3D Dynamic Scene Reconstruction from Multi-View Image Sequences

PhD Confirmation Report

Carlos Leung

Supervisor : A/Prof Brian Lovell (University Of Queensland)
Dr. Changming Sun (CSIRO Mathematical and Information Sciences)

March 25, 2003

Contents

Chapter 1

Introduction

A confirmation report outlining my PhD research plan is presented. The PhD research topic is 3D dynamic scene reconstruction from multiple view image sequences. Chapter 1 describes the motivation and research aims. An overview of the progress in the past year is included. Chapter 2 is a review of volumetric scene reconstruction techniques and Chapter 3 is an in-depth description of my proposed reconstruction method. The theory behind the proposed volumetric scene reconstruction method is also presented, including topics in projective geometry, camera calibration and energy minimization. Chapter 4 presents the research plan and outlines the future work planned for the next two years.

1.1 Motivation

3D scene reconstruction from 2D images have been an old and challenging problem. The task of computer vision and image processing is to be able to bring sight to the computer and provide it with vision analysis. Automatic robots capable of self navigation needs to be able to analyze the world with vision, automated cars would benefit from a vision system, intelligent surveillance requires a vision system. There has been much interest in automatic face recognition in recent years. While researchers have claimed that depth information is not as important as texture in the recognition of faces [31], 3D face recognition has recently been demonstrated to produce promising results [5, 41]. While the current 3D recognition system uses a laser scanned model of the target face, the development of a 3D face recognition system using photographs and videos will be extremely useful.

Being able to restore the depth information of an image and recreate the original 3D scene from images alone have many applications in computer vision. Teleconferencing requires a complete 3D world to be reconstructed. Interactive visualization of remote environments by a virtual camera, virtual modification of a real scene for augmented reality tasks, robot navigation, multimedia computing to generate new virtual views of scenes, virtual reality, games and even

special effects for motion pictures, can all benefit from accurate real-time 3D reconstruction of scenes. While reconstruction of 3D scenes can also be accomplished through the use of specialized hardware such as laser scanners, our focus will be on reconstructing 3D scene using images and photographs captured from cameras or videos.

The reconstruction of a dynamic, complex 3D scene from multiple images has been a fundamental problem in the field of computer vision. While numerous studies have been conducted on various aspects of this general problem, such as the recovery of the epipolar geometry between two stereo images [19], the calibration of multiple camera views [68], stereo reconstruction by solving the correspondence problem [58], the modelling of occlusions [17], and the fusion of stereo and motion [24], little has yet been done to produce a unified framework to solve the general reconstruction problem. Kutulakos and Seitz recently proposed the Space Carving Algorithm aimed at solving the N-view shape recovery problem [30]. The photo hull, the volume of intersection of all views, is determined by computing the photo-consistency of each voxel through projections onto each available image. While these approaches have produced excellent outcomes, apart from the fact that they require a vast number of input images, improvements can be made by imposing spatial coherence, replacing the voxel-based analysis with a surface orientated technique. My aim is therefore to research into more accurate and efficient methods to reconstruct 3D scenes by developing a new approach which combines the advantages of surface evolution and volumetric scene modelling techniques.

1.2 Research Aims

My research aim is to develop techniques and methods that can accurately and efficiently reconstruct a 3D scene given a set of images. My focus will be on advancing the development of volumetric scene modelling techniques by imposing spatial and temporal coherence. A new approach to 3D reconstruction is proposed through the design of a 3D voxel volume, such that all the image information and camera geometry are embedded into one feature space. By customising the volume to be suitable for segmentation, the key idea is to recover a 3D scene through the use of energy minimization and surface evolution. Thus my aim is to not only develop expertise in 3D reconstruction, but also in the area of camera calibration and surface evolution.

I have also a keen interest in the reconstruction of a dynamic 3D scene, recovering both the 3D scene and its motion given a set of multi-view image sequences. While there are many studies in the area of stereo and motion analysis from stereo rigs, we propose the design of a 4D voxel volume by augmenting the design of our 3D voxel feature volume to a 4-D feature space. We present a novel approach to the analysis of stereo motion for the reconstruction of dynamic 3D scenes by including time information into volumetric scene modelling. This novel formulation of the stereo motion problem is a new technique in the analysis of multi-view image sequences.

