Developing A Simulation Toolbox For Biomedical Plenoptic Imaging

C Meah[1]
c.j.meah@pgr.bham.ac.uk
R Marshall[1]
r.j.marshall@pgr.bham.ac.uk
E Claridge[2]
e.claridge@cs.bham.ac.uk
K Bongs[3]
k.bongs@bham.ac.uk
I Styles[2]
i.b.styles@cs.bham.ac.uk

[1] PSIBS DTC
School of Chemistry
[2] School of Computer Science
[3] School of Physics and Astronomy
University of Birmingham
Edgbaston
UK, B15 2TT

Abstract

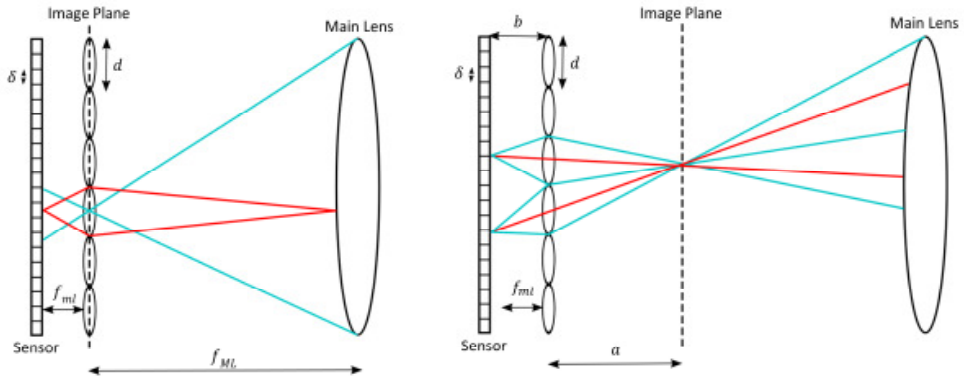
Plenoptic imaging allows a user to gain 3-D information from a single acquisition. A toolbox is presented with unified plenoptic simulation, rendering, and reconstruction, which allows the possibility for complete *in silico* exploration of potential designs and applications. The example of 3-D fluorescence microscopy is explored with plenoptic tomographic reconstruction.

1 Introduction

Plenoptic, or light field, imaging is an exciting new technology which provides a user with 3-D information about a scene from a single camera in a single acquisition. Through post-processing of plenoptic data we can create a synthetic aperture, refocus digitally, generate novel viewpoints, and achieve 3-D reconstruction. These abilities have potential benefits in a wide variety of applications. In particular, medical imaging applications can benefit from this technology since it can dramatically reduce acquisition time. In translucent scenes, such as in microscopy, retinal imaging, or particle image velocimetry (PIV), this allows snapshot volumetric imaging which provides improved insight into dynamic micro-biology [6].

The main limitation of plenoptic imaging is the loss of spatial resolution, which is due to the spatio-angular trade-off [4]. Particular applications will require different resolutions, and so optimisation of plenoptic design is required for optimal imaging. This optimisation, however, is non-trivial and practical exploration can be expensive since many possible components must be available. Furthermore, placement of optical components must be accurate on the order of microns for optimal imaging to take place.

Simulation allows this exploration to occur without the need for a physical plenoptic system. This is not only beneficial for system design, but also for algorithm development



(a) (b)

Figure 1: A plenoptic camera utilises a microlens array to preserve the directional information of incoming rays. (a) shows the plenoptic 1.0 set-up, where the microlens array is placed on the main lens image plane. (b) shows the plenoptic 2.0 set-up, where the microlens array and sensor act as a relay system.

since validation and testing can be performed on well-characterised simulated data. This is important in biomedical imaging applications utilising 3-D segmentation and tracking.

This paper presents a unified, open source toolbox being developed to allow complete *in silico* plenoptic exploration. A camera can be designed, image acquisition simulated, and performance evaluated with in-built reconstruction algorithms. Virtual exploration is cheaper, easier, quicker, and more readily lends itself to quantitative evaluation. As an example of the tool's effectiveness, the software is utilised to investigate a plenoptic camera reconstruction of 3-D fluorescence microscopy.

