Computational Cameras: Convergence of Optics and Processing

Changyin Zhou, *Student Member, IEEE,* and Shree K. Nayar, *Member, IEEE,*

Abstract—A computational camera uses a combination of optics and processing to produce images that cannot be captured with traditional cameras. In the last decade, computational imaging has emerged as a vibrant field of research. A wide variety of computational cameras have been demonstrated to encode more useful visual information in the captured images as compared to conventional cameras. In this paper, we survey computational cameras from two perspectives. First, we present a taxonomy of computational camera designs according to the coding approaches, including object side coding, pupil plane coding, sensor side coding, illumination coding, camera arrays and clusters, and unconventional imaging systems. Second, we use the abstract notion of light field representation as a general tool to describe computational camera designs, where each camera can be formulated as a projection of a high dimensional light field to 2D image sensor. We show how individual optical devices transform light fields and use these transforms to illustrate how different computational camera designs (collections of optical devices) capture and encode useful visual information.

I. INTRODUCTION

A camera is a device that captures light from scenes. Over the last century, the evolution of cameras has been truly remarkable. However, through this evolution the underlying camera model remains essentially the same, namely, the camera obscura (Figure 1(b)). The traditional camera has a detector and a standard lens which captures only those principal rays that pass through its center of projection, or an effective pinhole, to produce familiar linear perspective images. In other words, the traditional camera performs a very simple and restrictive sampling of the complete set of rays, or the light field that resides in any real scene [1] [2].

A computational camera (Figure 1(c)), in contrast, uses a combination of novel optics and computations to produce the final image [1]. The novel optics are used to map rays in the light field of the scene to pixels on the detector in some unconventional fashion. For instance, a ray shown in Figure 1(c) has been geometrically redirected by the optics to a different pixel from the one it would have arrived at in the case of a traditional camera. As illustrated by the change in color from yellow to red, the ray could also be photometrically altered by the optics. In general, the new arrangement of rays helps to encode more useful visual information in a computational camera than a conventional camera.

C. Zhou and S.K. Nayar are with the Department of Computer Science, Columbia University in the City of New York, New York, NY 10027, USA e-mail: {changyin, nayar}@cs.columbia.edu.
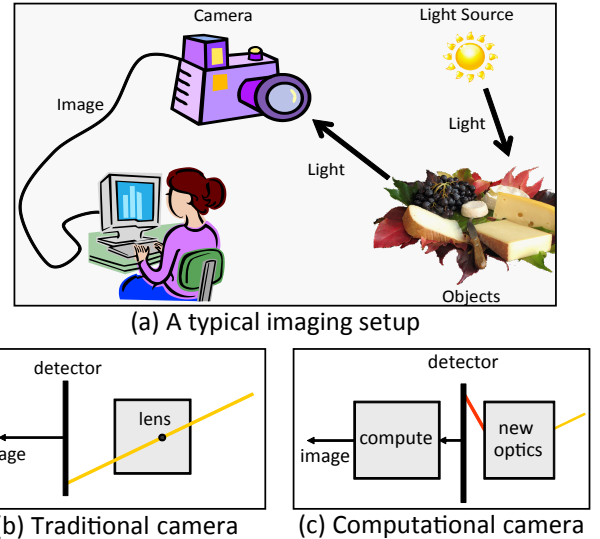

Fig. 1. (a) In a typical scene for imaging, light rays from sources are reflected by objects, collected by camera lens, and then converted to digital signals for further processing. (b) A traditional camera model captures only those principal rays that pass through its center of projection to produce the familiar linear perspective image. (c) A computational camera uses optical coding followed by computational decoding to produce new types of images. [1]

Although the images captured by computational cameras are optically coded and may not be visually meaningful in their raw form, the information can be recovered by using computation. This combination of novel optics and computation hence can produce new types of images that are potentially beneficial to a vision system. The vision system could either be a human observing the image or a computer vision system that uses the image to interpret the scene.

A. Overview of the Survey

In this paper, we survey computational camera techniques along two intertwined lines – coding approaches in camera design and their light field representations.

1) Coding approaches: The design space for the optics of computational cameras is large. It would be desirable to have a single design methodology that produces an optimized optical system for any given set of imaging specifications. The optimization criteria could incorporate a variety of factors, including performance and complexity. At this point in time, however, such a systematic design approach is largely missing. Consequently, as with traditional optics, the design of computational cameras remains part science and part art.

The coding methods used in today's computational cameras can be broadly classified into six approaches.
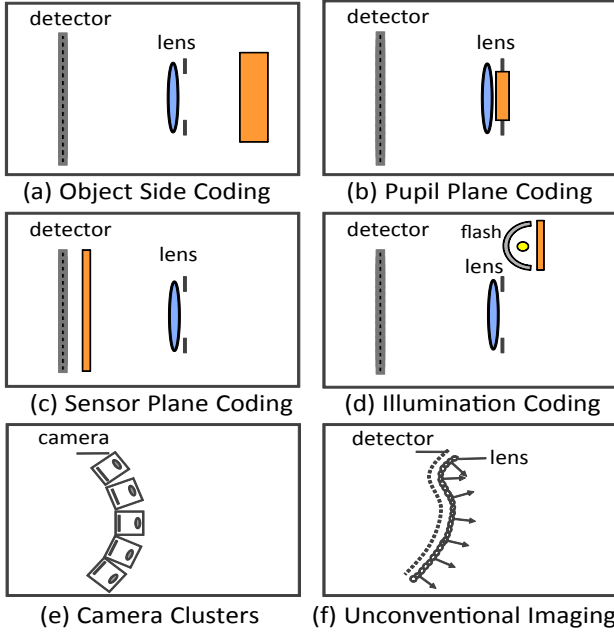
Fig. 2. Optical coding approaches used in computational cameras. (a) Object side coding, where an optical element is attached externally to a conventional lens. (b) Pupil plane coding, where an optical element is placed at, or close to, the aperture of the lens. (c) Sensor side coding, where an optical element is behind the lens. (d) Imaging systems that make use of coded illumination. (e) Imaging systems that are made up of a cluster or array of traditional camera modules. (f) Imaging systems using unconventional camera architectures or non-optical devices. [2]

- **Object side coding** (Figure 2 (a)) attaches external devices to the camera and is probably the most convenient way to implement computational cameras. Since the attached device is positioned in front of the main lens, objects at different field angles can be observed at different regions of the device. Therefore, object side coding would provide spatially varying light modulation. This property has been widely used for various applications. (Section III)

- **Pupil plane coding** (Figure 2 (b)) puts optical elements in the pupil plane of the main lens. Since any rays from objects ideally pass through the same pupil plane, pupil plane coding can be used to provide spatially invariant light modulation and manipulate the system Point-Spread-Function (PSF). (Section IV)

- **Sensor side coding** (Figure 2 (c)) places additional optical elements behind the lens, either on the sensor plane or in front of the sensor. In particular, coding on the sensor plane can yield pixel-wise modulation and is useful in many applications. An emerging type of sensor side coding is to use sensor motion. (Section V)

- **Illumination coding** (Figure 2 (d)) alters captured images by using a spatially and/or temporally controllable camera flash. This approach enables image coding in ways that are not possible by only modifying the imaging optics. Illumination coding has a long history in the field of computer vision. For example, virtually any structured light method (see [3, 4] for surveys) or a variant of photometric stereo [5] is based on the notion of illumination coding. (Section VI)

- **Camera clusters or arrays** (Figure 2 (e)) provide a more flexible and economical way to transcend the limits of individual cameras by combining multiple cameras. (Section VII)

- **Unconventional coding** (Figure 2 (f)) briefly discusses some techniques that cannot fit well into the above five categories. This includes computational camera designs using unconventional architectures or non-optical devices. (Section VIII)

2) Light field representation: Light field is a function that describes the amount of light of each wavelength traveling in every direction through every point in space and time. The concept of light field has a long history back to the 19th century [6]. It was introduced to vision and graphics in the 1990s [7, 8, 9]. A complete light field in geometrical optics is often represented by a 7D plenoptic function $L(x, y, z, \Phi, \theta, \lambda, t)$, where (x, y, z) is the spatial location of a light ray, (Φ, θ) is the direction of the ray, λ is the light wavelength, and t is the time. Given time and wavelength, the light field is 5D and can be further reduced to 4D in a free space because a light ray will not change along its propagation direction ([8] [9]).

The light field representation can be a common language to formulate various computational camera designs. Mathematically, a camera can be defined as a projection from high dimensional light fields to 2D images. The projection is implemented through a combination of sequential optical devices. In the paraxial realm, most optical devices can be formulated as linear transforms on light fields in the 4D space. For example, a lens can be defined as a shearing transform on the incoming light field and a prism can be defined as a translation on the incoming light field [10, 11]. As a result, each camera can be mathematically defined as a combination of linear element operators in the 4D light field space. One goal of this survey is to propose light field representations as a pedagogical tool in order to provide readers a deeper insight into various typical computational camera designs.

B. Definition of Scope

Two other concepts that are highly related to computational camera are *computational photography* and *computational imaging*. These concepts are heavily overlapping and the distinction among them is quite vague. Roughly speaking, computational photography techniques focus more on how multiple images (often captured with traditional cameras) are combined computationally for certain vision or graphics tasks. Computational imaging techniques focus more on the process of imaging than on the camera itself, and sometimes do not have cameras in a compact form. Finally, computational camera emphasizes on camera designs on how cameras can be designed to serve certain purposes. In this survey of computational cameras, we focus on camera designs and will not cover techniques without modifying the camera. In particular, many recently proposed computational photography techniques use traditional cameras and capture images with different camera settings (e.g., aperture size, exposure time, focus) (few examples are [12, 13, 14, 15, 16]). These techniques, as they involve no camera modification, will not be covered in this survey.

(a) Point pair on two planes   (b) Point pair on a spherical surface

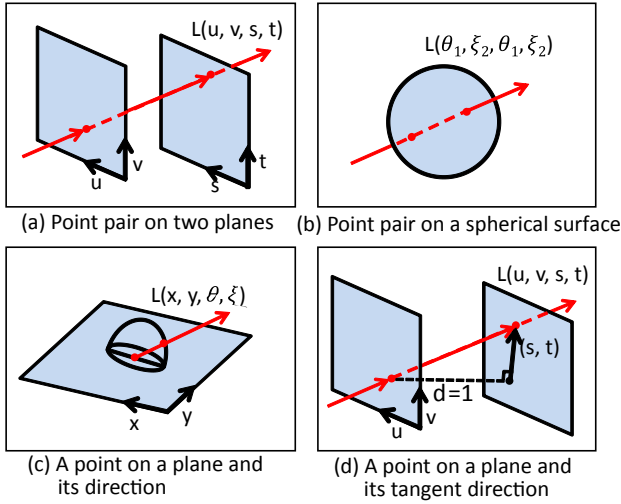(c) A point on a plane and its direction   (d) A point on a plane and its tangent direction

Fig. 3. Four types of light field parameterization. (a) Use a pair of points on two parallel planes. (b) Use a pair of points on a spherical surface. (c) Use a point on a plane and its direction. (d) Use a point on a plane and its tangent direction. The parameterization (d) greatly simplifies many formulations and derivations in light field transforms. Therefore, we have chosen to use it in this paper. This parameterization is also similar to phase space, which is often used in optics.

Camera designs can be very different at different scales [17, 18]. In this survey, we restrict ourselves to cameras at the medium scale, i.e. the ones that operate with incoherent light within or close to the visible spectrum for common vision and graphics tasks. Note that extremely large scale cameras (e.g., telescopes used in astronomy with focal lengths larger than 1m) usually have a very small field- of-view (FOV), work with certain special wavelengths and scenes, and are particularly concerned with optical aberrations. In contrast, cameras at extremely small scales (for example, camera lenses with focal lengths smaller than 1mm) are often concerned with pixel size and diffraction and require coherent illumination. For imaging systems at the micro or macro scale, one can find a large body of literature in optics and astronomy.

Most discussions in this paper are in the realm of geometrical optics. One exception is polarization. Polarization, although a concept in wave optics, can be easily observed and has often been used in computational camera designs. The geometrical light field representation can be generalized to Fourier optics by using the Wigner distribution function (e.g., [17, 19, 20, 21, 22, 23]). However, imaging techniques using wave optics will not be the focus of this survey. This survey will also not cover the work on Time-of-Flight cameras (see [24] for a survey of Time-of-Flight cameras.)

C. Previous Reviews

Nayar [1] discusses the concept of computational cameras using a number of typical computational camera designs. Levoy [25] gives an overview of computational camera designs for light field acquisition. In a recent technical report [2], the author proposes a taxonomy of computational cameras using coding approaches of camera design and briefly lists a number of typical techniques for each category. This paper follows the definition and taxonomy of computational cameras proposed in [2], but will provide a much more comprehensive review. In addition, we use a light field representation as a common
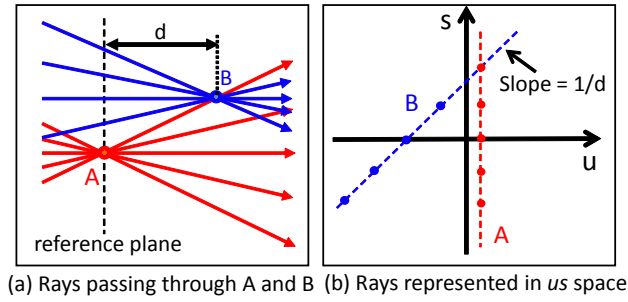

(a) Rays passing through A and B   (b) Rays represented in *us* space

Fig. 4. Rays represented in the *us* space. (a) A bunch of rays passing through Point A and Point B. (b) The light field on the reference plane with *us* representation. Each ray in (a) becomes a point in the *us* space. All the rays passing through a scene point is represented by a line in the *us* space, whose slope is the inverse of the distance from the point to the reference plane.

language to interpret typical computational cameras designs, in order to provide a more insightful understanding of this area.

Raskar et al. [26] discuss the scope and elements of computational photography and camera research, and survey the popular research topics at that time, which are broadly grouped into high dynamic range, aperture and focus, motion blur, and computational illumination.

Wetzstein et al. [27] provide a state-of-the-art report of the existing computational photography and computational camera techniques. The report gives another taxonomy that is different from that in [2] and this survey. It enumerates the imaging and computation techniques with respect to five dimensions in the plenoptic function of light field, including dynamic range, spectrum, spatial resolution, angular resolution, and time. The taxonomy in [27] and the one in this paper are quite orthogonal to each other. One is from the perspective of *what* visual information can be extracted [27]. The other one, as used in this paper and in [2], is from the perspective of *how* this information can be extracted. The latter thus focuses more on how computational cameras work and how they can be designed.

II. FUNDAMENTALS IN LIGHT FIELD REPRESENTATION

In geometrical optics, the 4D light field can be parameterized in several different ways, as shown in Figure 3. In this paper, we will use Position (u, v) and its tangent direction (s, t) to represent light rays, as shown in Figure 3 (d). While all parameterizations are essentially the same, this particular $uvst$ representation makes the formulation most concise and intuitive to understand. This is because light field transforms caused by most typical devices in a camera can be represented by simple linear operators under this parameterization.[1]

As an example, Figure 4 illustrates how rays passing through two scene points (a) are represented in the $uvst$ space (b) (using the plane of Object A as a reference plane). For the purpose of illustration, we use a 2D light field in us representation instead of the 4D $uvst$ representation. We can see that all the rays through one scene point will form a line in the us space. The slope of the line is the inverse of d, the

---

[1]Phase space is a concept in wave optics that is highly related to this light field representation. For more information about phase space, one can refer to the optics literature (e.g., [28, 20, 19, 29, 21]).

distance from the point to the reference plane. For example, for Point A, which is on the reference plane ($d = 0$), the light field is a vertical line.