(a) (b) (c)

Figure 1.1: (a, b) A stereo pair of the parking meter image. (c) Disparity map computed using dynamic programming

1.3 Progress to Date Overview

Much reading and research has been accomplished before I decided to pursue my PhD in the area of 3D reconstruction by volumetric scene modelling. During year 2002, I have investigated topics in the area of face recognition, head modelling, multi-view geometry, stereo motion, camera calibration, trinocular stereo, tensors, and level sets and surface evolution. 2002 has been a year of much reading and exploring. I have experienced coding in the vxl environment and have coded up a DP algorithm to analysis stereo images (Figure 1.1).

Volumetric scene reconstruction have also been investigated. By projecting image sequences of simple motion, such as pure translation, a projective volume and a metric volume based on variance have been constructed. A DP algorithm has been applied to the metric volume in the attempt to segment the volume without surface evolution computations. It has been observed however that a DP analysis on the metric volume is meaningless. Detailed descriptions of the preliminary work accomplished in volumetric reconstruction can be found in Chapter 3.

Chapter 2

Volumetric Literature Review

The reconstruction of a dynamic, complex 3D scene from multiple images has been a fundamental problem in the field of computer vision. Given a set of images of a 3D scene, in order to recover the lost third dimension, depth, it is necessary to compute the relationship between images through correspondence. By finding corresponding primitives such as points, edges or regions between the images, such that the matching image points all originate from the same 3D scene point, knowledge of the camera geometry can be combined in order to reconstruct the original 3D surface.

One approach to the correspondence problem involves the computation of a disparity map, where each pixel in the map represents the disparity of the matching pixels between two images. The optimisation of a cost function is a common approach in order to obtain the disparity map [16, 44, 49, 58]. Taking advantage of the epipolar constraint, which enables the search area to collapse from a 2-dimensional image to 1-dimensional epipolar lines, along with the ordering [66], uniqueness and continuity constraint [34], algorithms have been proposed which compute the disparity map to sub-pixel accuracy. However, when factors such as noise, lighting variation, occlusion and perspective distortion are taken into account, stereo disparity algorithms are still challenged to model accurately discontinuities, epipolar line interactions and multi-view stereo [10, 20]. An evaluation of stereo algorithms using real imagery with dense ground truth can be found at [50, 60].

While the aforementioned techniques operate in 1 or 2D space, there also exists a class of stereo algorithms that operate in 3D scene space. Introduced by Collins [9] and Seitz and Dyer [51], these algorithms, instead of using disparities to compute the depth of an image point, directly project each image into a 3D volume, such that the locations of 3D world points are inferred through analysis of each voxel's relationship in 3D space. Scene reconstruction based on computations in three-dimensional scene space replaces the image-based search

problem into volumetric scene modelling. While stereo algorithms require scene elements to be mostly visible from both cameras, volumetric methods can handle multiple views where very few scene elements are visible from every camera. A survey of methods for volumetric scene reconstruction from multiple images is presented in this chapter. Other reviews of methods for volumetric scene reconstruction can be found at [13, 54].

All the methods described here for building volumetric scene models assume accurately calibrated cameras or images that are taken at known viewpoints. This is necessary in order to recover the absolute relationship between points in space and visual rays, so that voxels in the object scene space can be projected to its corresponding pixels in each image. Image calibration and the computation of the projection matrix is itself a very challenging problem and a large number of literature has been devoted to the recovery of camera geometry [19]. Most volumetric techniques also assume that the surfaces are Lambertian, that is surfaces reflect light equally in all directions, such that the radiance observed of a 3D point is independent of its viewing direction. Apart from the work of Szeliski [61] and DeBonet [3], who considered the modelling of partly transparent objects, nearly all the volumetric scene reconstruction methods assume that the object surfaces are entirely opaque.

