Efficient Representation of Multi-View Data

Pier Luigi Dragotti and Mike Brookes
Electrical and Electronic Engineering Department, Imperial College London,
Exhibition Road, London SW7 2BT, UK

Abstract

We consider the problem of representing large sets of multiview images efficiently. We first study the properties of such data and recall the notion of plenoptic function. Then propose an efficient segmentation algorithm based on level-set method.

Keywords : Plenoptic Function, Multiview Imaging, Image Segmentation

1. Introduction

Recent advances in sensor network technology are radically changing the way in which we sense, process and transport signals of interest. In this work we consider camera networks and assume that a large number of cameras are monitoring a certain scene from multiple viewpoints. The aim then is to fuse all this data acquired to perform scene interpretation and classification, preferably in an automatic fashion.

Traditional algorithms do not scale properly with the number of cameras and become impracticable when the number of images acquired is large. We therefore aim to develop algorithms that are able to perform classification and scene interpretation when the quantity of data acquired is huge and to exploit the intrinsic redundancy of the information to increase robustness.

The data acquired by multiple cameras from multiple viewpoints can be parameterized with a single function called the plenoptic function. It was first introduced by Adelson and Bergen [1] in an attempt to describe what one sees from an arbitrary viewpoint in space. Such a function requires seven dimensions in order to characterize all the free parameters. In most cases, assumptions can be made to reduce the number of parameters. For instance, Levoy and Hanrahan in 1996 introduced the 4D light field parameterization [5] of the plenoptic function.

From its introduction, the light field parameterization has benefited from a large popularity thanks to the highly structured nature of the images. The idea is basically to setup a 2D camera array that is uniformly sampled. Several of such arrays have already been developed mainly with the goal of performing Image Based Rendering (IBR). As the problem is in essence a sampling problem, the spectral properties of the data have been extensively studied [3, 9, 10]. In these papers, it is shown in various ways that the plenoptic function is approximately bandlimited and therefore most IBR techniques rely on spectral based algorithms. However, we believe spatial based algorithms are more adapted for multiview image representation and interpolation. Following this train of thought we define the following approach.

The approach consists in an accurate segmentation such that the scene can be efficiently represented in a layer based fashion. Indeed, layers, otherwise known as epipolar tunnels or tubes, capture local coherence and make occlusion events

explicit. This observation has already been made in the context of video processing and compression [8]. Here, we make a parallel between multiview data and moving images. Unfortunately, it seems that a robust automatic layer extraction algorithm remains an unsolved problem. While most segmentation algorithms perform the segmentation using two consecutive frames, we believe that the processing should be done on all the images at once in a multidimensional manner in order to fully take advantage of the structured data. Furthermore, we believe that the redundant nature of multiview data should enable a more accurate segmentation. After segmentation, each extracted layer should be properly classified and localized. The classification issue is not considered in this paper and is the subject of future research.

The paper is organized as follows: In the next section we introduce the notion of the plenoptic function and recall its main properties. Then, in Section 3, we discuss a layer-based representation of the plenoptic function and propose a novel segmentation algorithm based on the level-set method. We present some preliminary results in Section 4 and conclude in Section 5.

2. Properties of Multiview Data

The plenoptic function was introduced by Adelson and Bergen in order to characterize general free-viewpoint vision. The idea is to describe the intensity of each light ray that reaches a point in space. It can therefore be characterized by seven parameters namely the visual angle, the wavelength, time and the viewing position:

$$P_7 = P(q, f, l, t, V_x, V_y, V_z)$$

Figure 1 shows the concept where a camera symbolizes the viewing point. Intuitively, we see that the camera has 3 degrees of freedom (dof) for its position in space and itself has 2 dof to address the pixels of the image. With two more parameters, namely time and wavelength, it is possible to characterize any light ray. The general function is difficult to analyze due to its high number of dimensions. Thankfully, certain valid assumptions can be made in order to reduce the complexity. First, we simplify the wavelength into three channels for red, green and blue or one channel for greyscale images. Second, we consider that air is transparent, thus intensity does not change along a light ray unless it is occluded. Third, we limit ourselves to static scenes and drop the time parameter. And finally, restrictions can be made to the viewing position. Indeed, the viewer can be constrained to a plane, a line or a point, removing one, two or three dimensions respectively.