A. Optical Element Formulation

A camera consists of a number of optical elements. The input light goes through layers of optical devices and reaches the sensor. These optical devices are chosen and arranged in a way such that a desired optical processing will be made to encode useful visual information. Shown in Figure 5 are six types of commonly used optics devices in camera design. The functionalities of these six devices can be formulated as linear operators within the $uvst$ space [10, 30, 11]. As before, for simplicity, we will be illustrating the transforms in a 2D us space.

1) **Space**, although is not often called a physical optical device, is important in camera design. As shown in Figure 5 (a), when light field propagates from one plane to another parallel plane, it is sheared in the u dimension. The shearing angle is exactly d, the distance between the two planes.

2) **Lens** is often used to focus light rays. As shown in Figure 5(b), it shears input light fields in s dimension by an angle of $1/f$, where f is the focal length of the lens.

3) **Prism** is another typical device often used in computational cameras. With the paraxial assumption, it deviates incoming rays by a constant angle $\theta = (n - 1) \cdot \alpha$, where n is the refractive index of the glass and α is the angle of the wedge of prism. In the us space, a prism shifts light fields along s dimension as shown in Figure 5(d). In addition, a prism can also produce the effect of dispersion, which can be used for multi-spectrum imaging.

4) **Optical diffuser** is a device that scatters light rays (Figure 5 (e)). Diffusers of various scattering patterns can be made by manipulating their holographic surface profiles. In the us space, the scattering is presented as a convolution in the s dimension, where the convolution kernel is determined by its scattering pattern [31, 32].

5) **Intensity modulator** is an optical device that attenuates the intensity of the incoming rays (Figure 5(e)) and can be made of many materials (e.g., photomasks [33, 34, 35, 36], Liquid Crystal display (LCD) [37], Liquid Crystal on Silicon (LCoS) [38] and Digital Micromirror Device (DMD) [39, 40, 41, 42]). Color filter is also a type of intensity modulator whose attenuation is wavelength dependent [43, 44, 45]. In the us space, it can be formulated as a dot-product in u dimension.

6) **Phase modulator** is an optical device that only changes the phase of the incoming rays and is usually a piece of glass of a certain 3D profile (Figure 5 (f)). Lens and prism are two types of mostly commonly used phase modulators. A phase modulator refracts the light according to the deviation of the 3D surface. In the us space, it distorts light fields in s dimension and the amount of distortion at each u location is proportional to the corresponding surface gradient.

B. Traditional Camera

An ideal traditional camera consists of a lens, an aperture stop, a sensor, and free spaces between them. Figure 6 (a) shows three scene points at different depths imaged by a traditional camera. Figure 6 (b) shows how the light field is transformed during the process of imaging from object plane, to aperture, to lens, and finally to the sensor plane. We can see that the light field on Plane L_0 is first sheared in u dimension due to the space between L_0 and L_1, then cropped due to the aperture (L_2), sheared in s dimension due to the lens (L_3), sheared again due to the space between lens and sensor (L_4), and finally projected on the u axis to produce the final image as shown in (c). This process of light field transforms in a traditional camera will be frequently used in this survey as a reference to understand other computational camera designs.

C. Visual Information in Light Field

In the area of imaging, visual information in light fields has been widely discussed. As shown in Figure 6, the input to the camera is the light field on Plane L_2. A camera cannot produce extra information, but can only be designed to best preserve desired visual information in the input light field during the transforms. In the 2D light field on Plane L_2 illustrated in Figure 6, each line represents the rays from a single scene point; the line slope encodes the depth information; and the variation in a single line tells the reflectance property of the corresponding scene point.

When the complete 7D light field $L(x, y, z, \Phi, \theta, \lambda, t)$ is considered, imaging is a projection from a 7D space to a 2D image space. For different tasks, one may want to recover information in other dimensions [27]. Adelson and Bergen [7] discuss early visual information (e.g. image edges) in light field and suggested several image filters to extract these information.

In this survey, we will discuss particularly how each coding approach helps to preserve useful visual information for further processing or decoding, and will formulate and illustrate typical camera designs in the light field space to facilitate insightful understanding of these techniques.

III. Object Side Coding

Object side coding approaches require optics to be externally attached to a traditional camera (Figure 2(a)). This is the most convenient way to implement a computational camera. Because of the distance between the optical element and the lens, the cones of light rays from objects at different field angles will intersect with the element at different areas. As a result, if the surface profile is not homogeneous, object side coding will yield spatially varying modulation. This property has been widely used to encode more useful visual information. Since extra visual information is captured in object side coding by making multiple different observations, the processing afterwards usually involves pixel-wise or patch registration and fusion. Another type of object side coding, although less common, has been proposed by using homogeneous filters. This type of design will use temporal variations instead of spatial variation to encode information.

(a) Space propagation shears a light field in u dimension

(b) A single lens shears a light field in s dimension

(c) A prism shifts a light field in s dimension

(d) An optical diffuser blurs a light field in s dimension

(e) A photomask performs a dot-product in u dimension

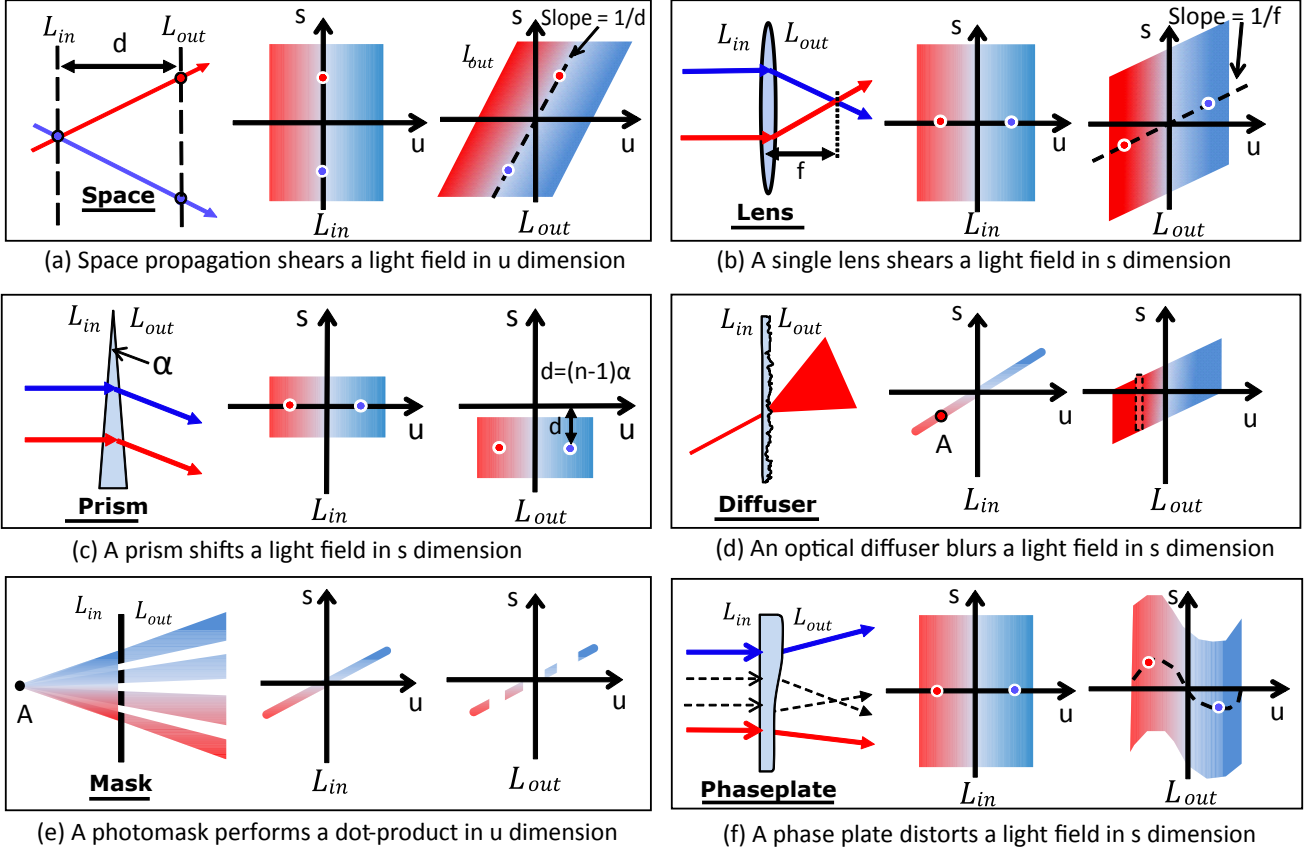(f) A phase plate distorts a light field in s dimension

Fig. 5. Six types of typical optical elements that are often used in computational cameras. (a) Space, although not an optical device, is widely used in optical design. Space propagation shears light fields in u dimension. (b) A lens shears light fields in s dimension by an amount of $1/f$, where f is the lens focal length. (c) A prism shifts light fields in s dimension. (d) An optical diffuser performs a convolution in u dimension in the spatial domain or a dot-product in the Fourier domain. (e) In contrast to optical diffusers, a photomask performs a dot-product in u dimension in the spatial domain. (f) A phaseplate distorts light fields in s dimension. The amount of distortion is proportional to the derivative of the phaseplate surface.

(a) Geometry of a traditional camera

(c) The captured image

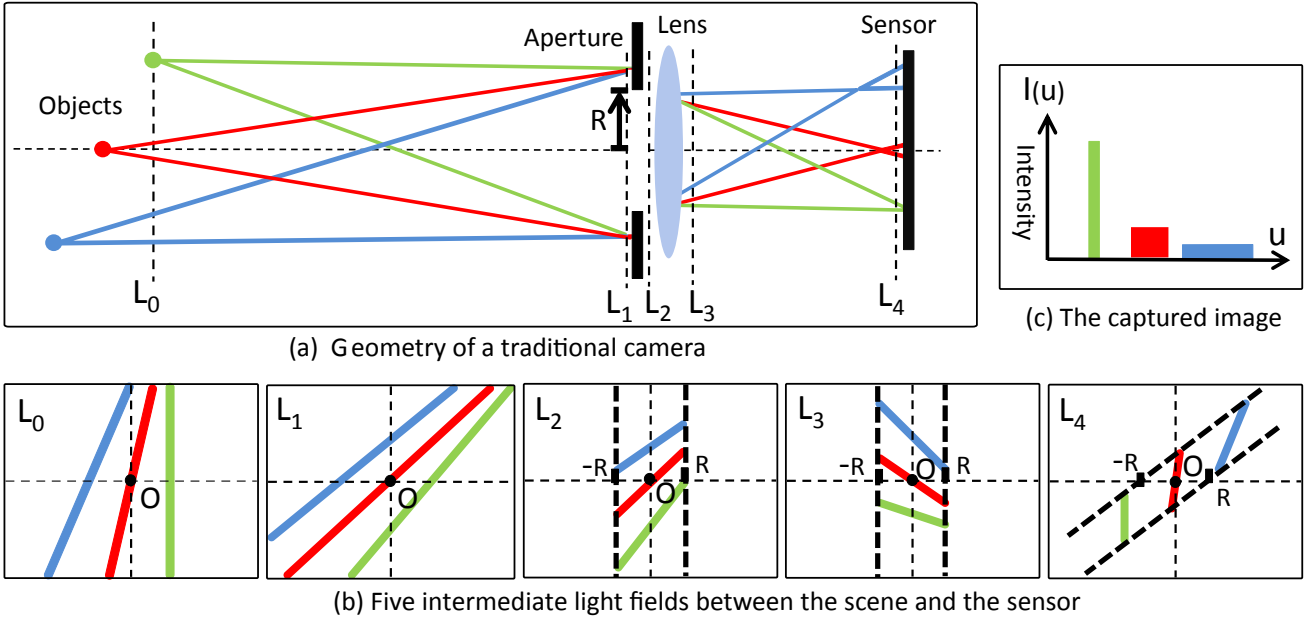(b) Five intermediate light fields between the scene and the sensor

Fig. 6. Light field transforms in a traditional lens camera. (a) Three scene points at different depths are imaged by a traditional camera. The green point is in focus. (b) From left to right are the light fields on L_0 (the plane of nearest scene point), L_1 (the plane right before the aperture), L_2 (the plane right after the aperture), L_3 (the plane after the lens), and L_4 (the plane right before the sensor), respectively. The red line intersects the u axis at the origin because the red point is at the center of field of view. (c) The final captured image is achieved by projecting the light field at L_4 onto the u axis.
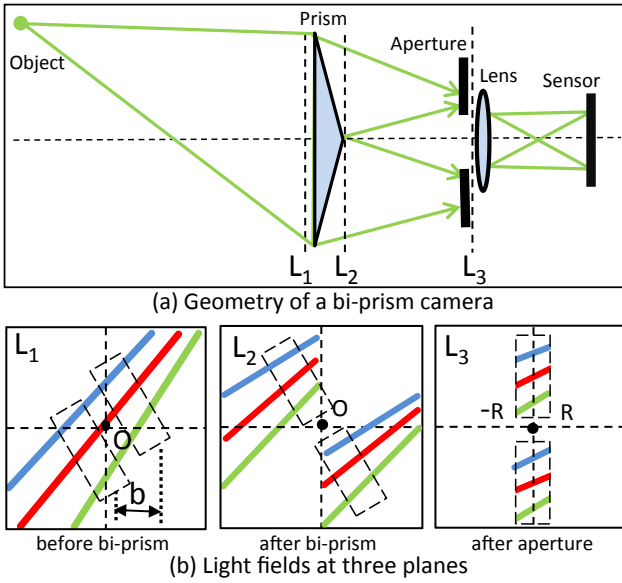
Fig. 7.   Light field transform in a bi-prism stereo camera [46]. (a) The geometry of a bi-prism camera. Light rays from a scene point will be slitted by the bi-prism into two clusters and forms two image points on the sensor. The distance between these two points can be used as the disparity in stereo vision for depth estimation. (Red and blue points are not shown for the simplicity of illustration.) (b) From left to right are the intermediate light fields before the bi-prism, after the bi-prism, and after the aperture, respectively. Compare the light field on L_3 with the one on L_2 in Figure 6, we can see that two copies of the light field have been created.

A. Spatially Variant Filter

1) Depth Estimation: Lee et al. [46] proposed using a bi-prism in front of lens for stereo vision with a single camera. As shown in Figure 7(a), light rays from any single point will be split into two by the biprism and produce two image points on the sensor as if viewed from two viewpoints. This yields an effect of stereo.

The bi-prism system can be understood easily and thoroughly from its light representation. Figure 7 (b) illustrates the light fields at Plane L_1, L_2, and L_3, respectively. The bi-prism splits the input light field into two pieces and shifts them by θ in different s directions, where θ is the absolute angle of deviation of each prism. Comparing the light field after the aperture (L_3) here with that of a traditional camera (Figure 6(b) L_2), we now have two copies of light fields. These two copied will be mapped onto two separated regions in the sensor. Tracing these two copies back to the input light field (Figure 7(b) left), and we can see that two samples from each stripe in the input light field are captured. From the two samples, one can compute the depth as the inverse of the slope of each stripe. As a stereo technique, its baseline can be easily derived as $b = 2 \cdot \theta \cdot d$, where d is the distance from the bi-prism to the lens.