2.1 Reconstruction via Voxel Occupancy

Early works in volumetric model reconstruction from images employ techniques that reconstruct the 3D object based on geometric intersection. By determining voxel occupancy, the task is to decide with the scene constructed as a set of 3D voxels whether each individual voxel is empty or occupied. These algorithms differ from stereo and optical-flow techniques since window-based image correlation and image correspondence is not performed in the reconstruction process, but are found during the scene traversal through voxel projections. The most typical solution is by silhouette intersection, either from multiple views of a single object [38, 39] or using a single camera with the object rotating on a turntable [42], to approximate the visual hull of the imaged object. Laurentini [32] described the visual hull of an object as the best approximation, the maximal shape obtainable in the limit by an infinite number of silhouettes captured from all viewpoints, that gives the same silhouette as the actual object outside the convex hull of the object. It is the intersection of all back-projected silhouette cones.

A silhouette image is a binary image that represents whether an image point, projected as a visual ray from the camera center, intersects the imaged object's surface in the scene. Each pixel is either classified as a silhouette in the foreground, or as belonging to the background. The idea behind geometric intersection and shape from silhouettes is that since each silhouette point defines a ray in scene space that intersects the object at some unknown depth along the ray, the union of all such rays should produce a visual hull in which the actual object must lie within. Thus the visual hull is always guaranteed to enclose the true

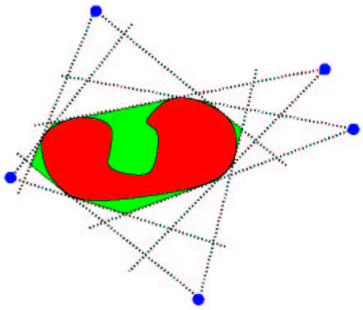


Figure 2.1: A surface (red) is viewed from five cameras (blue), and the visual hull is illustrated in green. Notice how the concavity cannot be reconstructed using silhouettes intersection. Image courtesy of [11]

object. A typical method is to segment each image into foreground and background, such that the geometric intersection of all back projected foreground regions form the inferred visual hull.

The earliest work on volumetric modelling through a visual hull description began with Martin and Aggarwal [35]. In order for the scene space traversal to become more efficient a coarse to fine hierarchy was explored and an octree representation of the visual hull was developed [8]. The octree representation begins with an initial cube that encloses the entire scene. Each voxel is projected into the images to determine its voxel occupancy. If a projected voxel does not intersect the silhouette in any image, it is marked transparent and removed from analysis. Voxels that intersect only silhouette pixels in all images are marked opaque, while voxels that intersects both background and foreground are subdivided into octants, and each sub-voxel is recursively processed. Due to its efficient description of volumetric scene space, many methods used the octree representation when computing shape from silhouettes [59, 47, 57, 53].

Shape from voxel occupancy and silhouette intersection techniques has a major disadvantage in that they fail to exploit the consistency of the scene between different views. It was noted qualitatively by Martin and Aggarwal [35] and Srinivasan [56] that the visual hull was in not an accurate or complete representation of the 3D surface since it depends on the region exposed to the viewpoint. Reconstruction from silhouettes alone can in general only accurately reconstruct convex objects and cannot sufficiently describe surface concavities [32] (see figure 2.1 for illustration).

2.2 Reconstruction via Photo-Consistency

Apart from exploiting the geometric relationship between the available images, introduced by Seitz, [51], color consistency between images can also be used in

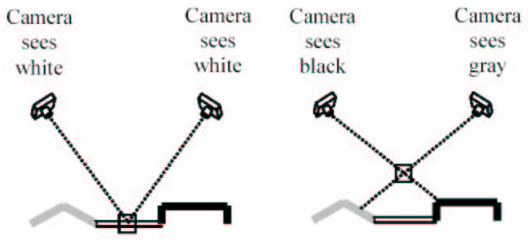


Figure 2.2: Voxels on the object surface are photo-consistent, while voxels not belonging to the surface do not have their projections agreeing. Image courtesy of [54]

order to guide the reconstruction. When the input images are grayscale or color rather than a binary image processed from shape from silhouette methods, the additional photometric information can be extremely useful in improving the 3D reconstruction. Under a Lambertian framework, where surfaces are assumed to reflect light equally in all directions such that the radiance observed of a 3D point is independent of its viewing direction, image points from different images originating from the same 3D point must theoretically have the same color. Thus the color consistency constraint can be applied by defining a point as photo-consistent with a set of images if, for each image in which the point is visible, the image irradiance of that point is equal to the intensity of all the corresponding image pixels. Through this constraint, points on the surface can be distinguished from points not on the surface as illustrated in figure 2.2.