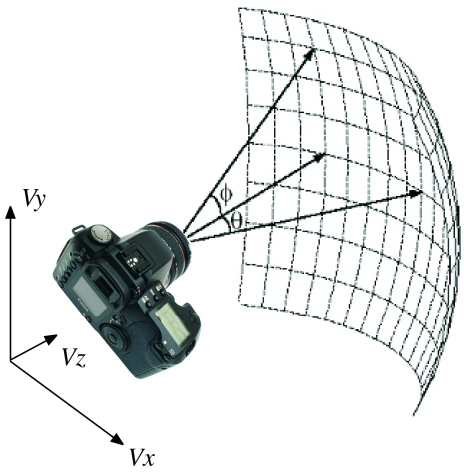

Figure 1: The plenoptic function describes the intensity of each light ray that reaches any point in space at any time. It is therefore characterized by 7 parameters, namely the viewing position, the viewing direction, time and wavelength

It is interesting to notice that common pictures and videos are a particular case of the plenoptic function. Indeed, if we constrain the viewing point on all three axes and remove time and wavelength, we have a 2D function $P_2 = P(q, f)$ which is in fact a picture. The addition of the time parameter creates a well known 3D plenoptic function $P_3 = P(q, f, t)$: the video. The next step is to add dimensions to the viewing position. To this effect, several

representations have been proposed like the light field [5] and the concentric mosaics [7]. Both representations extend the viewing space to finite planes.

At first, this function may appear difficult to work with. However, it is in fact a highly structured function especially in the case where the viewing points are structured.

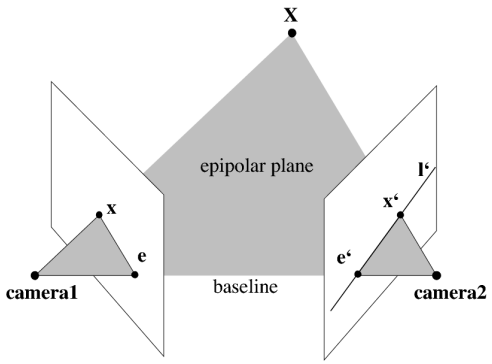

Figure 2: Illustration of epipolar geometry and the epipolar plane. A point in space X is projected onto two image planes with camera centres in camera1 and camera2. The point $\mathbf{x}'$ in the second image is constrained to the epipolar line l'

The geometry that governs the correlations in multiple view imaging is known as epipolar geometry [4]. The basic idea is to correlate the positions in each image of a point in space. Let us consider the simple case where there are two cameras viewing a scene from different locations and there is a point X visible in both views as seen in Figure 2. The rules that govern the relation between the position $\mathbf{x}$ of the point in first image and the position $\mathbf{x}'$ of the same point in the second image are known as epipolar constraints. We call the line that passes through both camera centres the baseline and its intersection with the image planes the "epipoles", $\mathbf{e}$ and $\mathbf{e}'$. Then the constraint is the intersection of the image planes with the plane that contains both the baseline and the point of interest X. This plane, shown in Figure 2, is the "epipolar plane". The intersections of this plane with the image planes are called the epipolar lines l and l'. Suppose we know the location $\mathbf{x}$. Then the location $\mathbf{x}'$ is restricted to the epipolar line l' in the second image. The benefit of this observation for stereo correspondence is that the search for $\mathbf{x}'$ need not be pursued on the whole image. The relation between $\mathbf{x}$ and $\mathbf{x}'$ can be formalized with a single matrix called the fundamental matrix $\mathbf{F}$ [4]. The epipolar constraint may then be represented as a matrix equation

$$\mathbf{x}'^{T}\mathbf{F}\mathbf{x} = 0$$

where $\mathbf{x}$ and $\mathbf{x}'$ are expressed in homogenous coordinates. The advantage of such a description is that $\mathbf{F}$ can be computed from image correspondences alone, without computing any of the cameras' intrinsic parameters. In practice, $\mathbf{F}$ may be found using the 8 point algorithm. We refer the reader to [4] for a more detailed discussion.