2 Methods

A software tool has been written, using MATLAB (Release 2013a, The MathWorks Inc., Natick, MA, USA), which provides simulation and reconstruction methods for plenoptic data. This reconstruction can render images or 3-D visualisations from real or simulated plenoptic data. The simulation tool allows a user to design a bespoke virtual plenoptic camera, input a custom scene, and simulate the image acquisition to generate a raw image. The software is beneficial for optimisation of design, algorithm validation, and exploring applications.

2.1 Plenoptic Imaging

The light field is a term used to describe the distribution of light rays in space. A ray can be described by its position (x,y,z) and direction (θ,ϕ), giving a 5-D function often called the plenoptic function, $L_F(x,y,z,\theta,\phi)$. In free space, where there are no occluders, the radiance of a ray does not change along its length. This means we can lose one dimension, although this makes the parameterisation less obvious. We can characterise a ray by its intersection with two parallel planes (the uv- and st-planes), giving us a 4-D representation: $L(u,v,s,t)$.

A conventional camera only captures the positional information of light, but by placing a microlens array between the main lens and image sensor we create a plenoptic camera

and can capture the full 4-D function. The placement of the microlens array dictates the type of data captured, and therefore has bearing on the resolution achieved by the camera. Ng *et al*. developed a camera design with the microlens array on the main lens imaging plane, and the sensor one focal length behind the microlenses [9]. In this set-up, known as the *plenoptic 1.0* architecture, spatial resolution is determined by the number of microlenses, with the angular resolution dictated by the number of pixels under a microlens. This sacrifice in spatial resolution is a major drawback of plenoptic imaging. Lumsdaine and Georgiev traded back some of this angular resolution gain by adapting the camera design and creating a relay system with the microlens array and sensor [7]. In this set-up, known as the *plenoptic 2.0* architecture, the microlenses are a distance a from the main lens image plane, and the sensor is a distance b behind, where the distances obey the thin lens equation

$$1/f = 1/a + 1/b\,. \tag{1}$$

Now resolution is dictated by the a/b relationship and is therefore more flexible. These two different cameras capture different data and therefore require different rendering algorithms in order to achieve optimal reconstruction.

In plenoptic 1.0 rendering, since each microlens captures information for one spatial point, we can simply sum the pixels underneath each microlens and tile them together. In terms of light field irradiance, the value at point s,t can be calculated as

$$I(s,t) = \iint L(u,v,s,t)\mathrm{d}u\mathrm{d}v\,. \tag{2}$$

By applying u and v limits to the integrals, we can dictate the viewpoint and aperture size of the image produced. Refocusing, detailed in Ng *et al*. [9], can be achieved since rays can be traced from a virtual plane to the sensor, and we can interpolate the values that would have been captured at the refocusing plane.

In plenoptic 2.0 rendering, we can take multiple pixels from each microlens to contribute to a final rendered image rather than summing them together, meaning a dramatic increase in spatial resolution. The size of the window of pixels taken from each microlens dictates the focal depth at which objects are shown "in focus". Note "in-focus" in this case means an image void of plenoptic artifacts [3]. The effect of aperture can be incorporated into rendering by summing together information from neighbouring microlenses. Viewpoint is dependent on the difference between microlens and pixel window centres.

By utilising Equation 2 we can achieve these rendering effects from a single plenoptic data set. These are included functions in the toolbox presented, although a full description of the methods are beyond the scope of this paper (see [3][8] for details).

2.1.1 3-D Reconstruction

There are a number of methods available in order to gain a 3-D reconstruction from a plenoptic data set. A depth map representing the surface topography of a scene can be obtained by calculating the disparity between microimages [3].

Another method of 3-D reconstruction can be taken from the field of microscopy, where images focused on different image planes are acquired in order to produce a z-stack. We can attempt to eliminate contributions from out of focus planes in each image by performing deconvolution on this stack, yielding a 3-D model of the data [6]. This pipeline can be translated into plenoptic imaging since we can digitally refocus meaning a z-stack can be created post-acquisition.