Instead of using a bi-prism, other object side coding techniques have also been proposed by using mirrors [47, 48], or rotating glass plates or mirrors [49, 50, 51].

2) Light Field Acquisition: Georgeiv et al. [52] propose using an array of lens-prism pairs in front of the main lens to capture light fields as shown in Figure 8. The geometry of the camera is shown in Figure 8 (a). Each prism in the array has a different angle of deviation, and therefore, similar to the
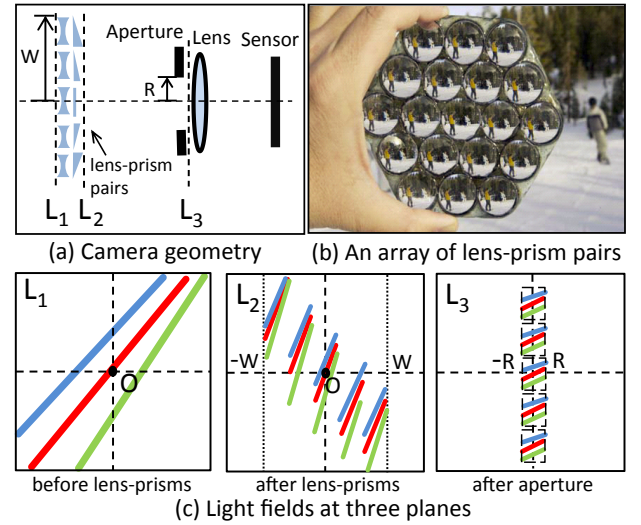

Fig. 8.   Light field camera using an array of pairs of negative lens and prism [52]. (a) The geometry of the light field camera. (b) A prototype of the lens-prism array. (c) From left to right are the intermediate light fields before the lens-prism array, after the array, and after the aperture, respectively. Similar to Figure 7, the prism arrays will split the input light field and create multiple copies, each of them capturing one low-dimensional slice of the high-dimensional light field.

bi-prism design (Figure 7), the prism array splits the field of view into multiple pieces, each corresponding to a different viewpoint. In other words, the camera observes the same small field-of-view (FOV) but from different viewpoints. The information captured by the sensor can be used to reconstruct the 4D light field. A negative lens in front of each prism is used for increasing the overall field-of-view. This light field camera sacrifices spatial resolution for angular resolution (the number of lens-prism pairs). In [52] and [11], the authors also mentioned other possible object side configurations for light field acquisition by arranging prisms and lens in different ways.

3) Wide Field of View Imaging: Catadioptric techniques combine lenses and mirrors in camera design and is often used to increase camera FOV [53, 54, 55, 56, 57, 58, 59, 60]. These technique have significant impacts on a variety of real-world applications, including surveillance, autonomous navigation, virtual reality, and video conferencing [61, 62, 63].

One problem with wide FOV catadioptric camera designs is that while a convex mirror can significantly increase camera FOV, it may at the same time distort the image. As a result, the computationally recovered image often has spatially uneven resolution, which is undesirable in many applications. Various configurations of mirrors have been proposed to produce images of uniform resolution(e.g., [64, 65, 66]).

Another important design goal for catadioptric cameras is to choose the proper shape of the mirrors so that the camera has a single effective viewpoint. A single viewpoint is important in many vision applications that require perspective input images [67]. Baker and Nayar [68] derive the complete class of single-lens, single-mirror catadioptic cameras that have a single viewpoint. Hicks [69] proposes a technique of designing a mirror to realize any given projection in a numerical approximation manner.

4) Other Applications: Spatially variant intensity modulators can also be used in object side coding to capture more useful visual information. Schechner and Nayar [33] propose to enhance the capabilities of traditional cameras by using spatially variant intensity modulators in front of the camera. The key observation is that each scene point will be captured multiple times as a camera moves. Due to the spatially varying properties of the filter, multiple measurements are obtained at different optical settings. Registering and fusing these measurements produces a high dynamic range and multiple-spectral image mosaic.

Talvala et al. [70] capture multiple images by shifting a structured occlusion mask in front of the camera in order to remove veiling glaring. They show that the proposed technique yields much higher signal-to-noise ratio than the traditional convolution-based technique. Kuthirummal and Nayar [71] use a radial catadioptric camera design for wide FOV imaging, estimating depth map, and measuring bidirectional reflectance distribution functions (or BRDFs) of surfaces. Du et al. [72] use the dispersion of prisms for multi-spectral imaging.

B. Homogeneous Filter

Relatively fewer object side coding techniques use spatially homogeneous optical devices. Umeyama and Godin [73] and Nayar et al. [74] propose capturing images with different polarization directions in order to remove specular reflections. Lin and Lee [75] use polarization to detect specularity. Zhou et al. [31] place a piece of optical diffuser on the object side and capture images for depth estimation. By doing so, one can achieve large angle triangulation without using a large imaging system for producing high precision depth maps. Raskar et al. [76] use an external shutter for coded exposure in order to preserve more information in the case of motion blurring. Although the shutter can be placed almost at any layer in the camera, object side coding is the most convenient way to do so. Rouf et al. [77] use a star filter mounted in front of a cameras to encode the visual information for saturated areas and then use computation to recover high dynamic range images.

IV. PUPIL PLANE CODING

Pupil plane coding places optical element at or close to the pupil plane of a traditional lens. Unlike object side coding, its effect on imaging is spatially invariant. Pupil plane coding therefore is often used to modulate the Point-Spread-Function (PSF) of an imaging system. In Fourier optics (incoherent light), the relation between pupil plane coding and the resulting PSF can be simply described by a Fresnel transform [28, 17]:

$$f(x) = |\mathcal{F}(W(x) \cdot Q_d(x))|^2, \tag{1}$$

where $f(x)$ is the PSF function, $\mathcal{F}(\cdot)$ is the Fourier transform, $W(x)$ is the aperture coding function, which can be complex for wavefront coding or real for coded aperture techniques, and $Q_d(x)$ is a quadratic phase term determined by focus. The captured image $i(x)$ is the convolution of the latent focused image $i_0(x)$ and the PSF $f(x)$, $i(x) = i_0(x) \otimes f(x)$. The power spectrum of PSF $f(x)$ is often referred to as Modulation Transfer Function (MTF) and is used to measure the optical quality of imaging systems. Equation 1 holds as long as the F-number is not extremely small and the field angle is not too large [17], and works for most cameras in the fields of computer vision and graphics.

When PSFs are known, one can use them to deconvolve captured images and recover high-quality sharp images. Conceptually, image deconvolution is dividing the Fourier transform of the blurred image by the Fourier transform of the PSF ([78, 79]). For better deconvolution results, MTFs should be broadband and have fewer zero-crossing frequencies. Zhou and Nayar [36] formulate how MTFs together with image priors and noise models affect the deconvolution quality.

Also, when PSFs are known, one can further compute the depth, because PSFs are determined not only by aperture coding which is usually known, but also by object depths. In particular, when $W(x)$ is real (in traditional cameras with or without coded apertures), this depth estimation is often referred to as depth from defocus (DFD) or focus (DFF).

Obviously, an accurate PSF estimation is critical for both image deconvolution and depth estimation. PSF estimation from a single image $i(x) = i_0(x) \otimes f(x)$ has to solve two unknowns from a single equation and therefore is usually ill-posed. Various prior knowledge about latent images or PSFs have been used to better constraint the ill-posed problem [34]. To obtain more reliable and precise PSF estimation, multiple images with different pupil plane coding are often required (e.g., [80, 81, 82, 83, 84]).

Extended depth of field (EDOF) is another promising solution to recover sharp images without knowing the depth. EDOF cameras are designed such that PSFs are depth-invariant and can be precomputed or calibrated. Therefore, sharp images can be recovered directly by deconvolving the captured images with a single known PSF. However, since PSFs of EDOF cameras are independent of depth, it cannot be used to compute depth - depth information is lost during capturing.

A. Coded Aperture

Pupil plane coding based on intensity modulators (i.e. $W(x)$ is positive and real) are often referred to as coded aperture techniques or sometimes apodizer techniques in optics. Coded apertures can only affect the information in the spatial dimensions (u, v in the $uvst$ space). That is, when diffraction and optical aberration are negligible, the shape of the PSF is simply determined by the aperture pattern, and the scale is determined by the amount of defocus.

1) Coded Aperture for Defocus Deblurring: An aperture pattern good for defocus deblurring should be broadband and have no zero-crossing frequencies in the Fourier domain. Early work in optics has proposed using coded apertures (e.g., [85, 86]) to preserve more high frequency information in the case of defocus. In astronomy, optimized patterns like the Modified Uniformly Redundant Array (MURA) are often used for lensless imaging [87, 88] in order to improve the signal-to-noise ratio of the captured images.

Partly inspired by the work in optics and astronomy, researchers in computer vision make use of computational power

to optimize coded aperture patterns. Veeraraghavan et al. [35] and Zhou and Nayar [36] optimize aperture patterns so that more image information can be preserved in the case of defocus deblurring. Zhou and Nayar [36] further study the effects of image priors and image noise models in optimizing the pattern. They show that the optimal coded aperture patterns are different at different noise level. When the noise level is extremely high, a wide open aperture is optimal; and when the noise level decreases, the optimal pattern becomes more unstructured.

2) Coded Aperture for Depth from Defocus: Traditional depth-from-defocus (DFD) techniques use circular aperture cameras. They capture multiple images at different focus settings or aperture settings for depth estimation (e.g., [80, 81, 82, 83, 84]). Levin et al. [34], Zhou et al. [89] and Levin [90] investigate how coded aperture affects the performance of depth from defocus and accordingly optimize aperture patterns for these tasks.

The basic observation is that in a single image scenario, a good aperture pattern for depth estimation should have obvious zero-crossing frequencies in the Fourier domain to help identify the PSF scale; and for multiple image scenario, the power spectra and phase of aperture patterns should compensate each other in the Fourier domain. They have shown that cameras with the optimized coded apertures can achieve much more reliable and precise depth map from defocus than with traditional circular apertures.

Stereo vs. Depth from Defocus Each stripe illustrated in Figure 6 represents the light from a single object point. The stripe slope is the inverse of the point to the reference plane. Both stereo vision and depth from defocus are about finding the slope of each stripe in the $uvst$ space. Binocular stereo estimates the slopes by taking two or more samples from each stripe as shown in Figure 9 (a). Stereo matching algorithms (See [91, 92, 93] for surveys) find the correspondence between each pair of samples and then compute the depth. Here, we have a slope $= d'/b$, where b is the baseline between the stereo camera, and d' is the disparity normalized by the focal length of the camera.

Instead of capturing two discrete samples, depth from defocus techniques capture two different integrals along each stripe and then use them to estimate the slope. Schechner and Kiryati [84] suggest capturing two images by shifting the pinhole. In this setting, the light fields after the aperture are two vertical slices (Figure 9 (b) Left), which are the same as in binocular stereo. A typical depth from defocus captures two images by changing the aperture size ([80, 81, 82, 83, 84]). As shown in Figure 9 (a), each image captures a different integral along each stripe. From two integrals, depth from defocus algorithm can be used to estimate the blur size, which is equivalent to the disparity in stereo vision. Then, the slope of each stripe can be computed as $slope = d'/r_2$, where d' is the radius of the blur kernel normalized by the focal length of the camera lens, and r_2 is the diameter of the aperture.

3) Other Coded Aperture Techniques: Liang et al. [37] propose using a programmable aperture camera to capture light fields. Bando et al. [45] designs a color-filter aperture so that three color channels will capture images with different aperture

(a) Binocular Stereo

Shifted small circulars   Large/small circulars   Coded pattern pair
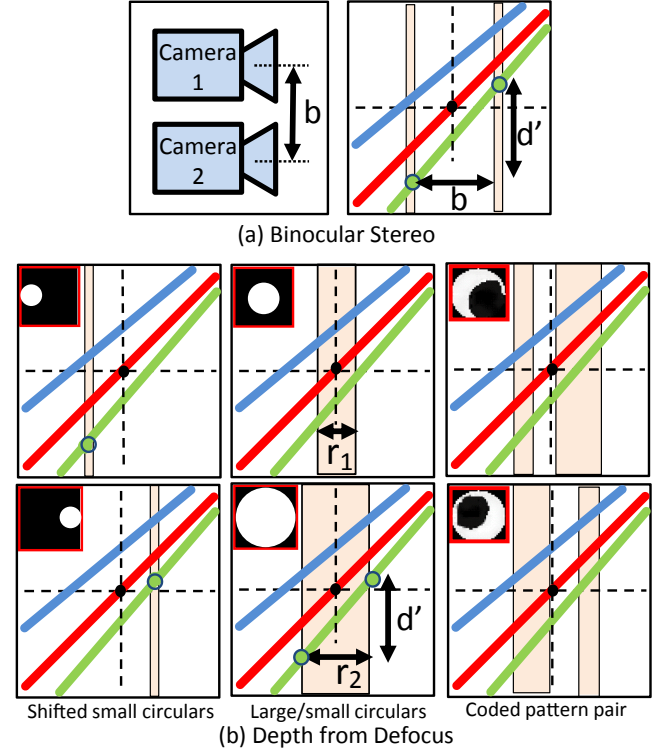(b) Depth from Defocus

Fig. 9.   Compare binocular stereo and depth from defocus. Depth estimation is about finding the slope of each stripe in the $uvst$ space. In binocular stereo (a), each camera captures one sample of each stripe. The two corresponding samples are used to compute the slope, which is the inverse of the depth. The correspondence between the two samples are found by using stereo matching algorithms. (b) A typical depth from defocus technique captures images by changing aperture size or pattern. Left: when two shifted pinhole apertures are used [84], each image captures one sample of each stripe. This is virtually the same as in stereo vision (a); middle: two images are captured by using two circular apertures of different sizes; right: two images are captured by using two coded aperture patterns [89]. Each depth from defocus algorithm captures two different integrals to estimate the stripe slope.

patterns. Then, depth information can be estimated from the three color channels in a single image for matting.

Coded aperture techniques usually cannot be used to extend depth of field, because in geometrical optics, the scale of the PSF varies with depth in an ideal camera. PSFs can be made relatively invariant to depth if diffraction effect can be properly made use of. For example, a zone plate - a circular mask with annular rings of specific radii, can produce multiple focus depths in an imaging system and can be used to extend depth of field (EDOF) [94, 95].

4) Coded Aperture Implementation: Coded apertures can be implemented in several different ways. Levin et al. [34] and Veeraraghavan et al. [35] insert cut paper boards into lenses. Zhou et al. [89][36] print coded patterns onto precision photomasks. Liang et al. [37] use an LCD and a series of rolling paper boards to implement a number of fast switching aperture patterns. Nagahara et al. [38] design a relay optics with an LCoS device for programmable aperture imaging. This programmable aperture camera design can be easily attached to different lenses, and afford high frame rate aperture switching and relatively high light efficiency. Green et al. [96] and Aggarwal and Ahuja [97] use specially designed and arranged mirrors to split an aperture into several parts of different shapes.