Consider the case where the cameras are pointing in the direction perpendicular to the line passing through the camera centres. In this case, the baseline never intersects the image planes and the epipoles are at infinity. The fundamental matrix equation reduces to $y = y'$. Furthermore, when more cameras are involved, the relations between two consecutive images remain the same as long as the cameras are equally spaced. One common setup which follows this layout is the light field. In the next section, we will concentrate on the geometrical properties of this particular case.

2.1 The Light Field Parameterization

The light field representation of the plenoptic function is a 4-dimensional parameterization proposed by Levoy and Hanrahan [5] in order to characterize the case of a planar camera array where all the cameras are pointing in the direction perpendicular to the camera plane. This parameterization was proposed with the objective of performing Image Based

Rendering. The popularity of this representation is due to the simplicity of the parameterization and the layout. The idea is to characterize a light ray with four parameters

$$P_4 = P(u, v, s, t)$$

that relate to the correspondence between the location of the camera on a plane (s,t) and the image plane (u,v) as seen in Figure 3. The distance that separates the two planes is the focal length of the cameras f.

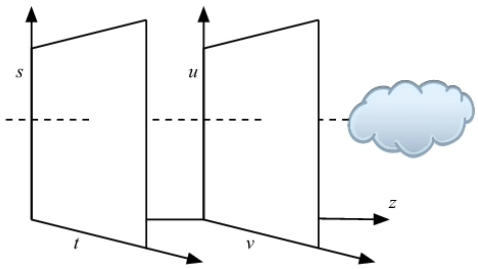

Figure 3: 4D light field (or light slab) parameterization. A light ray is uniquely characterized by its intersection between the camera plane (*s,t*) and the image plane (*u,v*). Each light ray is therefore addressed by its coordinate (*u,v,s,t*).

This parameterization has the property of mapping points onto their epipolar lines. Therefore, the plenoptic domain consists of a collection of straight lines with slopes that are inversely proportional to the distance from the camera plane. In this case, the plenoptic domain is also known as the Epipolar Plane Image (or EPI). This property is observed in Figure 4 where we show the 2D light field and the corresponding plenoptic function. The 2D function is obtained by taking a (t,v) slice of the light field thus fixing s and u. The z-axis represents the depth and the v and t axes represent the focal and the camera planes respectively. Geometrically, we see that a point in space (t, z) is projected on to the focal plane, $z = f$, according to

$$v - v' = (t - t')\, fz^{-1}$$

where t and t' correspond to two different camera locations. Notice that the depth z can be retrieved thanks to the dependence on z^{-1} of the slopes. Furthermore, the intensity along the line remains constant under the assumption of Lambertian surfaces, i.e. surfaces that reflect any incident light uniformly in all directions.

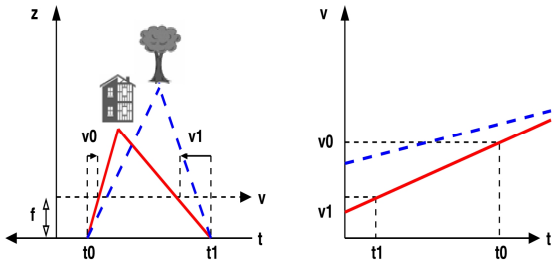

Figure 4: 2D Light field parameterization where *z* represents the depth in the scene, *t* is the camera position axis and *v* is the focal axis of the cameras (positioned in *z=f*)

In the discrete case, the problem becomes a 4-dimensional sampling and interpolation problem. There is not only the sampling in the (u,v)-plane which corresponds to the pixels of a digital camera for instance but also the sampling in the (s,t)-plane which corresponds to the number of camera viewpoints in the array. In [5], the authors render novel views using basic quad-linear interpolation. For example, a new light ray (u_0, v_0, s_0, t_0) is computed from the 16 neighbouring light rays. One main advantage of this rendering is that it can be done without any knowledge of the scene geometry, hence the term "Image Based Rendering". However, as a result, the number of samples needed is very large. This problem is overcome by reducing the signal bandwidth with appropriate pre-filtering, at the expense of image sharpness. The sampling problem and spectral characteristics are discussed in the next section.

