image, in Section 3 we describe the process of recovering the information contained by the EPI from the pixels of the swiped image, and in Section 4 we show the application of the techniques from Section 3 on real images. Finally, we conclude in section 5.

2. PROBLEM FORMULATION

2.1 The plenoptic function

Consider a scene comprising a single vertical plane onto which a texture that is a sum of sine waves has been pasted. Figure 3 illustrates the scene viewed from above in which a point in space has coordinates (x, z), the camera centre has coordinates $(t, 0)$, ϕ is the angle of the plane, and coordinate s measures the distance along the plane. In this view, the image plane forms a line at $z = f$, the focal length. In the analysis below, a 2D slice of the EPI is considered by restricting $y = w = 0$.

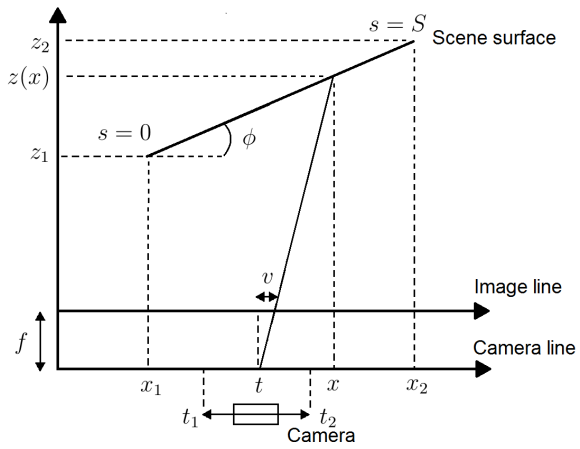

Figure 3. An example of a slanted plane scene.

The light ray that arrives at pixel location v satisfies the equation $fx - ft - vz = 0$, where f is the focal length of the camera. This ray originates from a point on the scene surface with curvilinear coordinate s, given by:

$$s(v, t) = rt + q$$

where

$$r = \frac{f}{f\cos(\phi) - v\sin(\phi)}$$

$$q = \frac{fx_1 + vz_1}{f\cos(\phi) - v\sin(\phi)}$$

where (x_1, z_1) and (x_2, z_2) are the limits of the plane in the scene surface as illustrated in Figure 3.

Under the assumptions given above, the intensity of rays arising from a point on the scene surface can be expressed as a sum of complex sinusoids $I(s) = A + \sum_{n=1}^{N} Be^{j\omega_n s}$, which results in the image intensity:

$$I(v, t) = A + \sum_{n=1}^{N} B_n e^{j\omega_n s(v,t)} = A + \sum_{n=1}^{N} B_n e^{j\omega_n (rt+q)} \tag{1}$$

where ω_n is a frequency of a complex sinusoid on the texture surface, and N is the number of sinusoids. Equation 1 is valid for $f\frac{(x_1-t)}{z_1} < v < f\frac{(x_2-t)}{z_2}$, the range of v in which the plane surface is seen.

2.2 Swiping the camera

Now the effect of swiping the camera from t_1 to t_2 at uniform velocity, with the shutter left open during the swipe, can be determined. The resultant image may be obtained by integrating (1) over the range $t_1 < t < t_2$, shown in Figure 4, which results in:

$$I_I(v) = \int_a^b A + \sum_{n=1}^{N} B_n e^{j\omega_n(rt+q)} dt = \left[At + \sum_{n=1}^{N} \frac{B_n}{j\omega_0 r} e^{j\omega_n(rt+q)} \right]_a^b$$

where the integration limits, a and b, depend on the value of v and the scene geometry, and are given by:

$$a = \min(\max((x_1 - z_1)\frac{v}{f}, t_1), t_2)$$
$$b = \min(\max((x_2 - z_2)\frac{v}{f}, t_1), t_2)$$

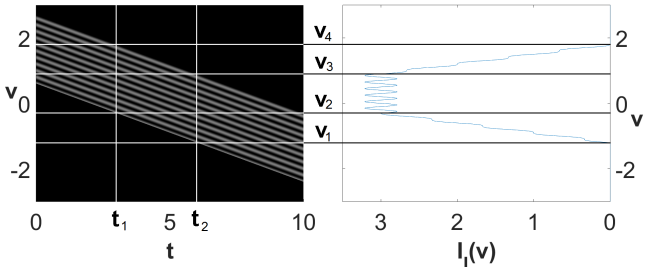

Figure 4. The swiped image (right) formed from the EPI (left) of a fronto-parallel plane with a sinusoidal texture surface.