(a) Multi-aperture photography using tilted mirrors

(b) Coded aperture using photomasks

A Scroll of Paper Patterns   Liquid Crystal Array

3cm

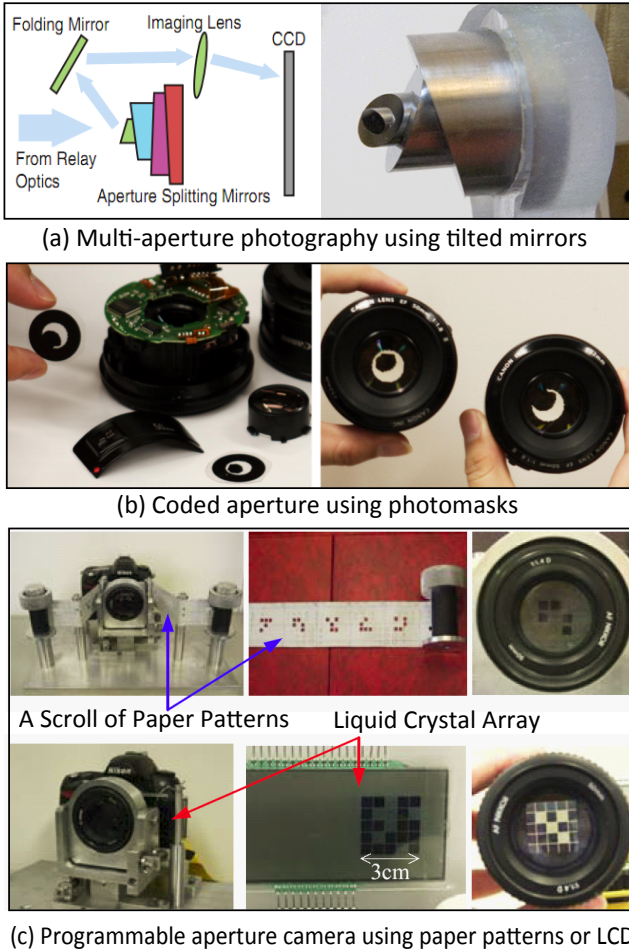(c) Programmable aperture camera using paper patterns or LCD

Fig. 10. Coded aperture implementations. (a) Use a 4-way aperture-splitting mirror to divide the lens aperture [96]. (b) Insert photomasks into lenses [36]. (c) Use a scroll of paper patterns or an LCD right behind of the lens [37].

B. Wavefront Coding

Pupil plane coding using phase modulators (i.e. $W(x)$ is complex) is often referred to as wavefront coding. A phase modulator is usually a plate of glass of certain 3D profile. As illustrated in Figure 5(d), a phase plate will distort the input light field in s dimension, and the resulting PSF will simply be the histogram of the derivation of the wavefront function $W(x)$. In wave optics, this relation can be formulated using Equation 1. Wavefront coding techniques have been studied for decades in optics for a variety of applications.

Wavefront coding for depth estimation prefers using PSFs that change dramatically with depth. Dowski [98] design a phase plate that has responses at only a few frequencies, which makes the imaging system more sensitive to depth variations. Greengard et al. [99] design a phase plate that can produce PSFs which will rotate as depth varies. This rotational PSF design is shown to effectively increase depth variations of PSFs and therefore is more optimal for depth estimation.

More often, wavefront coding is used to extend depth of field and a lot of work has been done in this line. As mentioned above, the PSF of a good EDOF technique should be a broadband filter without zero-crossing frequencies in the Fourier domain and should, at the same time, be depth invariant. Cathey and Dowski [100] [101] propose a cubic phase plate
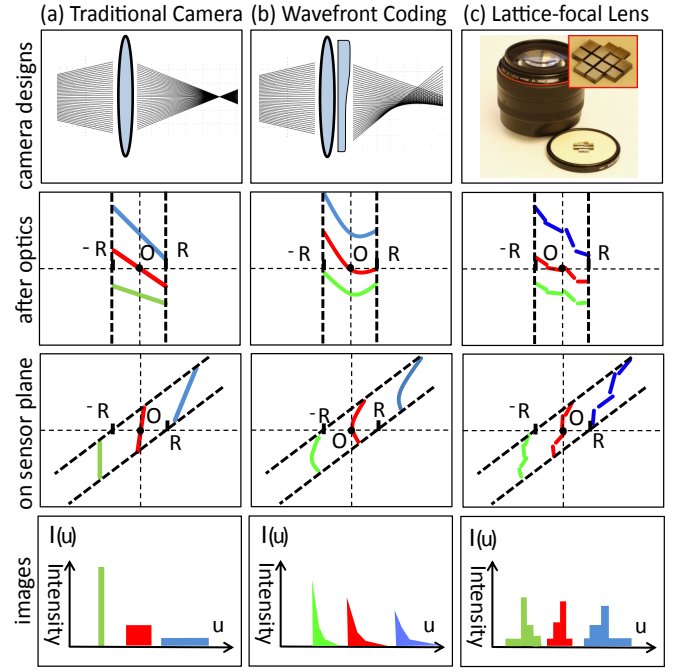

Fig. 11. Light field transforms in EDOF cameras. An ideal EDOF camera produces PSFs that are broadband in the Fourier domain (or sharp in the spatial domain) and invariant to depth. From (a) to (c) are a traditional camera, a cubic phase plate EDOF camera [100, 98], and a lattice-focal lens camera [104], respectively. From the second row to the fourth row, we show light fields after the optics, light fields on the sensor plane, and the final captured images, respectively. We can see that the cubic phase plate EDOF camera and the lattice-focal lens camera produce sharp PSFs at various depths, while PSFs of traditional cameras become flat quickly when objects move away from the focal plane.

design which yields a broad-band spectrum and is relatively depth-invariant. One way to understand its property of depth invariance is that when $W(x)$ is a 3rd order polynomial function, it will overpower the quadratic defocus term Q_d and therefore the system appears to be depth-invariant.

The performance of the cubic phase plate has been analyzed extensively from the perspectives of light fields, wave optics, and frequency domain [102, 103, 104]. A intuitive understanding of the cubic phase plate EDOF camera can be found by using a light field representation. In Figure 11, we illustrate two light fields in a cubic phaseplate camera (b), after the optics and on the sensor plane, and compare them with that in a traditional camera (a). In the traditional camera, objects at different depths yield straight lines of different slopes, and then produce pillbox PSFs of different radii when projected onto u axis. Meanwhile, in a cubic phase plate camera, objects at different depths yield 2nd order polynomial curves in the $uvst$ space. When the curves on the sensor plane are projected onto u axis, the intensity distribution is proportional to the curve slopes and has a peak at the vertical regions. The resulting PSFs have sharp peaks that help to preserve high frequency information. Note that the wavefront error function of spherical aberration and chromatic aberration are also 2nd order polynomials, same as defocus errors. Therefore, the cubic phase plate can also be used to suppress these two aberrations [105].

Many other phase plate designs have also been proposed for EDOF imaging. George and Chi [106] propose the use of a

(a) The geometry and a prototype of heterodyne light field camera
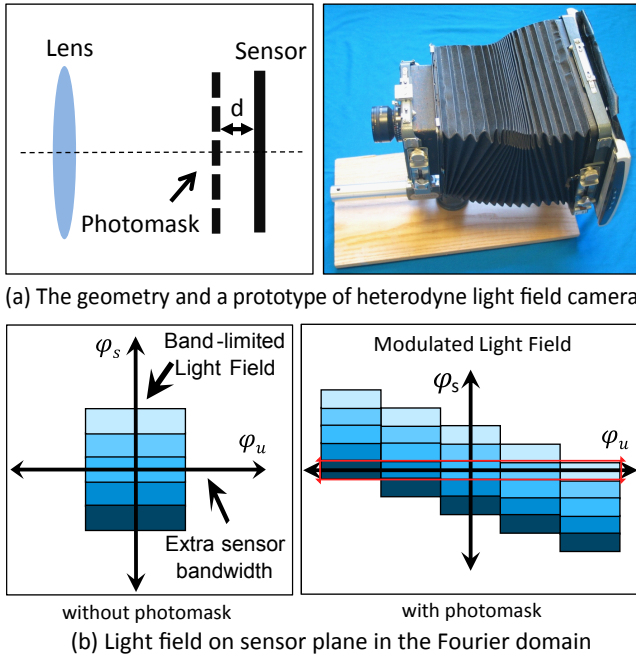

(b) Light field on sensor plane in the Fourier domain

Fig. 12. A heterodyne light field camera using a photomask between sensor and lens [35]. (a) The geometry of the heterodyne camera, and its prototype implementation. (b) Left: an illustration of the band-limited light field in a traditional camera (Fourier domain). According to the Fourier slice theory [109], the camera sensor captures a slice of the light field ($\Phi_s = 0$). Right: an illustration of the light field in the Fourier domain in a heterodyne camera (Fourier domain). This heterodyned light field contains multiple copies of the band-limited light field. Therefore, 1D slice of the heterodyned light field contains 2D light field information

logarithmic asphere and image processing to increase the depth of field. Castro and Ojeda-Castañeda [107] present a family of asymmetric phase masks that extends the depth of field of an optical system. Cossairt et al. [32] and Garcia-Guerrero et al. [108] propose using radially symmetric optical diffusers to extend the depth of field. Levin et al. [104] propose using a focal lattice to extend depth of field. The light field illustration of this technique is shown in Figure 11(c). The authors prove that the performance of the lattice focus design is close to the theoretical upper-bound.

V. SENSOR SIDE CODING

Here, an optical element is placed on the sensor side of the lens. The element can be either placed in the space between the sensor and the lens, or placed on or close to the sensor, but their functionalities will be different. In this category, we also include the use of small physical motions of the image sensor or pixel-wise control of exposure. According to the Gaussian lens law, optical devices after the lens are dual to devices in front of the lens, and therefore sensor side coding can provide similar functionalities as object side coding – it can modulate light fields in both the u and s dimensions. One important advantage of using sensor side coding instead of object side coding is that it can be compactly built into a camera and hence is non-intrusive to the scene.

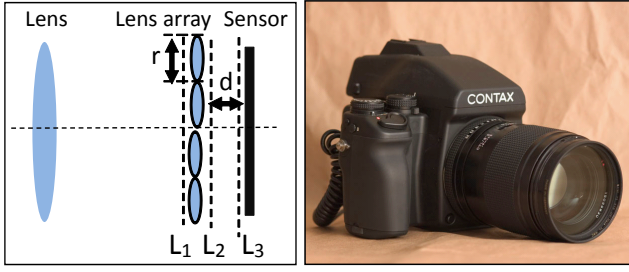### A. Coding in front of the sensor plane

Consider putting an optical element between Plane L_4 and L_5 in Figure 6. The light field will be first modulated by this

element, then sheared in the u dimension due to the space between the element and the sensor, and finally projected onto the sensor. When the element is an intensity modulator M, we have $I = \mathcal{P}[\mathcal{S}[L \cdot M]] = \mathcal{P}[\mathcal{S}[L] \cdot \mathcal{S}[M]] = \mathcal{P}[L_5 \cdot \mathcal{S}[M]]$, where $\mathcal{P}$ is the projection of the high dimensional light field onto the 2D sensor, $\mathcal{S}$ is the shearing operator on light fields, L is the light field right in front of the intensity modulator and L_5 is the light field on sensor plane as if the intensity modulator were removed.
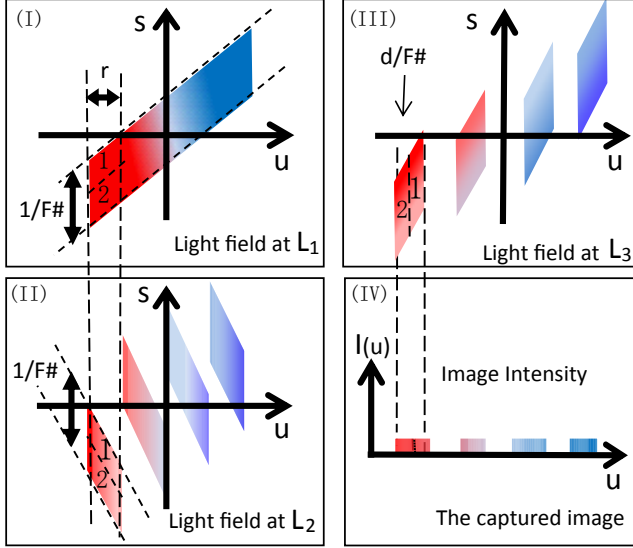
In the Fourier domain, the dot-product in the equation becomes convolution. According to the Fourier slice theory [109], the captured image is one slice of the light field in the Fourier domain. Veeraraghavan et al. [35] propose using a mask M which is the sum of several cosine signals in spatial domain (or the sum of several Dirac delta function in the Fourier domain) to modulate the light field so that the Fourier transform of the final light field will have several identical copies of $\tilde{L}_5$ (see Figure 12(d)). Since the copies are made in a tilted angle, different slices of the light field will be captured by the sensor and then be used to reconstruct the 4D light field. One important assumption in this technique is that the input light field is a band-limited one, otherwise multiple copies of the light field will overlap each other and cause severe aliasing. Raskar et al. [110] further study the glare effect and propose using the same setting as in [35] to reduce glare effects of camera lenses.

As in object side coding [52], lens arrays can also be used on the sensor side to capture light fields. The idea of the plenoptic camera has a long history since the early twentieth century [113, 114]. Since the 1990s, a variety of plenoptic cameras have been proposed and implemented in vision and graphics. Adelson and Bergen [7] proposes using a lenslet array in front of the sensor for light field acquisition. Ng et al. [111] give more detailed analysis on this approach and implement a prototype compact, hand-held light field camera, as shown in Figure 13 (a). Figure 13 (b) shows how the lens array transforms the light field and helps to capture the 4D light field. Consider the two small labeled blocks in the light field at L_1. The two blocks would be integrated into a single pixel if a sensor were placed in Plane L_1. The light field is sheared by the lenslet in the s dimension (L_2), and then sheared in the u dimension due to the space between the lens array and the sensor (L_3). In the light field on Plane L_3 (Figure 13(b)III), the two blocks are now parallel vertically and can be mapped to different pixels. As a result, a 1D sensor can capture a 2D light field. Similarly, a 2D sensor can capture a 4D light field.

The position, size, and focus length of the lenslet array have to be chosen carefully. To avoid overlapping or waste pixels, the optimal setting is to have $r = d/F\#$, which indicates that the F-number of the main lens should be identical to the F-number of the lenslets. Also, note that the 4D light fields are captured by sacrificing the spatial resolution (u) for angular resolution (s). Therefore, there is a trade-off between spatial and angular resolution in plenoptic light field camera designs. To achieve different amount of trade-offs, Lumsdaine and Georgiev [115] and Bishop et al. [116] propose several different strategies of positioning lenslets and sensors.

(a) The geometry and a prototype of plenoptic camera

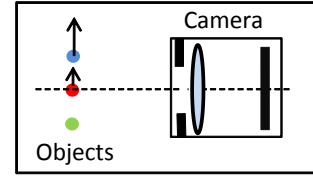(b) Light field transform at three planes and the final image

Fig. 13. Plenoptic light field camera using a lens array between sensor and lens [111]. (a) The geometry of the plenoptic camera, and its prototype implementation. (b) From (I) to (III) are the intermediate light fields on L_1 (before the lensarray), L_2 (after the lensarray), and L_3 (before the sensor), respectively. The image sensor captures the projection of the light field (L_3) on the u axis, as shown in (IV). We can see that the incoming 2D light field (I) is rearranged in such a way (as shown in (III)) that it can be projected onto a 1D image sensor.

B. Coding on the Sensor plane

Coding on the sensor plane usually does not affect the signal in the angular dimension, but provides pixel-wise modulations. Color filter arrays, such as the Bayer mode array, are widely used here to encode color information in a monochromatic sensor [117, 43]. Other color filter patterns have also been proposed [44, 118], and various demosaicing algorithms have also been used to a high quality color images [119, 120].