2.2 Spectral Analysis of the Plenoptic Function

Sampling of the plenoptic function and view interpolation has until now mainly

been considered in a traditional framework. The idea is to apply the Fourier transform to the signal and sample it according to its spectrum. Chai et al. in [3] found that, assuming the scene is Lambertian and occlusion-free, the plenoptic function is approximately bandlimited. Indeed, the support in the frequency domain is bound by the maximum and minimum depths in the scene irrespective of how complicated it is. Recall from (4) that a point in space is mapped to a line in the EPI with slope inversely proportional to its depth. When the scene is at constant depth z_0 then the Fourier transform is reduced to a line

$$fz_0^{-1}\Omega_v + \Omega_t = 0$$

It is shown in the paper that in the case of a scene with depths bound between $z_{\min}$ and $z_{\max}$ as in Figure 5(a), the spectral support is still approximately bandlimited. An illustration of the spectrum is shown in Figure 5(b). Furthermore, it was later shown by Zhang and Chen in [10] that the assumption of approximate bandlimitedness still holds in the case of scenes with occlusions.

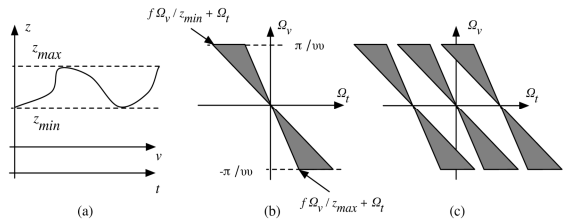

Figure 5: Spectrum of a light field. (a) A scene with varying depth and no occlusions. (b) Idealized spectrum of the EPI. (c) Spectrum of the sampled EPI

Assume we capture a scene with a finite number of cameras. The sampling process causes the replication of the spectrum as shown in Figure 5(c) and aliasing occurs when interpolated viewpoints are rendered. The ways of getting round this problem are the same as in classical sampling theory. One possibility is to use high sampling frequencies such that the aliasing does not occur. This setup implies a high density of cameras along the t-axis.

The observation that the plenoptic function is bandlimited is only a first order approximation. There are numerous reasons why the band is in fact not limited. The first, and most intuitive, is that objects that have non bandlimited textures pasted onto them cause the band of the plenoptic function to be infinite. Furthermore, object boundaries and occlusions also cause discontinuities that have the same effect. Therefore, it seems natural to pursue efficient representations or scene interpolation in a space based fashion rather than a spectrum based one.

3. The Layer-Based Representation of the Plenoptic Function

Most common image representations are based on 'low-level' image processing concepts like the discrete cosine transform and the wavelet transform for example. An ideal 'high level' coding algorithm would recognize objects, however it seems unlikely that such techniques will emerge in the near future. In [8], it suggested that a 'mid-level' layer based representation can prove to be very efficient. Indeed, the knowledge of object and occlusion boundaries is critical in multiple view imaging. Therefore, we believe that the key to efficiently represent multiview data lies in segmentation and constructing a layer based representation of the scene.

In this section, we provide a new algorithm for object segmentation based on video segmentation schemes and the level set method. We emphasize the importance of the capacity of handling occlusions. Throughout the section, we set ourselves in the context of a uniformly sampled light field and assume Lambertian surfaces.

3.1 Representing Scenes with Layers

The layer based representation was introduced in [8] in order to provide a

coding scheme with a certain understanding of the scene. Indeed, it takes into account several aspects like segmentation, depth perception, coherent movement and occlusions. While this representation is proposed for moving images, we apply it to the multiview case. Indeed, the plenoptic function can be represented as a superposition of epipolar layers. The whole scene can then be compressed into a plane plus parallax representation for each tunnel. Furthermore, this representation has the advantage of providing information for view interpolation with coherent occlusion handling. As mentioned in Section 2.2, the plenoptic function is not bandlimited and the sampling process causes aliasing. Spectral based algorithms were used in previous attempts to interpolate new viewpoints. With the advantage of scene segmentation, we can interpolate scenes by interpolating the motion parameters of the objects in a way that makes physical sense as we can follow the epipolar constraints. Thus, there will be no blurring effect around the edges.