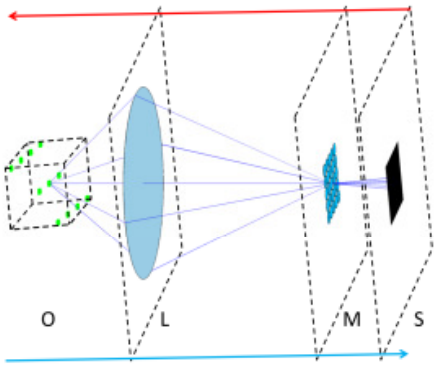


Figure 2: A 3-D schematic of the simulation. The input in this case is a volume representing fluorescent point sources in a transparent medium. Sample rays are shown traced through the system. The dashed lines represent (from left to right) the bounding box of the input volume (O), the main lens plane (L), the microlens array plane (M), and the sensor plane (S). The *red* arrow shows the direction of propagation for backwards simulations, and the *blue* for forwards simulation.

An alternative approach, which does not require the generation of a focal stack, is to pose the problem in terms of limited-angle tomography. Using techniques borrowed from computed tomography, we can iteratively solve the ill-posed and under-determined problem [1]. By treating measurements as projections through a volume, we can reconstruct voxel densities. This can be done separately for each "slice" of the volume by using the light field digital refocusing algorithm to project rays through slices [9]. The method outlined in [1] has been implemented in the reconstruction software so that a user can specify a volume of interest in object space. This is transformed into an image-side volume using the thin lens equation, seen in Equation 1, to calculate image distance (d_i) from object distance (d_o). The volume is also scaled with respect to magnification using $M = -d_i/d_o$. The light field can be projected through this image-side volume, and algebraic reconstruction techniques can be used to iteratively calculate the voxel densities.

2.2 Simulation

The simulation package allows a user the freedom to create a custom virtual camera. Ray transfer matrices (RTMs) [5] are used to propagate rays through an optical system. Ray transfer matrices are easily extensible, with the ability to account for intensity, and also incorporate more complex ray models [2]. Matrix notation allows the code to be highly vectorised which leads to improved software efficiency. Further speed improvements have been achieved through parallel computation using the MATLAB Parallel Toolbox™.

An optical component, such as a lens or microlens array, can be created by specifying parameters such as focal length, diameter, and thickness. For most simulations, the thin lens equation is sufficient for calculations, however a thick lens approximation is available if the radii of curvature are known for the lens. A component can be custom created so long as an operator matrix can be defined for it. A user can create an arbitrary system by placing optical components at specified positions.

We can combine optical component matrices to form a system matrix, meaning a single calculation can propagate a ray through a system. However, when considering a microlens

array we must first investigate which lens a ray hits to calculate how it is affected by that microlens.

The simulation can either propagate rays forwards or backwards through a system, with user specified ray space discretisation. Forwards simulation, the direction of which is shown in Figure 2 by the blue arrow, means that a ray originates from a point source and travels through the optical system where successful rays will hit a sensor pixel. This method depicts real-world imaging, but can be inefficient since there is no guarantee a ray will hit the sensor. The scene must be discretised into point source locations, with the level of discretisation affecting the quality of simulation. Backwards ray tracing, shown by the red arrow in Figure 2, means rays start at a sensor pixel and are traced until they hit an object in the scene. This is more efficient, particularly for dense scenes, but does not represent the physical process of imaging. Intersection points must be calculated for rays and objects in the scene which can be computationally intensive, however this can be alleviated through discretising space using an octree representation.

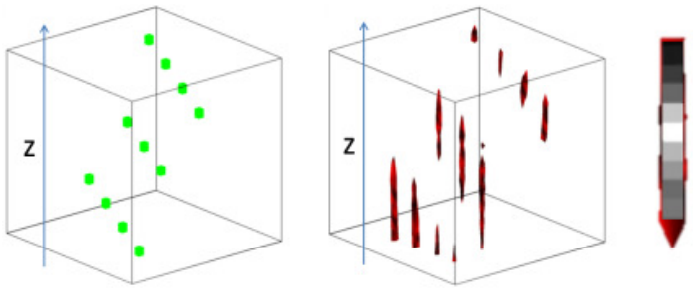
The input scene for the simulation software can be a voxel field or a triangulated mesh. The software allows for customising of meshes, for instance recolouring or combining meshes, which are treated like Lambertian surfaces. Meshes are widely used in computer graphics, and are good representations for many applications. The voxel field input is useful for representing isotropic point sources as found in confocal microscopy or PIV, although an option for user defined ray inputs allows for exploration of scenes which may result from more complex modelling such as monte-carlo simulations through scattering media.