Various intensity modulators (e.g., color filter, neutral filter, polarizer) can be used on the sensor plane to capture different visual information. Nayar and Narasimhan [121] generalize the color filter array to assorted filter arrays in order to capture extra multi-spectral and high dynamic range information. Ben-Ezra et al. [122] suggest modifying the shape and layout of pixels for super-resolution. Their proposed penrose layout is shown to significantly improve the factor of super-resolution.

In the past decades, tremendous advances on sensor design has been achieved to reduce pixel size, increase light efficiency, extend dynamic range, suppress noise, and etc [123, 124, 125, 126, 127]. Computational sensor is another overlapping research area, where people are developing detectors that can perform image sensing as well as early vision

(a) An imaging scene with three moving objects

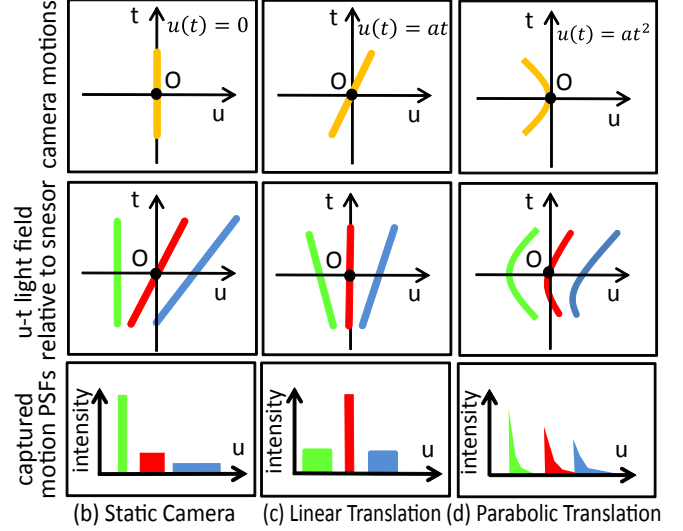(b) Static Camera (c) Linear Translation (d) Parabolic Translation

Fig. 14. Motion invariant photography [112]. (a) An imaging scene with three objects of different velocities. From (b) to (d), we illustrate how sensor motions affect the motion point-spread-functions (PSFs) in three cameras, including a static camera, a camera with a linear translation, and a camera with a parabolic translation, respectively. Each figure includes, from top to bottom, the function of camera motion, the light field relative to the moving sensor in the $u - t$ space, and the captured PSFs, respectively. We can see that a camera with a parabolic translation yields sharp motion PSFs and is relatively invariant to the velocity of object motions.

processing [128, 129, 130]. We will not be able to cover these approaches in this survey.

C. Sensor Motion

Another type of important sensor plane coding is to use camera motions. During the exposure time when an image is captured, one can move the sensor in various directions and patterns to help preserve useful information. Nagahara et al. [131] propose a focal sweep technique to extend depth of field, which moves the sensor along optical axis during the exposure time. They have also demonstrated using discontinuous motions to achieve discontinuous depth of field, and using motions with rolling shutter sensors to achieve curved depth of field. A similar focal sweep idea has been exploited in the area of microscope by moving the specimen instead of the sensor [132]. Sensor motion can also be made on the sensor plane. Lenz and Lenz [133] and Ben-Ezra et al. [134] [135] propose capturing multiple images with subpixel sensor motions for higher resolution images.

Sensor motion techniques can also be used to reduce or remove motion blur. Analogous to the cubic phase plate idea for EDOF imaging, Levin et al. [112] propose moving the sensor in a parabolic trajectory so that the motion-blur PSF will be invariant to object motion and invertible. The captured images can then be deconvolved with a single PSF to achieve blur-free images. In Figure 14, we illustrate the

spatial-temporal light field representations of three cameras: a traditional camera, a linear translation camera [76], and the parabolic translation camera [112]. We can see that object motions of different speed lead to rectangle PSFs of different scales in a traditional camera or a linear translation camera; and in the motion invariant camera, the resulting PSFs are sharp and almost invariant to the velocity of the moving objects.

One limitation of this method [112] is that it is designed to deal with motions in one specific direction. Cho et al. [136] propose capturing two images by using parabolic motions in two orthogonal directions in order to estimate motion and remove motion blurs in arbitrary directions. Rav-Acha and Peleg [137] show that deconvolution would be more robust if multiple images with different motion direction could be captured.

VI. ILLUMINATION CODING

The basic function of the camera flash has remained the same since it first became commercially available in the 1930s. It is used to brightly illuminate scene inside the camera FOV during the exposure time of the image detector. It essentially serves as a point light source. With significant advances made with respect to digital projectors, the flash now plays a more sophisticated role in capturing images. It enables the camera to project arbitrarily complex illumination patterns onto the scene, capture the corresponding images, and extract scene information that is not possible to obtain with a traditional flash.

In this case, the complete imaging system can still be thought of as a computational camera where captured images are optically coded due to the patterned illumination of the scene (Figure 2(d)). Figure 15 shows two computational cameras with built-in coded illumination elements. Kinect depth sensor, a Microsoft product for gaming released in 2010, combines an infrared projector with a monochrome CMOS sensor for 3D reconstruction [138]; and Nikon Coolpix camera, which was released in 2009, provides a built-in projector.

Structured light can be regarded as a spatial illumination coding technique that has a long history in the field of computer vision (see [3] for a survey). This technique improves the performance of stereo vision by making the correspondence matching problem easier to solve. Consider an input light field as shown in Figure 9(a). Stereo matching algorithms need to find two samples in each stripe in order to compute the slope – the inverse of the depth. Structured light techniques make each stripe look different (either in intensity, color, or local pattern) and therefore make the correspondence matching an easier problem.

Many other illumination coding techniques for depth estimation or 3D reconstruction have been proposed in recent years. Zhang and Nayar [139] and Gupta et al. [140] recover depth from defocused projections; Kirmani et al. [141] measure the depths of points outside the camera's field of view by using echoes of pulsed illumination; Raskar et al. [142] use multiple flashes for depth edge measurement; Ma et al. [143] estimate specular and diffuse normals by using

(a) Microsft Kinect depth sensor fox Xbox game console

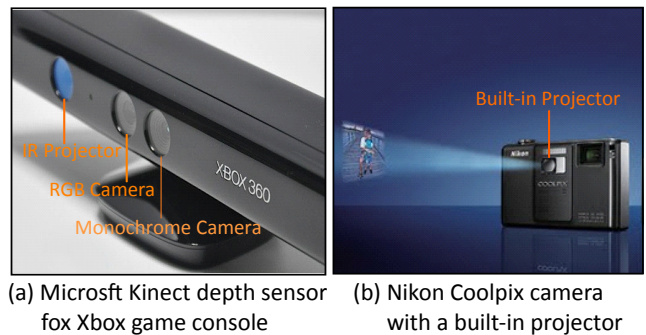(b) Nikon Coolpix camera with a built-in projector

Fig. 15.   Two computational cameras with built-in components of illumination coding. (a) A Microsoft Kinect depth sensor uses an infrared projector and a monochrome CMOS sensor for 3D reconstruction. [138] (b) A Nikon Coolpix camera with a built-in projector.

gradient illumination; and Zhang et al. [144] produce robust 3D reconstruction by using space-time stereo.

Structured illumination has also been used for a variety of other vision and graphics tasks, including the separation of direct and global illumination [145], image enhancement using flash and no-flash images [146, 147, 148], BRDF invariant surface reconstruction using Helmholtz stereopsis [149], and the measurement of light transport in a scene [150, 151, 152, 153]. Multiplexed illumination is proposed for improving SNR in the case of weak sources [154], object relighting [155], and multi-spectral imaging [156].

Structured illumination techniques based on a phenomenon known as the Moire effect have been used to overcome the resolution limits of microscopy [157, 158] and other imaging systems [159, 160] (see [161] for a survey of the Moire technique). Structured illumination using diffuse optical tomography has been used for volume density estimation [162, 163].

VII. CAMERA CLUSTERS OR ARRAYS

Cameras use a set of devices to process the input light field in order to capture useful information. The capability of a single camera is virtually constrained by optical size, which physically determines the field of light to be captured. One way to transcend this limit is by using larger lenses. However, it is often too expensive and difficult to built large imaging systems of high quality. In recent years, techniques have been proposed to use a number of low cost small cameras to capture more visual information.

A. Camera Arrays

Camera arrays have been used for stereo vision for a long history. Multi-view stereo helps to solve the ambiguity problem in stereo matching and hence increases the precision of depth estimation [166, 167, 168, 169, 170]. Ding et al. [171] use a $3x3$ camera array to track distorted feature points beneath a fluid surface in order to dynamically recover fluid surfaces. Yang et al. [172] propose a real-time distributed light field camera of a 8x8 video camera array to capture dynamically changing light field and allow multiple users to navigate virtual cameras. The rendering algorithm is distributed in order to overcome the data bandwidth problem. In [165], an array of

(a) Camera array for light field acquisition

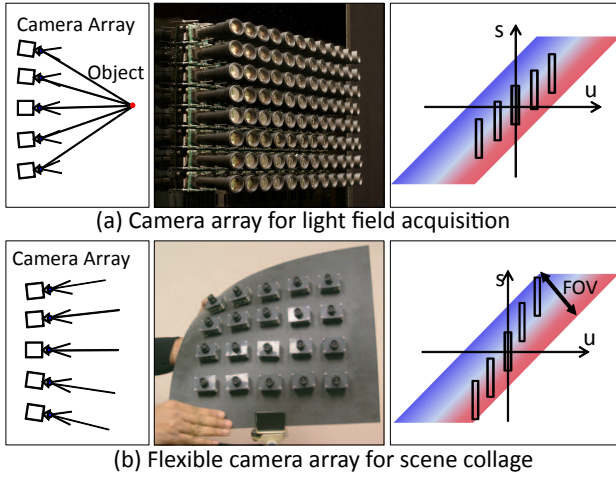(b) Flexible camera array for scene collage

Fig. 16. Camera array. (a) A regular array of cameras with overlapping field of view for light field acquisition [164]. From left to right are the camera geometry, a prototype camera array, and its light field illustration. (b) A flexible array of cameras with divergent field of view for scene collage [165]. From left to right are the geometry, a prototype camera array, and its light field illustration.

(a) A single element design for a gigapixel camera

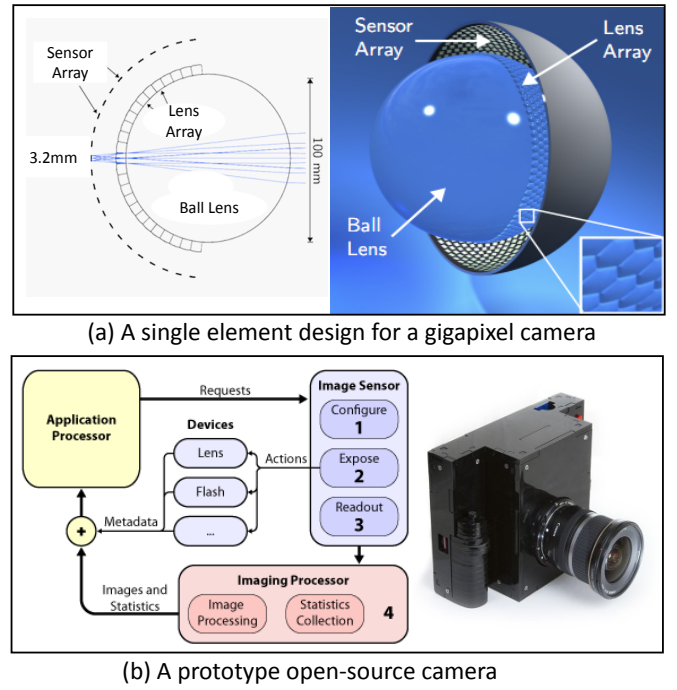(b) A prototype open-source camera

Fig. 17. Two unconventional computational camera designs. (a) A single element design using a ball lens for gigapixel imaging [183]. This design uses computation to achieve high resolution while reducing lens complexity and camera size. (b) The Frankencamera provides an open-source experimental platform for computational photography [184].

video cameras are used to stabilize the video jittering due to camera shaking. The key idea is to interpolate one smooth camera trajectory from 2x2 unstable cameras.

The high performance of camera arrays in HDR, FOV, Synthetic aperture, and high frame rate capturing has been studied in [164]. For example, as illustrated in Figure 16 (a), when all the cameras have an overlapping FOV, each of them captures one slice of the 4D light field at a specific (s, t) point. Then, 4D light fields can be reconstructed by combining all the captured images. When FOVs are divergent as shown in Figure 16 (b), the camera array will be able to capture high resolution and wide FOV images [173].

B. Camera Clusters

Camera clusters without regular layouts are often used in graphics for image-based rendering (see [174] for a review) and panoramic imaging (see [175] for a review). Ideally, one wants images of an object or scene to be captured from all possible perspectives. Image-based rendering techniques in graphics use a finite number of cameras to capture multiple images from different perspectives and then use view interpolation algorithms to synthesize those un-sampled pixels. The camera layout in the cluster has been studied extensively in the 1990s. The basic idea is that more cameras should be located from perspectives where the scene complexity is high [176, 177].

Shum and He [178] and Chai et al. [179] propose using a number of cameras mounted on a rotating holder to uniformly sample a light field for concentric mosaicking. This technique is then further analyzed with respect to non-Lambertian surface, occlusions, and sampling strategies [180, 181]. Inward and outward camera clusters have been widely used for virtual reality [182].

VIII. UNCONVENTIONAL IMAGING SYSTEMS

There are several other computational camera designs that cannot fit well in the previous five categories. Work has been done to simplify cameras by using computations instead of extending the functionalities of the camera. Stork and Robinson [185] and Robinson and Stork [186] discuss several mathematical and conceptual foundations for digital-optical joint optimization, and propose a singlet lens design and a triplet lens design with improved image quality after computation. Robinson and Stork [187] exploit the idea of digital and optical joint optimization for super-resolution.

A number of techniques have been proposed to make use of axial chromatic aberrations for extended depth of field imaging. These techniques only require lenses as simple as singlet, doublet, or triplet, and achieve much larger depth of fields for grayscale scenes [188, 189, 190, 191]. Guichard et al. [192] also extend DOF by exploiting chromatic aberrations, but can deal with color images by finding the best focused color channel and transferring high frequency information to the remaining color channels. Cossairt and Nayar [191] show that for a system with axial chromatic aberrations, even the best focused color channel is blurred and this limits the quality of the recovered images. They therefore propose an SFS technique, which creates an approximately depth-invariant PSF, and use deconvolution to recover more scene details.

It is also possible to change the overall architecture of cameras. Zomet and Nayar [193] propose lensless cameras with one or multiple layers of controllable apertures for imaging. Although this design requires a larger video detector and loses some light, it yields more flexible field of view than traditional cameras. Tumblin et al. [194] conceptually propose a gradient camera that measures image gradients instead of intensities and demonstrates its advantage in high dynamic range imaging by simulation.

While most imaging techniques are interested in capturing perspective images as if through one pinhole, multiperspective imaging combines what is seen from several viewpoints into a single image and is potentially advantageous to understand the structure of certain scenes [195, 196]. For example, an XSlit camera [197] collects all rays that pass through two non-coplanar lines. Yu and McMillan [198] present a General Linear Camera (GLC) model that unifies many multiperspective cameras and reveal three new and previously unexplored multiperspective linear cameras by using the GLC model. See [199] for a survey on multiperspective modeling, rendering, and imaging.