Voxel Coloring is an algorithm introduced that demonstrates that 3D reconstruction can be achieved through color consistency alone without the need to compute volume intersection [51]. It begins with a volume of opaque voxels that encapsulates the scene to be reconstructed. Every voxel is subsequently back projected to each image from which the voxel is visible. This is to account for occlusions. The photo-consistency of all valid projections are evaluated using a metric, and voxels which exceeds a certain threshold or deemed inconsistent are carved away and termed transparent. The process occurs recursively for all voxels until all remaining opaque voxels are photo-consistent. Since each voxel is visited exactly once, the space and time complexities of voxel coloring is linear in the number of images. This major advantage enables a large numbers of images to be processed at once.

Modelling occlusions is a major challenge for all 3D reconstruction algorithms. Voxel Coloring attempts to overcome occlusion by first computing the visibility of each voxel. An ordinal visibility constraint is introduced so that voxels closest to the camera can be visited first. The near to far ordering relative to the camera allows voxels that are occluded to be evaluated for photo-consistency after the voxel that is occluding it has been visited. Therefore, if the voxel occluding this voxel has been termed photo-consistent, any subsequent

```
1   set all voxels opaque
2   loop {
3         AllVoxelsConsistent = TRUE
4         for every opaque voxel V {
5               find the set S of input image pixels from which V is visible
6               if S has consistent color {
7                     assign V the average color of all pixels in S
8               } else {
9                     AllVoxelsConsistent = FALSE
10                    set V to be transparent
11              }
12        }
13        if AllVoxelsConsistent = TRUE
14              quit
15   }
```

Figure 2.3: Pseudocode of the Space Carving algorithm. [54]

voxel behind it as viewed from a camera can be termed occluded. However, in order to enforce this single scan technique, typically all the cameras need to be placed on one side of the scene so that scanning can occur along planes that are successively further away from all camera viewpoints.

In order to overcome the ordinal visibility limitation, an extension to Voxel Coloring is introduced known as Space Carving [30]. It allows arbitrary camera placements by using multiple scanning. Since if a random voxel is carved without the ordinal visibility constraint, it is possible that the visibility of other voxels could also be altered, invalidating previous consistency tests. Space Carving repeats the consistency check until all carving has been completed. A pseudo code of the algorithm is presented in figure 2.3. Kutulakos has also proved that the algorithm is conservative, that is it will never carve a voxel that it should not have carved if a suitable consistency measure is used. Since the algorithm only turns opaque voxels into transparent and never vise versa, the voxels can only become more visible. A pixel that can view the voxel will be a subset of all the pixels that can view that same voxel at a later time. This is possible only if the consistency measure is monotonic, such that if a set of pixels are tested to be inconsistent, then any superset containing these pixels will also be classified inconsistent. The algorithm has further been demonstrated to produce a unique color consistent model, known as the photo hull, which is a superset of all the possible consistent models. To account for inaccurate camera calibrations an Approximate Space Carving algorithm has also been introduced which relaxes the photo-consistency constraint, such that a pair of pixel p and p' is claimed photo-consistent if any pixel within a distance r from p' has the same color as p [29].

Variations of Voxel Coloring and Space Carving have since been introduced. Culbertson [12] introduced an extension known as Generalized Voxel Coloring (GVC) that computes the exact visibility of each voxel. Different data structures have also been introduced to effectively account for the visibility of voxels. Item

buffers records for every pixel in an image the voxels which are visible [64]. If a voxel is carved, the item buffer is recomputed. Layered depth images [52] records for every pixel a depth-sorted list of all the surface voxels that projects to itself. This linked list structure, although it consumes more memory, provides an effective method to determine the changed voxels' visibility when a random voxel is carved away. While all the above mentioned methods determines the consistency of a voxel based on all images simultaneously, a multi-hypothesis voxel coloring technique has also been introduced that performs voxel removal one image at a time. This formulation is essentially a variation of the ordinal visibility constraint allowing for a front to back scan for each camera viewpoint.