The problem of segmenting objects in a scene is well studied but remains ill-posed. Take an outdoor scene for instance. The segmentation is actually quite subjective. Do you segment each branch or leaf of a tree or do you consider the tree as whole? If the camera zooms in, do you change the segmentation to separate each leaf? In the context of efficient representations, our main criteria will be the motion parameters that characterize the correlation of points in multiple views. Object based segmentation in a multiview framework is closely related to video segmentation. For example, consider the case of the 3D light field of a static scene. Then the images put in a stack are the same as a video taken with a camera that is translated along the t-axis of the light field. Therefore, video segmentation algorithms are applicable in our context as well with the added advantage that in our case the motion model is highly structured.

3.2 Level Set Based Segmentation

There are numerous object based segmentation methods for still images and video. In the case of still images, the main criteria to segment objects are intensity gradients. One approach is to use the level set methodology [6] to grow surfaces with a speed inversely proportional to the image gradient. The level set method has been used in region-based video segmentation [2]. The main idea is to use the level set method in order to minimize a certain energy functional which is usually a measure of variance along a motion trajectory. The curve stops at large gradients in the motion boundary map.

The key idea of the level set method is to represent a closed curve $\Gamma(t)$ as the zero level set of a 3D surface $f(x,y,t)$, i.e. $\Gamma(t)=\{(x,y)\,|\,f(x,y,t)=0\}$. The higher dimension function f may be set to be the signed distance function of $\Gamma(t)$. In this case, we have $\|\nabla f\|=1$. The curve is grown according to the partial differential equation

$$f_t + F|\nabla f| = 0$$

where F is the "speed function". This method provides several added advantages [6]. First, the curve may break or merge providing better handling in the case of topological changes. Second, the geometrical properties of the curve like curvature and normal vectors can be computed directly from the level set function. Third and most important in our case, the methodology allows for efficient numerical implementation in high dimensions.

3.3 Segmentation Applied to Multiview Images

In the context of multiview imaging, we replace the time dimension with the t-axis of the light field, representing camera

position, and a similar procedure can be used. In the case of a uniformly sampled and calibrated camera array, the motion model is highly structured. Moreover, the uniformity of the samples provides constant motion parameters throughout the stack of images. Under the assumption of Lambertian surfaces, we have constant intensity values for a point in space. Then a rigid object that is approximately planar should have the same motion parameters for all its points. Therefore, we use a motion estimation algorithm and use the parameters to segment scenes. In the first stages, we consider only a 3D light field comprising a single line of uniformly spaced cameras all pointing in the direction perpendicular to the baseline.

The first step is to define a motion model. Let $I(x, y, t_i)$ be the intensity of pixel (x, y) in the image taken from a camera in location t_i of a 3D light field. If the same point in space can be seen in two consecutive images then we have

$$I(x, y, t_i) = I\big(f(x, y), g(x, y), t_{i-1}\big)$$

where $f(x, y)$ and $g(x, y)$ describe a certain transformation. In the context of multiview imaging, these motion functions can be estimated using epipolar geometry. Recall from Section 2 that the EPI consists of a collection of lines with slopes inversely proportional to the distance between the point and the focal plane. Consider a scene made of planar objects with constant depth. Then the equivalent motion model consists strictly in translations along the x direction and we have

$$I(x, y, t_i) = I(x + p_1, y, t_{i-1})$$

where $p_1 \in \Re$. Figure 6 shows an example of such a scene with two planes. Notice that the plane that is closer occludes the background plane and that its slope in the EPI will be steeper. In the case of slanted planes, another dimension must be added to the motion parameters. The motion model can be adapted to the complexity of the scene resulting in a motion parameter vector, $\mathbf{p}$. In general, we may use any affine transform. In the context of a uniformly sampled 3D light field, the motion parameters remain constant for each pair of consecutive images. Therefore, we have

$$\begin{aligned} I(x, y, t_i) &= I\big(f(x, y), g(x, y), t_{i-1}\big) \\ &= I\big(f\big(f(x, y)\big), g\big(g(x, y)\big), t_{i-2}\big) \end{aligned}$$

and so on for the whole stack. The advantage here is that the parameter estimation is performed over all the images at once since the motion is constant. In practice, we use a modified block matching algorithm for parameter estimation. The minimization of the square error is not performed on two consecutive images but for the whole EPI.