High-resolution cameras require using large lenses to overcome the diffraction limits but are fundamentally limited by geometrical aberrations. In face of this fundamental limit, Ben-Ezra [200] and Gigapixel Project [201] propose large format sensors and scanning strategies for gigapixel imaging; Marks and Brady [202] and Brady and Hagen [203] increase complexity of lens design as the lens is scaled up; and Cossairt et al. [183] propose a shared ball lens architecture as shown in Figure 17 (a) and use computation to achieve high resolution while reducing lens complexity and camera size.

Adams et al. [184] propose FrankenCamera, an open-source experimental platform, for computational photography. Although the FrankenCamera design does not modify the optics directly, it makes the control of the optical devices much easier in experiments. Non-optical devices can also be used with cameras to capture extra information for image processing. Park et al. [204] propose using a 3-axis accelerometer to measure camera motion and then use the motion information for motion-blur deblurring; and Joshi et al. [205] combine gyroscopes and accelerometers for more precise motion measurement and better motion deblurring results.

IX. Discussion

Among the six coding approaches, object side coding, pupil plane coding, and sensor side coding are shown as modifications made to a traditional camera. Figure 18 gives an overview of the computational camera designs in these three categories. In the horizontal direction, we have object side coding, pupil plane coding, and sensor side coding. In the vertical direction, we have phase modulators (including lens, lens array, prisms, prism array, plate, phaseplate, and diffuser), intensity modulators (including masks, color filters, and polarizers), and others (including mirrors and motions). In each cell, we group the techniques according to the type of visual information that they want to capture, including light field, depth, image (i.e. the spatial resolution), EDOF, HDR, color, FOV, and motion (i.e. the temporal resolution). This table, although not exhaustive, provides an overview of existing computational camera designs and may inspire new ideas in this exciting research area.

We made several observations from this table. First, techniques in the same cell usually have a lot of similarities in their formulation and design principles, and are often different from each other by choosing different performance trade-offs or optimization strategies. For example, several sensor side

coding techniques have been proposed to use lens arrays for light field acquisition and these techniques achieve different spatial and angular resolution trade-offs; various masks in pupil plane coding have been used to improve the spatial resolution and they are differently designed mainly due to the different criteria of image quality measurement. Understanding the similarity and difference can help one to know computational cameras techniques better.

Second, the camera designs are distributed in the table in a non-uniform manner. This is first related to the fact that certain combinations of devices and coding approaches are good or not good at preserving certain visual information. For example, people usually do not use phaseplates on the sensor plane, because it does not affect the final captured images. The non-uniform distribution can also be a result of other factors, including market needs and implementation difficulties. For example, research in catadioptic camera design is greatly driven by the need of wide field of view imaging in autonomous navigation, virtual reality, and video conferencing; the popularity of wavefront coding techniques in the research community of vision and graphics is greatly limited by the difficulty in implementation.

There are many holes in the table. Some of them may be worthy of exploiting. For example, there is no much work using phaseplates for object side coding. The missing of this type of technique is probably because of the difficulties in implementing phaseplates and analyzing the effects of an object-side phaseplate. Overcoming these difficulties may lead to new useful techniques.

While the first three coding approaches are used to extract more useful information from the incoming light field, the latter three coding approaches (illumination coding, camera array or cluster, and unconventional computational cameras) are more about transcending the fundamental limits of traditional cameras. Illumination coding techniques modify incoming light fields by actively projecting light. Camera array or clusters overcome the physical limit of camera size and are able to sample light fields at a larger scale. Unconventional camera research optimizes or may even revolt camera architectures by leveraging the power of computation.

Acknowledgments

References

[1] S. K. Nayar, "Computational cameras: Redefining the image," *Computer*, vol. 39, no. 8, pp. 30–38, 2006.

[2] ——, "Computational camera: Approaches, benefits and limits," *Columbia University, Computer Science Department*, 2011.

Devices		Object Side Coding	Pupil Plane Coding	Sensor Side Coding
Phase Modulators	**Lens(es)**	Lightfield: [52, 11]	Depth: [104]	Lightfield: [113,114,111,115,116,206]
	Prism(s) Plate(s)	Depth: [46,50,208,209] Color: [72]		
	Phaseplate		Depth: [98, 99] EDOF: [100,101,105,106,107,210]	
	Diffuser	Depth: [31] HDR: [77]	EDOF: [32,108]	
Intensity Modulators	**Photomask**	HDR: [33,70,215] Motion: [76]	Lightfield: [37,38] EDOF: [94,95] Depth: [34,80,81,82,83,84,89,90,217,218] Image:[34,35,36,85,86,87,88,211,212,213]	Lightfield: [35] HDR: [110,121,214]
	Color Filter	Color: [33]	Depth: [45]	Color: [43,116,117,119,120,121]
	Polarizer	Separation: [73,74,75,219,220]		
Others	**Motion**	EDOF: [132]	Depth: [221]	EDOF: [134] Image: [133,134,135] Motion: [76,112,136]
	Mirror(s)	Depth: [47,48,49,51,71,223,226] FOV: [53,54,55,56,57,58,59,60,64, 65,66,67,68,69,71,222,224,225]		HDR: [41,97,215] FOV: [41]
Color Scheme: Lightfield, Depth, Image, EDOF, HDR, Color, Separation, FOV, Motion				

Fig. 18.   An overview of computational camera designs using object side coding, pupil plane coding, and sensor side coding. In the vertical direction are the optical devices that are often used in designing computational cameras. In each cell, we group the techniques according to the type of visual information to be captured, including light field, depth, image (i.e. spatial resolution), EDOF, HDR, Color, FOV, and motion (i.e. temporal resolution). Each group is differently colored. This table, although not exhaustive, provides an overview of existing computational camera designs and may inspire new ideas in this area.

[3] J. Salvi, J. Pages, and J. Batlle, "Pattern codification strategies in structured light systems," *Pattern Recognition*, vol. 37, no. 4, pp. 827–849, 2004.

[4] J. Salvi, S. Fernandez, T. Pribanic, and X. Llado, "A state of the art in structured light patterns for surface profilometry," *Pattern Recognition*, vol. 43, no. 8, pp. 2666–2680, 2010.

[5] R. Woodham, "Photometric method for determining surface orientation from multiple images," *Optical engineering*, vol. 19, no. 1, pp. 139–144, 1980.

[6] M. Faraday, "Thoughts on ray-vibrations," *Philosophical Magazine*, vol. 28, no. 188, pp. 447–452, 1846.

[7] E. Adelson and J. Bergen, "The plenoptic function and the elements of early vision," *Computational models of visual processing*, vol. 1, 1991.

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

[9] S. Gortler, R. Grzeszczuk, R. Szeliski, and M. Cohen, "The lumigraph," in *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*. ACM, 1996, pp. 43–54.

[10] K. Halbach, "Matrix representation of Gaussian optics," *American Journal of Physics*, vol. 32, p. 90, 1964.

[11] T. Georgiev and C. Intwala, "Light field camera design for integral view photography," Adobe, Tech. Rep., 2006.

[12] P. Debevec and J. Malik, "Recovering high dynamic range radiance maps from photographs," in *SIGGRAPH*. ACM, 1997, pp. 1–10.

[13] G. Surya and M. Subbarao, "Depth from defocus by changing camera aperture: A spatial domain approach," in *IEEE Conference on Computer Vision and Pattern Recognition*, 1993, pp. 61–67.

[14] S. Hasinoff and K. Kutulakos, "Confocal stereo," in *European Conference on Computer Vision*. Springer, 2006, pp. 620–634.

[15] L. Yuan, J. Sun, L. Quan, and H. Shum, "Image deblurring with blurred/noisy image pairs," in *SIGGRAPH*. ACM, 2007, pp. 1–es.

[16] S. Hasinoff, K. Kutulakos, F. Durand, and W. Freeman, "Time-constrained photography," in *IEEE International Conference on Computer Vision*, 2009, pp. 333–340.

[17] J. Goodman, "Introduction to fourier optics, mcgaw-hill physical and quantum electronics series," 1968.

[18] J. Geary, *Introduction to lens design: with practical ZEMAX examples*. Willmann-Bell, 2002.

[19] M. Bastiaans, "Wigner distribution function and its application to first-order optics," *Journal of the Optical Society of America A*, vol. 69, no. 12, pp. 1710–1716, 1979.

[20] ——, "The Wigner distribution function applied to optical signals and systems," *Optics Communications*, vol. 25, no. 1, pp. 26–30, 1978.

[21] R. Castaneda, "Phase space representation of spatially partially coherent imaging," *Applied Optics*, vol. 47, no. 22, pp. E53–E62, 2008.

[22] S. Oh, G. Barbastathis, and R. Raskar, "Augmenting light field to model wave optics effects," *Arxiv preprint arXiv:0907.1545*, 2009.

[23] Z. Zhang and M. Levoy, "Wigner distributions and how they relate to the light field," in *International Conference on Computational Photography*, 2010, pp.

1–10.

[24] A. Kolb, E. Barth, R. Koch, and R. Larsen, "Time-of-flight cameras in computer graphics," in *Computer Graphics Forum*, vol. 29, no. 1. Wiley Online Library, 2010, pp. 141–159.

[25] M. Levoy, "Light fields and computational imaging," *Computer*, vol. 39, no. 8, pp. 46–55, 2006.

[26] R. Raskar, J. Tumblin, A. Mohan, A. Agrawal, and Y. Li, "Computational Photography," in *Eurographics State of the Art Report*, 2006, pp. 1–24.

[27] G. Wetzstein, I. Ihrke, D. Lanman, and W. Heidrich, "Computational Plenoptic Imaging," in *Eurographics State of the Art Report*, 2011, pp. 1–24.

[28] M. Born, E. Wolf, and A. Bhatia, *Principles of optics*. Pergamon Pr., 1975, vol. 10.

[29] A. Torre, *Linear ray and wave optics in phase space*. Elsevier, 2005.

[30] A. Gerrard and J. Burch, *Introduction to matrix methods in optics*. Dover Publications, 1994.

[31] C. Zhou, O. Cossairt, and S. Nayar, "Depth from Diffusion," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2010, pp. 1110–1117.

[32] O. Cossairt, C. Zhou, and S. Nayar, "Diffusion coded photography for extended depth of field," in *SIGGRAPH*. ACM, 2010, pp. 1–10.

[33] Y. Schechner and S. Nayar, "Generalized mosaicing," in *IEEE International Conference on Computer Vision*, vol. 1, 2001, pp. 17–24.

[34] A. Levin, R. Fergus, F. Durand, and W. Freeman, "Image and depth from a conventional camera with a coded aperture," *ACM Transactions on Graphics (TOG)*, vol. 26, no. 3, pp. 70–es, 2007.

[35] A. Veeraraghavan, R. Raskar, A. Agrawal, A. Mohan, and J. Tumblin, "Dappled photography: Mask enhanced cameras for heterodyned light fields and coded aperture refocusing," *ACM Transactions on Graphics (TOG)*, vol. 26, no. 3, p. 69, 2007.

[36] C. Zhou and S. Nayar, "What are good apertures for defocus deblurring?" in *International Conference on Computational Photography*, 2009, pp. 1–8.

[37] C. Liang, T. Lin, B. Wong, C. Liu, and H. Chen, "Programmable aperture photography: multiplexed light field acquisition," *ACM Transactions on Graphics (TOG)*, vol. 27, no. 3, 2008.

[38] H. Nagahara, C. Zhou, T. Watanabe, H. Ishiguro, and S. Nayar, "Programmable aperture camera using LCoS," in *European Conference on Computer Vision*. Springer, 2010, pp. 337–350.

[39] D. Dudley, W. Duncan, and J. Slaughter, "Emerging digital micromirror device (dmd) applications," in *Proc. SPIE*, vol. 4985, no. 14-25, 2003, p. 1.

[40] J. Castracane and M. Gutin, "Dmd-based bloom control for intensified imaging systems," in *Proc. SPIE*, vol. 3633, 1999, p. 234.

[41] S. Nayar, V. Branzoi, and T. Boult, "Programmable imaging: Towards a flexible camera," *International Journal of Computer Vision*, vol. 70, no. 1, pp. 7–22, 2006.

[42] C. Gao, N. Ahuja, and H. Hua, "Active aperture control and sensor modulation for flexible imaging," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2007, pp. 1–8.

[43] B. Bayer, "Color imaging array," Jul. 20 1976, uS Patent 3,971,065.

[44] R. Lukac and K. Plataniotis, "Color filter arrays: Design and performance analysis," *IEEE Transactions on Consumer Electronics*, vol. 51, no. 4, pp. 1260–1267, 2005.

[45] Y. Bando, B. Chen, and T. Nishita, "Extracting depth and matte using a color-filtered aperture," *ACM Transactions on Graphics (TOG)*, vol. 27, no. 5, pp. 1–9, 2008.

[46] D. Lee, I. Kweon, and R. Cipolla, "A biprism-stereo camera system," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, 2002.

[47] S. A. Nene and S. K. Nayar, "Stereo with mirrors," in *IEEE International Conference on Computer Vision*, 1998, pp. 1087–1094.

[48] E. Mouaddib, R. Sagawa, T. Echigo, and Y. Yagi, "Stereovision with a single camera and multiple mirrors," in *IEEE International Conference on Robotics and Automation*, 2005, pp. 800–805.

[49] A. Goshtasby and W. Gruver, "Design of a single-lens stereo camera system," *Pattern Recognition*, vol. 26, no. 6, pp. 923–937, 1993.

[50] C. Gao and N. Ahuja, "Single camera stereo using planar parallel plate," *Pattern Recognition*, vol. 4, pp. 108–111, 2004.

[51] A. Clark and S. Chan, "Single-camera computational stereo using a rotating mirror," in *Proc. British Machine Vision Conference*, 1994, pp. 13–16.

[52] T. Georgeiv, K. Zheng, B. Curless, D. Salesin, S. Nayar, and C. Intwala, "Spatio-Angular Resolution Tradeoff in Integral Photography," in *In Eurographics Symposium on Rendering*, 2006.

[53] S. Bogner, "Introduction to panoramic imaging," in *IEEE SMC Conference*, vol. 54, 1995, pp. 3100–3106.

[54] D. Buchele, "Unitary catadioptric objective," May 12 1953, uS Patent 2,638,033.

[55] J. Chahl and M. Srinivasan, "Reflective surfaces for panoramic imaging," *Applied Optics*, vol. 36, no. 31, pp. 8275–8285, 1997.

[56] J. Charles, R. Reeves, and C. Schur, "How to build and use an all-sky camera," *Astronomy Magazine*, 1987.

[57] J. Hong, X. Tan, B. Pinette, R. Weiss, and E. Riseman, "Image-based homing," *IEEE Control Systems Magazine*, vol. 12, no. 1, pp. 38–45, 1992.

[58] K. Yamazawa, Y. Yagi, and M. Yachida, "Omnidirectional imaging with hyperboloidal projection," in *International Conference on Intelligent Robots and Systems*, vol. 2, 1993, pp. 1029–1034.

[59] S. Kuthirummal and S. Nayar, "Flexible Mirror Imaging," in *IEEE International Conference on Computer Vision*, 2007, pp. 1–8.

[60] G. Krishnan and S. Nayar, "Cata-Fisheye Camera for Panoramic Imaging," in *IEEE Workshop on Applica-

tions of Computer Vision, 2008, pp. 1–8.

[61] T. Boult, X. Gao, R. Micheals, and M. Eckmann, "Omni-directional visual surveillance," *Image and Vision Computing*, vol. 22, no. 7, pp. 515–534, 2004.