The resulting integrated image $I_I(v)$ is made of 5 sections that are related to the limits of integration. The four segment boundaries, or switchpoints, are the values of v where t_1 and t_2 cross the edges of the scene plane in the EPI. The switchpoints $v_1, ..., v_4$ can be related to the scene geometry using the equations:

$$v_1 = f(x_1 - t_2){z_1}^{-1} \quad (2) \qquad v_2 = f(x_1 - t_1){z_1}^{-1} \quad (3)$$
$$v_3 = f(x_2 - t_2){z_2}^{-1} \quad (4) \qquad v_4 = f(x_2 - t_1){z_2}^{-1} \quad (5)$$

In practice, the continuous function $I_I(v)$ cannot be measured directly and only the pixels $p[k]$ of the swiped image are seen. Information in $I_I(v)$ is distorted due to the blur introduced by the camera lens (modelled by its point spread function) and the limited density of imaging sensors. This process is similar to the way that acquisition is modelled in traditional sampling theory, and the pixels of one row of the swiped image are given by the inner product:

$$p[k] = \int_{-\infty}^{\infty} \varphi(\frac{v}{T} - k) I_I(v) dv \triangleq \left\langle \varphi(\frac{v}{T} - k), I_I(v) \right\rangle$$

where p are the pixels, k is the pixel index, T is the spacing between pixels and $\varphi(v)$ is the point spread function of the camera or sampling kernel.

Given the pixels $p[k]$ of the swiped image acquired, the goal is to retrieve the information contained within the EPI by using the techniques in Section 3. Recovering the frequencies of the sinusoids in the texture surface, the amplitude and phase of the sinusoids in the texture surface, and the locations of the switchpoints allows the EPI to be exactly determined. With knowledge of the EPI, new views of the scene can be created for arbitrary camera locations.

3. EPI RECOVERY FROM A SWIPED IMAGE

We now consider the case where the plane in the scene is fronto-parallel (i.e $\phi = 0$). Under this assumption $I_I(v)$ can be written as follows:

$$I_I(v) = \sum_{d=1}^{3}\sum_{n=1}^{N} B_{d,n} e^{j\omega_{d,n} v + \psi_{d,n}} \xi_d(v) + \sum_{\rho=0}^{1} A_{d,\rho} t^{\rho} \xi_d(v) \tag{6}$$

where d indexes the segments of $I_I(v)$ between consecutive switchpoints $v_1, ..., v_4$, N is the number of complex exponentials in the texture, $\xi_d(t)$ is the Heaviside step function, ρ is the order of the linear component and $\psi_{d,n}$ is the phase of the sinusoid.

$I_I(v)$ is a piecewise signal in which each segment is a sum of sinusoids; it can therefore be recovered from the pixels $p[k]$ using FRI theory[22–27]. It is assumed here that the point spread function $\varphi(v)$ is an exponential reproducing kernel, as in FRI theory[25], but less restrictive choices are also available[26]. In what follows, we show that it is possible to retrieve the parameters of $I_I(v)$, that is the frequency of the texture and the switch points by adapting to this case the method in[27].

3.1 Determining the sections for local recovery

It has been shown[27] that if a sampled signal $p[k]$ is a sum of weighted exponentials $p[k] = \sum_{n=1}^{N} a_n u_n^k$ then a filter with the z transform of $H_{\mathbf{u}}(z) = \prod_{n=1}^{N}(1 - u_n z^{-1})$ will annihilate the signal $p[k]$. This can be expressed in matrix form as:

$$\mathbf{Ph} = \begin{pmatrix} \vdots & \vdots & \dots & \vdots \\ p[0] & p[-1] & \dots & p[-N] \\ p[1] & p[0] & \dots & p[-N+1] \\ \vdots & \vdots & \ddots & \vdots \\ p[N] & p[N-1] & \dots & p[0] \\ \vdots & \vdots & \dots & \vdots \end{pmatrix} \begin{pmatrix} h[0] \\ h[1] \\ \vdots \\ h[N] \end{pmatrix} = 0$$

where $p[k]$ is the finite difference of the pixel values, $\mathbf{P}$ is a $(N+1) \times (N+1)$ Toeplitz matrix created with $2N+1$ samples of $p[k]$, and $\mathbf{h}$ defines the impulse response of $H(z)$.