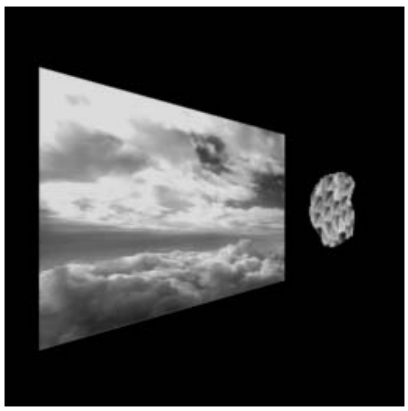

Figure 6: Scene made of two planar objects

3.4 Layer Extraction

The problem of object based segmentation once the motion parameters have been computed boils down to expanding the block as long the pixel intensities along those motion parameters are consistent. The idea is to use the level set method with a speed function that minimizes the variance of the intensity along a motion trajectory. Assume the motion parameters $\mathbf{p}$ for the object have already been computed with a block matching algorithm. Then we extend

the parameters to the whole stack of images and retain the areas where the photo consistency is good. In practice, we use the level set method with a specific speed function for each layer.

Let $x_{\mathbf{p}}(t_i; x, t)$ be a motion trajectory through a stack of N images defined by the motion parameters $\mathbf{p}$. In other words, $x_{\mathbf{p}}$ is the spatial position of a pixel in the image from camera t_i that moved from position x on camera t with motion parameters $\mathbf{p}$. Since the points need not be on the sampling grid, we denote the interpolated intensity as $\tilde{I}(x, y, t)$. Then the speed function

$$F_{\text{layer}}(x, y) = \left(1 + s^{2}(\mathbf{x}, t, \mathbf{p}_{\text{layer}})\right)^{-1}$$

where

$$s^{2}(\mathbf{x}, t, \mathbf{p}) = N^{-1} \sum_{i=1}^{N} \left(\tilde{I}\left(x_{\mathbf{p}}(t_i; \mathbf{x}, t), t_i\right) - m(\mathbf{x}, t, \mathbf{p})\right)^{2}$$

$$m(\mathbf{x}, t, \mathbf{p}) = N^{-1} \sum_{i=1}^{N} \tilde{I}\left(x_{\mathbf{p}}(t_i; \mathbf{x}, t), t_i\right)$$

is the inverse of the variance of the intensity along the motion trajectory. The level set equation becomes

$$\frac{\partial f}{\partial t} = F_{\text{layer}} \left\| \nabla f \right\|$$

for each layer. The speed function relates only to one object tunnel at a time. In order to pass on to the next layer, we remove the current one from the stack of images thus geometrically orthogonalising the process. Notice that in order to perform the iterative segmentation, we have to start with the closest layer and work our way backwards. This order ensures that non-occluded objects are segmented first and further occlusions are explained. The foremost layer in the images is chosen to by finding the one with the maximum disparity. Once all the layers have been extracted, the fully non occluded versions are obtained by averaging the intensity values along the motion of the layer disregarding all the pixels that belonged to other layers, thus $I_{\text{layer}}(x, y) = m(\mathbf{x}, t, \mathbf{p}_{\text{layer}})$ with m as defined above.

4. Simulation Results

A basic simulated scene has been created in order to evaluate the performance of the algorithm. We place ourselves in the case of two planar layers with natural textures pasted on to them. Sixty uniformly distributed viewpoints have been created. Notice that the layers are parallel to the camera baseline and therefore they both undergo translations which are proportional to their depth.

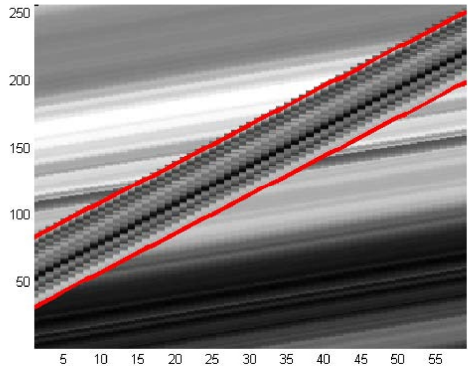

Figure 7: Simulation result. Illustration of the segmentation in the Epipolar Plane Image where the 2D contour (shown in red) is extended according to the motion parameters of the layer. The horizontal and vertical axes correspond to camera position, *t*, and pixel position, *v*, within a single horizontal scan line