[62] N. Winters, J. Gaspar, G. Lacey, and J. Santos-Victor, "Omni-directional vision for robot navigation," in *IEEE Workshop on Omnidirectional Vision*. IEEE, 2000, pp. 21–28.

[63] D. Chapman and A. Deacon, "Panoramic imaging and virtual reality–filling the gaps between the lines," *ISPRS journal of photogrammetry and remote sensing*, vol. 53, no. 6, pp. 311–319, 1998.

[64] T. Conroy and J. Moore, "Resolution invariant surfaces for panoramic vision systems," in *IEEE International Conference on Computer Vision*, 1999, p. 392.

[65] H. Nagahara, K. Yoshida, and M. Yachida, "An Omnidirectional Vision Sensor with Single View and Constant Resolution," in *IEEE International Conference on Computer Vision*, 1997, pp. 1–8.

[66] R. Hicks and R. Perline, "Equiresolution catadioptric sensors," *Applied Optics*, vol. 44, no. 29, pp. 6108–6114, 2005.

[67] R. Swaminathan, M. Grossberg, and S. Nayar, "Caustics of catadioptric cameras," in *IEEE International Conference on Computer Vision*, vol. 2, 2001, pp. 2–9.

[68] S. Baker and S. Nayar, "A theory of single-viewpoint catadioptric image formation," *International Journal of Computer Vision*, vol. 35, no. 2, pp. 175–196, 1999.

[69] R. Hicks, "Designing a mirror to realize a given projection," *Journal of the Optical Society of America A*, vol. 22, no. 2, pp. 323–330, 2005.

[70] E. Talvala, A. Adams, M. Horowitz, and M. Levoy, "Veiling glare in high dynamic range imaging," in *SIGGRAPH*. ACM, 2007, pp. 37–es.

[71] S. Kuthirummal and S. Nayar, "Multiview radial catadioptric imaging for scene capture," *ACM Transactions on Graphics (TOG)*, vol. 25, no. 3, pp. 916–923, 2006.

[72] H. Du, X. Tong, X. Cao, and S. Lin, "A prism-based system for multispectral video acquisition," in *IEEE International Conference on Computer Vision*, 2009, pp. 175–182.

[73] S. Umeyama and G. Godin, "Separation of diffuse and specular components of surface reflection by use of polarization and statistical analysis of images," *IEEE Pattern Analysis and Machine Intelligence*, vol. 26, no. 5, pp. 639–647, 2004.

[74] S. K. Nayar, X. Fang, and T. Boult, "Separation of reflection components using color and polarization," *International Journal of Computer Vision*, vol. 21, no. 3, pp. 163–186, 1997.

[75] S. Lin and S. Lee, "Detection of specularity using stereo in color and polarization space," *Computer Vision and Image Understanding*, vol. 65, no. 2, pp. 336–346, 1997.

[76] R. Raskar, A. Agrawal, and J. Tumblin, "Coded exposure photography: motion deblurring using fluttered shutter," in *SIGGRAPH*. ACM, 2006, pp. 795–804.

[77] M. Rouf, R. Mantiuk, W. Heidrich, M. Trentacoste,

and C. Lau, "Glare encoding of high dynamic range images," in *IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2011, pp. 289–296.

[78] R. Gonzalez and R. Woods, "Digital image processing," *Prentice Hall*, 2002.

[79] A. Levin, Y. Weiss, F. Durand, and W. Freeman, "Understanding and evaluating blind deconvolution algorithms," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 1964–1971.

[80] A. Pentland, "A new sense for depth of field," *IEEE Pattern Analysis and Machine Intelligence*, no. 4, pp. 523–531, 1987.

[81] M. Subbarao, "Parallel depth recovery by changing camera parameters," in *IEEE International Conference on Computer Vision*, vol. 1, 1988.

[82] S. Chaudhuri and A. Rajagopalan, *Depth from defocus: a real aperture imaging approach*. Springer Verlag, 1999.

[83] M. Watanabe and S. Nayar, "Rational filters for passive depth from defocus," *International Journal of Computer Vision*, vol. 27, no. 3, pp. 203–225, 1998.

[84] Y. Schechner and N. Kiryati, "Depth from defocus vs. stereo: How different really are they?" *International Journal of Computer Vision*, vol. 39, no. 2, pp. 141–162, 2000.

[85] W. Welford, "Use of annular apertures to increase focal depth," *Journal of the Optical Society of America A*, vol. 50, no. 8, pp. 749–752, 1960.

[86] J. Ojeda-Castaneda, P. Andres, and A. Diaz, "Annular apodizers for low sensitivity to defocus and to spherical aberration," *Optics Letters*, vol. 11, no. 8, pp. 487–489, 1986.

[87] E. Fenimore and T. Cannon, "Coded aperture imaging with uniformly redundant arrays," *Applied Optics*, vol. 17, no. 3, pp. 337–347, 1978.

[88] S. Gottesman and E. Fenimore, "New family of binary arrays for coded aperture imaging," *Applied Optics*, vol. 28, no. 20, pp. 4344–4352, 1989.

[89] C. Zhou, S. Lin, and S. Nayar, "Coded aperture pairs for depth from defocus," in *IEEE International Conference on Computer Vision*, 2009.

[90] A. Levin, "Analyzing Depth from Coded Aperture Sets," in *European Conference on Computer Vision*, 2010.

[91] M. Brown, D. Burschka, and G. Hager, "Advances in computational stereo," *IEEE Pattern Analysis and Machine Intelligence*, pp. 993–1008, 2003.

[92] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *International Journal of Computer Vision*, vol. 47, no. 1, pp. 7–42, 2002.

[93] S. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski, "A comparison and evaluation of multi-view stereo reconstruction algorithms," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, 2006, pp. 519–528.

[94] G. Indebetouw and H. Bai, "Imaging with Fresnel zone pupil masks: extended depth of field," *Applied Optics*, vol. 23, no. 23, pp. 4299–4302, 1984.

[95] J. Ojeda-Castaneda and L. Berriel-Valdos, "Zone plate for arbitrarily high focal depth," *Applied Optics*, vol. 29, no. 7, pp. 994–997, 1990.

[96] P. Green, W. Sun, W. Matusik, and F. Durand, "Multi-aperture photography," in *SIGGRAPH*. ACM, 2007, pp. 68–es.

[97] M. Aggarwal and N. Ahuja, "Split aperture imaging for high dynamic range," *International Journal of Computer Vision*, vol. 58, no. 1, pp. 7–17, 2004.

[98] E. Dowski, "Passive ranging with an incoherent optical system," 1993.

[99] A. Greengard, Y. Schechner, and R. Piestun, "Depth from diffracted rotation," *Optics Letters*, vol. 31, no. 2, pp. 181–183, 2006.

[100] W. Cathey and E. Dowski, "New paradigm for imaging systems," *Applied Optics*, vol. 41, no. 29, pp. 6080–6092, 2002.

[101] E. Dowski and W. Cathey, "Extended depth of field through wave-front coding," *Applied Optics*, vol. 34, no. 11, pp. 1859–1866, 1995.

[102] Q. Yang, L. Liu, J. Sun, Y. Zhu, and W. Lu, "Analysis of optical systems with extended depth of field using the Wigner distribution function," *Applied Optics*, vol. 45, no. 34, pp. 8586–8595, 2006.

[103] M. Somayaji and M. Christensen, "Frequency analysis of the wavefront-coding odd-symmetric quadratic phase mask," *Applied Optics*, vol. 46, no. 2, pp. 216–226, 2007.

[104] A. Levin, S. Hasinoff, P. Green, F. Durand, and W. Freeman, "4D frequency analysis of computational cameras for depth of field extension," in *SIGGRAPH*. ACM, 2009, pp. 1–14.

[105] A. Samokhin, A. Simonov, and M. Rombach, "Optical system invariant to second-order aberrations," *Journal of the Optical Society of America A*, vol. 26, no. 4, pp. 977–984, 2009.

[106] N. George and W. Chi, "Extended depth of field using a logarithmic asphere," *Journal of Optics A: Pure and Applied Optics*, vol. 5, p. S157, 2003.

[107] A. Castro and J. Ojeda-Castañeda, "Asymmetric phase masks for extended depth of field," *Applied Optics*, vol. 43, no. 17, pp. 3474–3479, 2004.

[108] E. Garcia-Guerrero, E. Mendez, H. Escamilla, T. Leskova, and A. Maradudin, "Design and fabrication of random phase diffusers for extending the depth of focus," *Optics Express*, vol. 15, no. 3, pp. 910–923, 2007.

[109] R. Ng, "Fourier slice photography," in *SIGGRAPH*. ACM, 2005, pp. 735–744.

[110] R. Raskar, A. Agrawal, C. Wilson, and A. Veeraraghavan, "Glare aware photography: 4D ray sampling for reducing glare effects of camera lenses," *ACM Transactions on Graphics (TOG)*, vol. 27, no. 3, pp. 1–10, 2008.

[111] R. Ng, M. Levoy, M. Brédif, G. Duval, M. Horowitz, and P. Hanrahan, "Light field photography with a hand-held plenoptic camera," *Stanford Computer Science Technical Report*, vol. 2, 2005.

[112] A. Levin, P. Sand, T. Cho, F. Durand, and W. Freeman, "Motion-invariant photography," in *SIGGRAPH*. ACM, 2008, pp. 1–9.

[113] G. Lippmann, "La photographie integrale," *Comptes-Rendus, Academie des Sciences*, no. 146, pp. 446–551, 1908.

[114] H. Ives, "Parallax panoramagrams made with a large diameter lens," *Journal of the Optical Society of America A*, vol. 20, no. 6, pp. 332–340, 1930.

[115] A. Lumsdaine and T. Georgiev, "The focused plenoptic camera," in *International Conference on Computational Photography*, vol. 5, 2009, p. 6.

[116] T. Bishop, S. Zanetti, and P. Favaro, "Light field superresolution," in *International Conference on Computational Photography*, 2009.

[117] P. Dillon, A. Brault, J. Horak, E. Garcia, T. Martin, and W. Light, "Fabrication and performance of color filter arrays for solid-state imagers," *IEEE Journal of Solid-State Circuits*, vol. 13, no. 1, pp. 23–27, 1978.

[118] J. Adams and J. Hamilton, "Design of practical color filter array interpolation algorithms for digital cameras," in *Proc. SPIE*, vol. 3028, 1997, pp. 117–125.

[119] B. Gunturk, J. Glotzbach, Y. Altunbasak, R. Schafer, and R. Mersereau, "Demosaicking: color filter array interpolation," *IEEE Signal Processing Magazine*, vol. 22, no. 1, pp. 44–54, 2005.

[120] J. Hamilton Jr and J. Adams Jr, "Adaptive color plan interpolation in single sensor color electronic camera," May 13 1997, uS Patent 5,629,734.

[121] S. K. Nayar and S. Narasimhan, "Assorted pixels: Multi-sampled imaging with structural models," in *European Conference on Computer Vision*. Springer, 2006, pp. 135–315.

[122] M. Ben-Ezra, Z. Lin, and B. Wilburn, "Penrose Pixels Super-Resolution in the Detector Layout Domain," in *IEEE International Conference on Computer Vision*, 2007, pp. 1–8.

[123] R. Baker, *CMOS: Circuit design, layout, and simulation*. Wiley- Press, 2010.

[124] T. Chen, P. Catrysse, A. El Gamal, B. Wandell *et al.*, "How small should pixel size be?" in *Proc. SPIE*, vol. 3965, 2000, pp. 451–459.

[125] P. Lee, R. Guidash, T. Lee, and E. Stevens, "Active pixel sensor integrated with a pinned photodiode," Apr. 29 1997, uS Patent 5,625,210.

[126] M. Schanz, C. Nitta, A. Bußmann, B. Hosticka, and R. Wertheimer, "A high-dynamic-range CMOS image sensor for automotive applications," *Solid-State Circuits, IEEE Journal of*, vol. 35, no. 7, pp. 932–938, 2000.

[127] J. Bechtel, J. Andrus, and T. Sherman, "Digital Image Processing and Systems Incorporating the Same," Aug. 5 2010, uS Patent App. 20,100/195,908.

[128] C. Mead, *Analog VLSI and neural systems*. Addison-Wesley, 1989.

[129] J. Wyatt, C. Keast, M. Seidel, D. Standley, B. Horn, T. Knight, C. Sodini, H. Lee, and T. Poggio, "Analog VLSI systems for image acquisition and fast early

vision processing," *International Journal of Computer Vision*, vol. 8, no. 3, pp. 217–230, 1992.

[130] T. Kanade and R. Bajcsy, "Computational sensors: A report from DARPA workshop," *IUS Proceedings*, 1993.

[131] H. Nagahara, S. Kuthirummal, C. Zhou, and S. Nayar, "Flexible depth of field photography," in *European Conference on Computer Vision*, 2008.

[132] G. Hausler, "A method to increase the depth of focus by two step image processing," *Optics Communications*, vol. 6, no. 1, pp. 38–42, 1972.

[133] R. Lenz and U. Lenz, "Progres 3000: a digital color camera with a 2-d array ccd sensor and programmable resolution up to 2994 x 2320 picture elements," in *Proc. SPIE*, vol. 1357, 1990, p. 204.

[134] M. Ben-Ezra, A. Zomet, and S. Nayar, "Jitter camera: high resolution video from a low resolution detector," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, 2004, pp. II–135.

[135] ——, "Video super-resolution using controlled subpixel detector shifts," *IEEE Pattern Analysis and Machine Intelligence*, pp. 977–987, 2005.

[136] T. Cho, A. Levin, F. Durand, and W. Freeman, "Motion blur removal with orthogonal parabolic exposures," in *International Conference on Computational Photography*, 2010, pp. 1–8.

[137] A. Rav-Acha and S. Peleg, "Restoration of multiple images with motion blur in different directions," in *IEEE Workshop on Applications of Computer Vision*, 2000, pp. 22–28.

[138] Microsoft, "Microsoft kinect depth sensor for xbox." [Online]. Available: http://www.xbox.com/en-US/kinect

[139] L. Zhang and S. Nayar, "Projection defocus analysis for scene capture and image display," *ACM Transactions on Graphics (TOG)*, vol. 25, no. 3, pp. 907–915, 2006.

[140] M. Gupta, Y. Tian, S. Narasimhan, and L. Zhang, "(de) focusing on global light transport for active scene recovery," 2009.

[141] A. Kirmani, T. Hutchison, J. Davis, and R. Raskar, "Looking around the corner using transient imaging," in *IEEE International Conference on Computer Vision*, 2009, pp. 159–166.

[142] R. Raskar, K. Tan, R. Feris, J. Yu, and M. Turk, "Non-photorealistic camera: depth edge detection and stylized rendering using multi-flash imaging," *ACM Transactions on Graphics (TOG)*, vol. 23, no. 3, pp. 679–688, 2004.

[143] W. Ma, T. Hawkins, P. Peers, C. Chabert, M. Weiss, and P. Debevec, "Rapid acquisition of specular and diffuse normal maps from polarized spherical gradient illumination," *Rendering Techniques*, vol. 2007, no. 9, p. 10, 2007.

[144] L. Zhang, B. Curless, and S. Seitz, "Spacetime stereo: shape recovery for dynamic scenes," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, 2003, pp. II–367.

[145] S. K. Nayar, G. Krishnan, M. Grossberg, and R. Raskar, "Fast separation of direct and global components of a scene using high frequency illumination," in *SIGGRAPH*. ACM, 2006, pp. 935–944.