If the $2N+1$ samples of $p[k]$ used to form $\mathbf{P}$ include contributions from only N sinusoids, then $\mathbf{P}$ will be rank deficient. A sliding window over $p[k]$ can be used to find window positions which contain only contributions from pure sinusoids, undistorted by switchpoints. Using the finite difference of the pixels to eliminate the constant term, windows in the third segment can be used to detect the frequency of the sinusoidal component. The frequency found by using these windows represents the frequency of the sinusoidal texture surface on the plane.

3.2 Determining the texture frequency

The frequency of the sine waves in each rank deficient window can be found by constructing the annihilating filter $H_{\mathbf{u}}(z) = \prod_{n=1}^{N}(1 - u_n z^{-1})$, and finding the roots of the filter. The roots will be pairs in the form $z = e^{j\omega_n}$ and $z = e^{-j\omega_n}$. The phase and amplitude of the sinusoids in the rank deficient window can then be found by constructing a Vandermonde system. For example, for a system with $N = 1$, the Vandermonde system can be written as:

$$\frac{1}{2}\begin{pmatrix} e^{j\omega_0 k} & e^{-j\omega_0 k} \\ e^{j\omega_0(k+1)} & e^{-j\omega_0(k+1)} \end{pmatrix} \begin{pmatrix} B\varphi(-\omega_0)e^{j\psi_0} \\ B\varphi(\omega_0)e^{-j\psi_0} \end{pmatrix} = \begin{pmatrix} p[k] \\ p[k+1] \end{pmatrix}.$$

3.3 Determining the switchpoint locations

Once the annihilating filter ($H_{\mathbf{u}}(z)$) has been found for the $\mathbf{P}$ matrix, it can be applied to the entire sequence of pixels to form a new signal[27]:

$$p^{'}[k] = h_{\mathbf{u}}[k] * \left\langle \varphi(\frac{v}{T} - k), I(v) \right\rangle = \left\langle \varphi(\frac{v}{T} - k) * \beta_{\mathbf{u}}(\frac{v}{T} - k), \hat{I}_I(v) \right\rangle$$

where $p^{'}[k]$ is the sampled pixels of the swiped image after the frequencies ω_N have been filtered out by $h_{\mathbf{u}}[k]$, $\hat{I}_I(v)$ is a stream of differentiated Diracs at the switchpoint locations, and $\beta_{\mathbf{u}}$ is an e-spline created from the ω_Ns[27]. Here, the fact that $p[k] = \left\langle \varphi(v - k), I_I(v) \right\rangle$ is also used.

This equation shows that the resulting signal $p^{'}[k]$ is the equivalent of the inner product between differentiated Diracs at the switch points, and a new sampling kernel $\varphi^{'}(t) = \varphi(\frac{v}{T} - k) * \beta_{\mathbf{u}}(\frac{v}{T} - k)$, which can be shown to still be able to reproduce exponentials[27].

To find the location of the switchpoints the moments of the signal can be calculated:

$$\tau[m] = \sum_{k \epsilon Z} c^{'}_{m,k} p^{'}[k]$$

where $c^{'}_{m,k}$ fulfills:

$$\sum_{k \epsilon Z} c^{'}_{m,k} \varphi^{'}(v - k) = e^{\alpha_m v}$$

where α_m are the frequencies reproduced by the e-spline sampling kernel, $\beta_{\mathbf{u}}(k)$.

The moments of $p^{'}[k]$ are a set of weighted exponential, and can therefore be annihilated by a filter of the form:

$$G(z) = \prod_{d=1}^{4} (1 - e^{\lambda v_d} z^{-1})^{8N}$$

where λ is the spacing between frequencies in α_m, and where the roots of the filter $G(z)$ can be used to find the switchpoint locations. Finally, the values of x_1, x_2, and $z_1 = z_2$ can be found from Eq. (2) to (5).

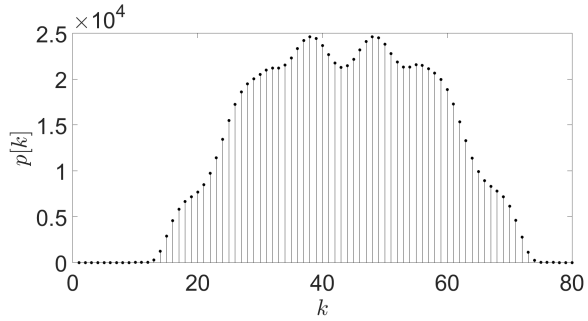

Figure 5. A sampled horizontal slice of a simulated swiped image of a fronto-parallel plane with a sinusoidal texture (i.e. $N = 2$).