Motion parameters have been computed using a least squares algorithm. Here we have tracked two blocks of size 20 by 20 pixels that were manually chosen such that they are entirely lying in their respective layers. As described above, the level set method is used to grow the block with a speed function that minimizes variance along the motion trajectory. In practise, we found that threshholding the speed function produces more accurate segmentations. In this case, we set the initial contour to the boarders of the block. After 9000 iterations of the fast marching algorithm, we obtain

the contour shown in Figure 7. The number of iterations was chosen to yield the best results from a batch of experiments. This 2D contour is then extended to 3D according to the motion parameters of the layer being segmented. The resulting layer extraction is shown in Figure 8.

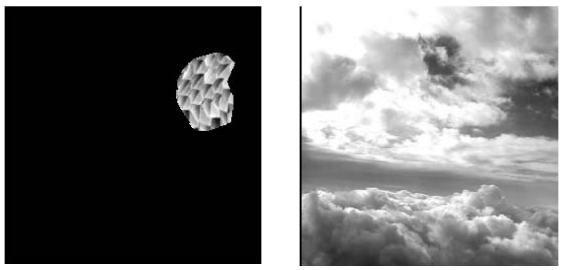

Figure 8: Illustration of the segmented layers

5. Conclusions

We have placed ourselves in a multiview framework and have studied the structure of the images in the case of calibrated light field cameras. We have shown that according to epipolar geometry, the Epipolar Plane Image consists of a collection of lines with slopes inversely proportional to the distance to the camera plane. In this case, occlusions are highly structured events.

We proposed a layer based representation of the plenoptic function and based ourselves on video processing algorithms to segment the scene according to the disparity. We emphasized the importance of multidimensional processing in order to fully take advantage of the structured nature of the data. We have proposed a new segmentation algorithm based on the level set method and have shown encouraging preliminary results. The layer based representation benefits from several advantages. First efficient representation as the scene can be coded as a sum of layers with their motion parameters. Second, view interpolation is straightforward as the motion parameters can be interpolated. Third, we have put forward the potential for performing classification and interpretation of the extracted layers; this will form the focus of later stages of this project.

References

[1] E.H. Adelson and J. Bergen. "The plenoptic function and the elements of early vision". In *Computational Models of Visual Processing*, pages 3–20, MIT Press, Cambridge, MA, 1991.

[2] S. Besson, M. Barlaud, and G. Aubert. "Detection and tracking of moving objects using a new level set based method". In *Proc International Conference on Pattern Recognition*, Vol 3, pages 1100-1105, September 2000.

[3] J.X. Chai, X. Tong, S.C. Chan, and H.Y. Shum. "Plenoptic sampling". In *Proc. International Conference on Computer Graphics and Interactive Techniques (SIGGRAPH)*, pages 307–318, July 2000.

[4] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2004.

[5] M. Levoy and P. Hanrahan. "Light field rendering". In *Computer Graphics (SIGGRAPH'96)*, pages 31–42, 1996.

[6] J. Sethian. *Level Set Methods*. Cambridge University Press, 1996.

[7] H. Shum and L. He. "Rendering with concentric mosaics". In *Computer Graphics (SIGGRAPH'99)*, pages 299–306, 1999.

[8] J. Y.A. Wang and E. H. Adelson. "Representing moving images with layers". *IEEE Trans on Image Processing*, 3(5):625–638, September 1994.

[9] C. Zhang and T. Chen. "Generalized plenoptic sampling". *Technical Report, Advanced Multimedia Processing Laboratory, Carnegie Mellon University*, September 2001.

[10] C. Zhang and T. Chen. "Spectral analysis for sampling image-based rendering data". *IEEE Trans on Circuits and Systems for Video Technology*, 13:1038–1050, November 2003.

Acknowledgements

The work reported in this paper was funded by the Systems Engineering for Autonomous Systems (SEAS) Defence Technology Centre established by the UK Ministry of Defence. This paper contains work done jointly with Jesse Berent (ICL).