[146] G. Petschnigg, R. Szeliski, M. Agrawala, M. Cohen, H. Hoppe, and K. Toyama, "Digital photography with flash and no-flash image pairs," *ACM Transactions on Graphics (TOG)*, vol. 23, no. 3, pp. 664–672, 2004.

[147] E. Eisemann and F. Durand, "Flash photography enhancement via intrinsic relighting," *ACM Transactions on Graphics (TOG)*, vol. 23, no. 3, pp. 673–678, 2004.

[148] A. Agrawal, R. Raskar, S. Nayar, and Y. Li, "Removing photography artifacts using gradient projection and flash-exposure sampling," in *SIGGRAPH*. ACM, 2005, pp. 828–835.

[149] T. Zickler, P. Belhumeur, and D. Kriegman, "Helmholtz stereopsis: Exploiting reciprocity for surface reconstruction," *International Journal of Computer Vision*, vol. 49, no. 2, pp. 215–227, 2002.

[150] S. Seitz, Y. Matsushita, and K. Kutulakos, "A theory of inverse light transport," in *IEEE International Conference on Computer Vision*, vol. 2, 2005, pp. 1440–1447.

[151] P. Sen, B. Chen, G. Garg, S. Marschner, M. Horowitz, M. Levoy, and H. Lensch, "Dual photography," *ACM Transactions on Graphics (TOG)*, vol. 24, no. 3, pp. 745–755, 2005.

[152] G. Garg, E. Talvala, M. Levoy, and H. Lensch, "Symmetric photography: Exploiting data-sparseness in reflectance fields," *Rendering Techniques*, pp. 251–262, 2006.

[153] M. O'Toole and K. Kutulakos, "Optical computing for fast light transport analysis," *ACM Transactions on Graphics (TOG)*, vol. 29, no. 6, p. 164, 2010.

[154] Y. Schechner, S. Nayar, and P. Belhumeur, "A theory of multiplexed illumination," in *IEEE International Conference on Computer Vision*, 2003.

[155] A. Wenger, A. Gardner, C. Tchou, J. Unger, T. Hawkins, and P. Debevec, "Performance relighting and reflectance transformation with time-multiplexed illumination," *ACM Transactions on Graphics (TOG)*, vol. 24, no. 3, pp. 756–764, 2005.

[156] J. Park, M. Lee, M. Grossberg, and S. Nayar, "Multispectral imaging using multiplexed illumination," in *IEEE International Conference on Computer Vision*, 2007, pp. 1–8.

[157] M. Gustafsson, "Surpassing the lateral resolution limit by a factor of two using structured illumination microscopy," *Journal of Microscopy*, vol. 198, no. 2, pp. 82–87, 2000.

[158] ——, "Nonlinear structured-illumination microscopy: wide-field fluorescence imaging with theoretically unlimited resolution," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 102, no. 37, p. 13081, 2005.

[159] J. Burch and C. Forno, "A high sensitivity moire grid technique for studying deformation in large objects," *Optical Engineering*, vol. 14, pp. 178–185, 1975.

[160] C. Forno, "Deformation measurement using high resolution moiré photography," *Optics and lasers in engineering*, vol. 8, no. 3-4, pp. 189–212, 1988.

[161] C. Sciammarella, "The moire methoda review," *Experimental Mechanics*, vol. 22, no. 11, pp. 418–433, 1982.

[162] B. Horn, "Density reconstruction using arbitrary ray-sampling schemes," *Proceedings of the IEEE*, vol. 66, no. 5, pp. 551–562, 1978.

[163] M. Lee, F. Bowman, A. Brill, B. Horn, R. Lanza, J. Park, and C. Sodini, "Optical diffusion data acquisition system," *Future Direction of Lasers in Surgery and Medicine, Salt Lake City, Utah*, 1995.

[164] B. Wilburn, N. Joshi, V. Vaish, E. Talvala, E. Antunez, A. Barth, A. Adams, M. Horowitz, and M. Levoy, "High performance imaging using large camera arrays," *ACM Transactions on Graphics (TOG)*, vol. 24, no. 3, pp. 765–776, 2005.

[165] B. Smith, L. Zhang, H. Jin, and A. Agarwala, "Light field video stabilization," in *IEEE International Conference on Computer Vision*, 2009, pp. 341–348.

[166] R. Hartley, *Multiple view geometry in computer vision*. Cambridge university press, 2008.

[167] M. Okutomi and T. Kanade, "A multiple-baseline stereo," *IEEE Pattern Analysis and Machine Intelligence*, vol. 15, no. 4, pp. 353–363, 2002.

[168] S. Baker, T. Sim, and T. Kanade, "A characterization of inherent stereo ambiguities," in *IEEE International Conference on Computer Vision*, vol. 1, 2001, pp. 428–435.

[169] M. Agrawal and L. Davis, "Trinocular stereo using shortest paths and the ordering constraint," *International Journal of Computer Vision*, vol. 47, no. 1, pp. 43–50, 2002.

[170] M. Goesele, B. Curless, and S. Seitz, "Multi-view stereo revisited," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, 2006, pp. 2402–2409.

[171] Y. Ding, F. Li, Y. Ji, and J. Yu, "Dynamic 3d fluid surface acquisition using a camera array," in *IEEE International Conference on Computer Vision*, 2011.

[172] J. Yang, M. Everett, C. Buehler, and L. McMillan, "A real-time distributed light field camera," in *Proceedings of the 13th Eurographics workshop on Rendering*. Eurographics Association, 2002, pp. 77–86.

[173] Y. Nomura, L. Zhang, and S. Nayar, "Scene collages and flexible camera arrays," in *Proceedings of Eurographics Symposium on Rendering*, 2007.

[174] H. Shum and S. Kang, "A review of image-based rendering techniques," *IEEE/SPIE Visual Communications and Image Processing (VCIP)*, vol. 213, 2000.

[175] D. Gledhill, G. Tian, D. Taylor, and D. Clarke, "Panoramic imaging–a review," *Computers & Graphics*, vol. 27, no. 3, pp. 435–445, 2003.

[176] C. Zhang and T. Chen, "Non-uniform sampling of image-based rendering data with the position-interval error (pie) function," *Visual Communication and Image Processing (VCIP)*, vol. 3, 2003.

[177] S. Fleishman, D. Cohen-Or, and D. Lischinski, "Automatic Camera Placement for Image-Based Modeling," in *Computer Graphics Forum*, vol. 19, no. 2. Wiley Online Library, 2000, pp. 101–110.

[178] H. Shum and L. He, "Rendering with concentric mosaics," in *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*. ACM, 1999, pp. 299–306.

[179] J. Chai, X. Tong, S. Chan, and H. Shum, "Plenoptic sampling," in *SIGGRAPH*. ACM, 2000, pp. 307–318.

[180] C. Zhang and T. Chen, "Spectral analysis for sampling image-based rendering data," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 11, pp. 1038–1050, 2003.

[181] Z. Lin and H. Shum, "On the number of samples needed in light field rendering with constant-depth assumption," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, 2000, pp. 588–595.

[182] T. Kanade, P. Rander, and P. Narayanan, "Virtualized reality: Constructing virtual worlds from real scenes," *IEEE Multimedia*, vol. 4, no. 1, pp. 34–47, 1997.

[183] O. Cossairt, D. Miau, and S. Nayar, "Gigapixel computational imaging," in *International Conference on Computational Photography*, 2010, pp. 1–8.

[184] A. Adams, D. Jacobs, J. Dolson, M. Tico, K. Pulli, E. Talvala, B. Ajdin, D. Vaquero, H. Lensch, M. Horowitz *et al.*, "The Frankencamera: an experimental platform for computational photography," in *SIGGRAPH*. ACM, 2010, pp. 1–12.

[185] D. Stork and M. Robinson, "Theoretical foundations for joint digital-optical analysis of electro-optical imaging systems," *Applied Optics*, vol. 47, no. 10, p. 64, 2008.

[186] D. Robinson and D. Stork, "Joint design of lens systems and digital image processing," in *International Optical Design Conference*. Optical Society of America, 2006.

[187] M. Robinson and D. Stork, "Joint digital-optical design of superresolution multiframe imaging systems," *Applied Optics*, vol. 47, no. 10, pp. B11–B20, 2008.

[188] M. Robinson, D. Stork, and R. Innovations, "Joint digital-optical design of imaging systems for grayscale objects," in *Proceedings of SPIE European Optical Design Conference*, 2008, pp. 1–9.

[189] M. Robinson, G. Feng, D. Stork, and R. Innovations, "Spherical coded imagers: Improving lens speed, depth-of-field, and manufacturing yield through enhanced spherical aberration and compensating image processing," in *Proc. SPIE*, vol. 7429, pp. 74 290M–1.

[190] P. Mouroulis, "Depth of field extension with spherical optics," *Optics Express*, vol. 16, no. 17, pp. 12 995–13 004, 2008.

[191] O. Cossairt and S. Nayar, "Spectral Focal Sweep: Extended depth of field from chromatic aberrations," in *International Conference on Computational Photography*, 2010, pp. 1–8.

[192] F. Guichard, H. Nguyen, R. Tessières, M. Pyanet, I. Tarchouna, and F. Cao, "Extended depth-of-field using sharpness transport across color channels," *Technical Paper, DXO Labs*, 2009.

[193] A. Zomet and S. Nayar, "Lensless Imaging with a Controllable Aperture," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2006, pp. 339–346.

[194] J. Tumblin, A. Agrawal, and R. Raskar, "Why i want a gradient camera," in *IEEE Conference on Computer*

Vision and Pattern Recognition, vol. 1, 2005, pp. 103–110.

[195] R. Gupta and R. Hartley, "Linear pushbroom cameras," *IEEE Pattern Analysis and Machine Intelligence*, vol. 19, no. 9, pp. 963–975, 1997.

[196] J. Semple and G. Kneebone, "Algebraic projective geometry," *Bull. Amer. Math. Soc. 59 (1953), 571-572. DOI: 10.1090/S0002-9904-1953-09763-4 PII: S*, vol. 2, no. 9904, pp. 09 763–4, 1953.

[197] A. Zomet, D. Feldman, S. Peleg, and D. Weinshall, "Mosaicing new views: The crossed-slits projection," *IEEE Pattern Analysis and Machine Intelligence*, pp. 741–754, 2003.

[198] J. Yu and L. McMillan, "General linear cameras," in *European Conference on Computer Vision*. Springer, 2004, pp. 14–27.

[199] J. Yu, L. McMillan, and P. Sturm, "State of the art report: Multiperspective rendering, modeling, and imaging," in *Eurographics State of the Art Report*, 2008, pp. 1–24.

[200] M. Ben-Ezra, "High resolution large format tile-scan camera: Design, calibration, and extended depth of field," in *International Conference on Computational Photography*, 2010, pp. 1–8.

[201] Gigapixel Project. (2007) The gigapixl project. [Online]. Available: "http://www.gigapixl.org/"

[202] D. Marks and D. Brady, "Gigagon: A monocentric lens design imaging 40 gigapixels," in *Imaging Systems*. Optical Society of America, 2010.

[203] D. Brady and N. Hagen, "Multiscale lens design," *Optics Express*, vol. 17, no. 13, pp. 10 659–10 674, 2009.

[204] S. Park, E. Park, and H. Kim, "Image deblurring using vibration information from 3-axis accelerometer," *Journal of the Institute of Electronics Engineers of Korea*, vol. 3, p. 111., 2008.

[205] N. Joshi, S. Kang, C. Zitnick, and R. Szeliski, "Image deblurring using inertial measurement sensors," *ACM Transactions on Graphics (TOG)*, vol. 29, no. 4, pp. 1–9, 2010.

[206] E. Adelson and J. Wang, "Single lens stereo with a plenoptic camera," *IEEE Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 99–106, 2002.

[207] K. Zheng, B. Curless, D. Salesin, S. Nayar, and C. Intwala, "Spatio-angular resolution tradeoff in integral photography," *Rendering Techniques*, pp. 263–272, 2006.

[208] C. Gao and N. Ahuja, "A refractive camera for acquiring stereo and super-resolution images," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, 2006, pp. 2316–2323.

[209] Y. Xiao and K. Lim, "A prism-based single-lens stereo-vision system: From trinocular to multi-ocular," *Image and Vision Computing*, vol. 25, no. 11, pp. 1725–1736, 2007.

[210] S. Prasad, T. Torgersen, V. Pauca, R. Plemmons, and J. van der Gracht, "Engineering the pupil phase to improve image quality," in *Proc. SPIE*, vol. 5108, 2003, pp. 1–12.

[211] A. Busboom, H. Schotten, and H. Elders-Boll, "Coded aperture imaging with multiple measurements," *Journal of the Optical Society of America A*, vol. 14, no. 5, pp. 1058–1065, 1997.

[212] E. Caroli, J. Stephen, G. Cocco, L. Natalucci, and A. Spizzichino, "Coded aperture imaging in X-and gamma-ray astronomy," *Space Science Reviews*, vol. 45, no. 3, pp. 349–403, 1987.

[213] T. Poon and M. Motamedi, "Optical/digital incoherent image processing for extended depth of field," *Applied Optics*, vol. 26, no. 21, pp. 4612–4615, 1987.

[214] M. Aggarwal and N. Ahuja, "High dynamic range panoramic imaging," in *IEEE International Conference on Computer Vision*, vol. 1, 2001, pp. 2–9.

[215] S. K. Nayar and V. Branzoi, "Adaptive Dynamic Range Imaging: Optical Control of Pixel Exposures Over Space and Time," in *IEEE International Conference on Computer Vision*, 2003, p. 1168.

[216] S. K. Nayar, V. Branzoi, and T. Boult, "Programmable imaging using a digital micromirror array," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2004.

[217] S. Hiura and T. Matsuyama, "Depth measurement by the multi-focus camera," in *IEEE Conference on Computer Vision and Pattern Recognition*, 1998, pp. 953–959.

[218] M. McGuire, W. Matusik, H. Pfister, J. Hughes, and F. Durand, "Defocus video matting," *ACM Transactions on Graphics (TOG)*, vol. 24, no. 3, pp. 567–576, 2005.

[219] Y. Schechner, S. Narasimhan, and S. Nayar, "Instant Dehazing of Images Using Polarization," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, 2001, pp. 325–332.

[220] L. Wolff, "Using polarization to separate reflection components," in *IEEE Conference on Computer Vision and Pattern Recognition*, 1989, pp. 363–369.

[221] A. Subramanian, L. Iyer, A. Abbott, and A. Bell, "Image Segmentation and Range Sensing Using a Moving Aperture Lens," *IEEE Transactions on Multimedia*, 2001.

[222] S. K. Nayar, "Catadioptric omnidirectional camera," in *IEEE Conference on Computer Vision and Pattern Recognition*, 1997, p. 482.

[223] J. Gluckman and S. Nayar, "Planar catadioptric stereo: Geometry and calibration," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, 1999.

[224] S. Lin and R. Bajcsy, "True single view point cone mirror omni-directional catadioptric system," in *IEEE International Conference on Computer Vision*, vol. 2, 2001, pp. 102–107.

[225] S. K. Nayar and V. Peri, "Folded catadioptric cameras," in *IEEE Conference on Computer Vision and Pattern Recognition*, 1999, p. 2217.

[226] K. Tan, H. Hua, and N. Ahuja, "Multiview panoramic cameras using mirror pyramids," *IEEE Pattern Analysis and Machine Intelligence*, vol. 26, no. 7, pp. 941–946, 2004.