The process of recovering the EPI from a row of pixels from a swiped image (such as shown in Figure 5) was verified using MATLAB. The technique outlined in this section allowed the recovery of the EPI, with switchpoint locations recovered exactly, even after subsampling the full resolution image.

4. NUMERICAL RESULTS

To verify the theory proposed in Section 3, experimental verification was performed. A DSLR camera mounted on a sliding mount was moved between two points. A printed paper sheet of a sinusoidal texture surface, with a single frequency, was set up as a fronto-parallel plane. A large number of still images were taken, of which a subset are shown in Figure 6(a), and added to form a swiped image of the scene. To simplify the analysis of the scene, there was no variation in the vertical direction, which allowed the lens distortion to be modelled as 1D. The pixel values outside the horizontal boundaries of the plane were set as zero to reduce noise.

Figure 6(a) shows 3 of the original images which contain noise in addition to the sinusoidal texture. In Figure 6(b), the resulting swiped image, which is very blurred, is shown, however it still contains all the information necessary to retrieve the original EPI. The original swiped image was subsampled by a factor of 50 to create a swiped image only 80 pixels across. The noise present in the images resulted in the amplitude of the signal deviating from what it should have been given no noise. There was also distortion due to the point spread function using an exponential reproducing kernel[26], despite the pixel values not exactly corresponding to those modelled in MATLAB in Section 3. To reduce the impact of this deviation, the frequency of the sinusoid and time of the switchpoints were calculated for each valid sliding window and averaged. Hard thresholding of samples outside the length of the sampling kernel also was employed to reduce the impact of noise.

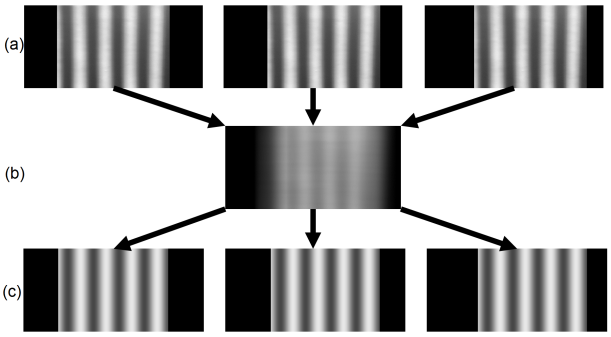

Figure 6. An example of EPI reconstruction from a swiped image. The images in (a) are combined into a swiped image (b), from which the EPI can be recovered and used to synthesise new images as in (c).

Use of the technique outlined in Section 3 showed that the frequency of the sinusoid could be recovered within 1 percent of its true value, and the locations of the switchpoints could be recovered within half a pixel. From the recovered values, the EPI of the scene could be reconstructed. This EPI can be used to generate an image of the scene from an arbitrary camera location, as shown in Figure 6(c).

5. CONCLUSION

This paper demonstrates that it is possible, given a swiped image of a scene of a flat plane with a sinusoidal texture signal, to reconstruct exactly the plenoptic function and hence the scene geometry. This is possible given knowledge of the distortion introduced by the camera lens and despite the image being sampled. Extensions of this method to more general settings are being investigated.

REFERENCES

[1] Adelson, E. H. and Bergen, J. R., [*The plenoptic function and the elements of early vision*], Vision and Modeling Group, Media Laboratory, Massachusetts Institute of Technology (1991).

[2] Gilliam, C., Dragotti, P. L., and Brookes, M., "On the spectrum of the plenoptic function," *IEEE Transactions on Image Processing* **23**(2), 502–516 (2014).

[3] Chai, J.-X., Tong, X., Chan, S.-C., and Shum, H.-Y., "Plenoptic sampling," in [*Proceedings of the 27th annual conference on Computer graphics and interactive techniques*], 307–318, ACM Press/Addison-Wesley Publishing Co. (2000).

[4] Da Cunha, A. L., Do, M. N., and Vetterli, M., "On the information rates of the plenoptic function," *IEEE Transactions on Information Theory* **56**(3), 1306–1321 (2010).

[5] Do, M. N., Marchand-Maillet, D., and Vetterli, M., "On the bandwidth of the plenoptic function," *IEEE Transactions on Image Processing* **21**(2), 708–717 (2012).