An interesting extension to the Voxel Coloring algorithm is the 6D hexel space introduced by Vedula to account for temporal coherency in time-varying scenes [62]. The 6D space is formed by connecting two time consecutive 3D voxel space together. The definition of photo-consistency requires that a hexel is consistent in color not only from all viewpoint, but also from both time instants. The output remains two reconstructed 3D voxel space, but the added constraint has been shown to produce more promising results.

A significant drawback of the Space Carving algorithm is that the quality of any reconstructed model is limited by the resolution of the voxels chosen. The computation time of the algorithm is subsequently also linked to the voxel resolution. Furthermore, all the techniques that reconstruct the 3D scene through Voxel Coloring does not explore the spatial coherence between voxels. Since these methods traverse the volume making hard decisions regarding the occupancy of each voxel, the ambiguity inherent in many image data makes these decisions unreliable and undoing such decisions is difficult.

2.3 Reconstruction via Energy Minimization

Another class of 3D reconstruction techniques involves the minimization of an energy function. By incorporating smoothness and visibility constraints into the energy function as well as a photo-consistency term, the 3D reconstruction can be enhanced to incorporate spatial coherency in the 3D reconstruction. Energy minimization has been a very popular approach in solving the stereo matching problem. While traditionally, simulated annealing has been used to solve the energy minimization problem, powerful algorithms based on graph cuts and variational calculus have demonstrated to be most practical and efficient while producing promising results for stereo [50, 60].

Minimizing an energy function using graph-cuts have been studied by researchers in the last few years [4, 20, 49]; and only a few researchers have attempted to apply graph cuts to reconstruct 3D scenes [25, 21, 55, 48]. It differs from Voxel Coloring techniques in that it avoids being trapped by early hard decisions and projection ambiguities are resolvable that are spatially coherent. Of the different graph cuts algorithms applied to scene reconstruction, [48] did not consider visibility and had a spatial smoothness term that was not discontinuity preserving and thus have a tendency to produce over-smoothed

results. [55] computes the global minimum of an energy function that was an alternative to silhouette intersection and thus consequently does not consider photo-consistency. [21] does not treat input images symmetrically and subsequently computes a disparity map with respect to a single camera; while [25] considers a selection of pairs of interacting cameras and can compute a disparity map for each camera. [61] also compute disparity maps for each camera but does not rely on graph cuts with an emphasis on two-camera stereo. While graph cuts are extremely powerful in that it can be formulated to compute the optimal solution avoiding being stuck in local minimas, not all energy functions can be embedded into a graph for minimization. Kolmogorov has presented a paper that discusses which energy functions can be minimized using graph cuts [26].

Apart from using graph cuts, energy minimization can also be accomplished by transforming the energy function into a partial differential equation (PDE) via variation calculus and computing a solution to the PDE. Faugeras presented a version of this method by adapting the level set technique to the scene reconstruction problem [14]. With the initial surface encompassing the scene, level sets was used to evolve the surface towards the objects in the scene. The advantage of level sets, although it has a tendency to be trapped in local minimas, is that it is fast and well-developed through the work of Osher and Sethian [45]. This method can also model occlusions through the computation of the visibility of each zero level set at each evolution step. Thus only cameras with an unoccluded viewpoint of the zero level set surface contribute to the computation of the speed function for the next time step. Unlike Voxel Coloring, a continuous surface and an analytic framework of the surface propagation can be modelled.

Many of these techniques however, that aims to reconstruct the 3D scene through energy minimization, have applied the method to pair-wise feature matching of the available images. The limitation is that pair-wise matching techniques can at best only reproduce a 2.5D sketch of the scene and cannot produce a true 3D reconstruction. Our research aims at combining the advantage of both volumetric reconstruction techniques and energy minimization techniques to produce an accurate reconstruction of a 3D scene.

Chapter 3

Volumetric Scene Reconstruction

A common approach to stereo reconstruction is the optimisation of a cost function, computed by solving the correspondence problem between the set of input images. The matching problem involves establishing correspondences between the views available and is usually solved by setting up a matching functional for which one then tries to find the extrema. By identifying the matching pixels in the two images as being the projection of the same scene point, the 3D point can then be reconstructed by triangulation, intersecting the corresponding optical rays.