To demonstrate the use of the software for investigating a plenoptic application we have simulated 3-D fluorescence microscopy. Point sources of equal intensity were virtually placed at varying positions in a volume representing optically transparent medium. A microlens array (pitch = 250μm, focal length = 500μm) was placed on the imaging plane of the main lens to which it was matched, and a 1 megapixel sensor was placed on the back focal plane of the microlens array.

3 Results and Discussion

The input volume and reconstruction for the application of 3-D fluorescence can be seen in Figure 3. For tomographic reconstruction the algorithm used to calculate voxel densities was based on Maximum Likelihood using Expectation Maximisation [10], although this choice of algorithm could be optimised for particular applications. It can be seen that the tomography performed on the plenoptic 1.0 data reconstructs the source locations in x-y plane accurately, but with an elongation in the z-direction. This can be attributed to the camera depth of field. Figure 3(c) shows an overlay of the voxel density values from the reconstruction. This shows that, with thresholding or further processing more definite object reconstructions could be estimated. The ability to accurately reconstruct tomographic data from a single plenoptic acquisition is an exciting prospect for practical experiments.

A new software toolbox for the simulation and reconstruction of plenoptic data has been created. The application of fluorescence microscopy has been demonstrated with tomographic reconstruction for point source localisation. The design of the plenoptic camera could be optimised by using the output from rendering methods as a metric, and optimisation of cameras and reconstruction algorithms for specific applications will be the basis of future work. More biomedical imaging problems could be addressed by incorporating specific light transport models into the input scene data. This means there is exciting poten-



(a) (b) (c)

Figure 3: Results from simulation. (a) The input scene of a volume containing 11 distributed point sources. The optical system for the simulation is above the volume (in the positive z direction shown by the arrow). (b) Isosurface of the tomographic reconstruction result. (c) An overlay of the reconstruction and the grayscale value showing the probability of density (white highest, black lowest).

tial to explore and optimise novel applications virtually which can inform future practical implementation.

References

[1] T Fahringer and B Thurow. Tomographic reconstruction of a 3-d flow field using a plenoptic camera. In *42nd AIAA Fluid Dynamics Conference and Exhibit*, 2012.

[2] E Freniere et al. Edge diffraction in monte carlo ray tracing. In *SPIE's International Symposium on Optical Science, Engineering, and Instrumentation*, pages 151–157. International Society for Optics and Photonics, 1999.

[3] T Georgiev and A Lumsdaine. Reducing plenoptic camera artifacts. In *Computer Graphics Forum*, volume 29, pages 1955–1968. Wiley Online Library, 2010.

[4] Tl Georgiev et al. Spatio-angular resolution tradeoffs in integral photography. *Rendering Techniques*, 2006:263–272, 2006.

[5] A Gerrard and J Burch. *Introduction to matrix methods in optics*. Courier Dover Publications, 2012.

[6] M Levoy et al. Light field microscopy. In *ACM Transactions on Graphics (TOG)*, volume 25, pages 924–934. ACM, 2006.

[7] A Lumsdaine and T Georgiev. The focused plenoptic camera. In *Computational Photography (ICCP), 2009 IEEE International Conference on*, pages 1–8. IEEE, 2009.

[8] R Ng. *Digital light field photography*. PhD thesis, stanford university, 2006.

[9] R Ng et al. Light field photography with a hand-held plenoptic camera. *Computer Science Technical Report CSTR*, 2(11), 2005.

[10] Y Vardi, LA Shepp, and L Kaufman. A statistical model for positron emission tomography. *Journal of the American Statistical Association*, 80(389):8–20, 1985.