[6] McMillan, L. and Bishop, G., "Plenoptic modeling: An image-based rendering system," in [*Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*], 39–46, ACM (1995).

[7] Levoy, M. and Hanrahan, P., "Light field rendering," in [*Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*], 31–42, ACM (1996).

[8] Bolles, R. C., Baker, H. H., and Marimont, D. H., "Epipolar-plane image analysis: An approach to determining structure from motion," *International Journal of Computer Vision* **1**(1), 7–55 (1987).

[9] Kang, S. B., Li, Y., Tong, X., and Shum, H.-Y., "Image-based rendering," *Foundations and Trends in Computer Graphics and Vision* **2**(3), 173–258 (2006).

[10] Shum, H. and Kang, S. B., "Review of image-based rendering techniques," in [*Visual Communications and Image Processing 2000*], 2–13, International Society for Optics and Photonics (2000).

[11] Wong, T.-T., Heng, P.-A., Or, S.-H., and Ng, W.-Y., "Image-based rendering with controllable illumination," in [*Rendering Techniques*], 13–22, Springer (1997).

[12] Nguyen, H. T. and Do, M. N., "Error analysis for image-based rendering with depth information," *IEEE Transactions on Image Processing* **18**(4), 703–716 (2009).

[13] Zhang, C. and Chen, T., "A survey on image-based rendering representation, sampling and compression," *Signal Processing: Image Communication* **19**(1), 1–28 (2004).

[14] Shum, H.-Y., Chan, S.-C., and Kang, S. B., [*Image-based rendering*], Springer Science & Business Media (2008).

[15] Pearson, J., Brookes, M., and Dragotti, P. L., "Plenoptic layer-based modeling for image based rendering," *IEEE Transactions on Image Processing* **22**(9), 3405–3419 (2013).

[16] Berent, J. and Dragotti, P. L., "Plenoptic manifolds," *IEEE Signal Processing Magazine* **24**(6), 34–44 (2007).

[17] Gelman, A., Dragotti, P. L., and Velisavljevic, V., "Multiview image coding using depth layers and an optimized bit allocation," *IEEE Transactions on Image Processing* **21**(9), 4092–4105 (2012).

[18] Gelman, A., Berent, J., and Dragotti, P. L., "Layer-based sparse representation of multiview images," *EURASIP Journal on Advances in Signal Processing* **2012**(1), 1–15 (2012).

[19] Ng, R., Levoy, M., Brédif, M., Duval, G., Horowitz, M., and Hanrahan, P., "Light field photography with a hand-held plenoptic camera," *Computer Science Technical Report CSTR* **2**(11), 1–11 (2005).

[20] Nayar, S. and Ben-Ezra, M., "Motion-based motion deblurring," *IEEE transactions on pattern analysis and Machine intelligence* **26**(6), 689–698 (2004).

[21] Shan, Q., Jia, J., and Agarwala, A., "High-quality motion deblurring from a single image," in [*ACM Transactions on Graphics (TOG)*], **27**(3), 73, ACM (2008).

[22] Vetterli, M., Marziliano, P., and Blu, T., "Sampling signals with finite rate of innovation," *IEEE Transactions on Signal Processing* **50**(6), 1417–1428 (2002).

[23] Tan, V. Y. and Goyal, V. K., "Estimating signals with finite rate of innovation from noisy samples: A stochastic algorithm," *IEEE Transactions on Signal Processing* **56**(10), 5135–5146 (2008).

[24] Bi, N., Nashed, M. Z., and Sun, Q., "Reconstructing signals with finite rate of innovation from noisy samples," *Acta applicandae mathematicae* **107**(1-3), 339–372 (2009).

[25] Dragotti, P. L., Vetterli, M., and Blu, T., "Sampling moments and reconstructing signals of finite rate of innovation: Shannon meets Strang-Fix," *IEEE Transactions on Signal Processing* **55**(5), 1741–1757 (2007).

[26] Uriguen, J. A., Blu, T., and Dragotti, P. L., "Fri sampling with arbitrary kernels," *IEEE Transactions on Signal Processing* **61**(21), 5310–5323 (2013).

[27] Berent, J., Dragotti, P. L., and Blu, T., "Sampling piecewise sinusoidal signals with finite rate of innovation methods," *IEEE Transactions on Signal Processing* **58**(2), 613–625 (2010).