Our proposed method to recover the 3D structure is a combination of volumetric scene reconstruction techniques and energy minimization techniques. Assuming a pinhole camera model, the 3D voxel volume is created by projecting all of the images into a 3D polyhedron, such that each voxel contains a feature vector of all the information contained in each camera view. A feature vector can include, for example, the RGB values of the voxel's projection into each camera image, the gradient of the image or information relating to the projected pixel's neighborhood. The construction of our voxel volume is similar to the concept proposed by Kimura, Saito and Kanade [23] in recovering the 3D geometry from the camera views. Their method however is restricted to only three orthogonal images. From the voxel volume, a metric volume can then be derived from this voxel space to become the input to the segmentation stage. Cost functions such as the variance between the projections or even a probability density function can be used. An energy function is then constructed so that the 3D object surface is the global minima of the cost function. By evolving a surface under the level set method towards the global minima, the 3D scene can be reconstructed.

Our work is most similar, but differ in a lot of ways, to Roy and Cox's maximum flow formulation of the stereo problem and Faugeras' work on stereo vision using Level Set methods. Roy and Cox [49] sdeveloped an algorithm for

solving the multi-view stereo correspondence problem by stacking the candidate matches of range disparity along each epipolar line into a cost function volume. Maximum flow analysis is then used in order to determine the disparity surface. While this approach computes the scene reconstruction through volume analysis and can provide a more accurate and coherent depth map than the traditional line-by-line stereo, these methods remain dependent on the computation of disparities and the accuracy of the matching correspondence stage. Although the optimisation of the cost function is performed in a three-dimensional space, the computation of a disparity surface remains only a 2.5-D sketch of the scene [34]. Our idea to use level sets to evolve a surface towards the 3D object to be constructed is similar in idea to Faugeras' approach [14]. Faugeras' definition of the energy function however is based on the cross correlation between different pairs of the input images. Our formulation aims to evolve the surface under the projective 3D volume and consider all the information available from all images at once rather than looking at pair-wise combinations. Due to a very different formulation of the problem, the method in which we aim to model occlusion is also different.

Our reconstruction algorithm can be split into three separate modules and the rest of this chapter will be devoted to a detailed description of each of these modules. The first module is the computation of the projective voxel volume. Camera calibration and the recovery of the camera geometry is required in order to determine the projection camera matrices for each camera necessary for projection. The second module involves the computation of the metric volume. And the third module is the segmentation stage where a surface representing the 3D scene is recovered.

3.1 3D Voxel Volume

The construction of the voxel volume is purely based on image projections and thus does not require the solution to the correspondence problem in order to produce a dense reconstruction. Unlike algorithms that use disparity maps to guide the 3D reconstruction, dense feature correspondence or area-based matching of every pixel is no longer necessary. However, while the computation of the 3D volume does not require dense correspondences, feature correspondence is needed in establishing the camera geometry and for the purpose of camera calibration. The accuracy of the projections, the reliability of the volume and the success of the reconstruction are highly dependent upon robust and accurate camera calibrations. As noted by Medioni [37], there is an imperative need in multi-view stereo for accurate camera calibration and consistency of matches between multiple-views. Therefore full calibration information needs to be provided with the image set or techniques similar to [69] must be used to obtain the calibration parameters.

While after creating the 3D projective voxel volume, the images are theoretically no longer needed since the volume encapsulates all the information available from all the view, the massive use of memory makes the storage of this

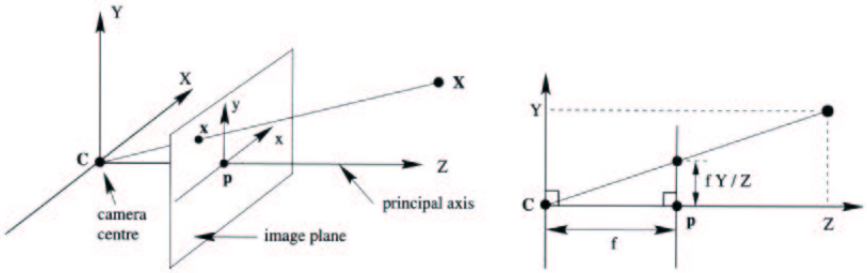


Figure 3.1: The pinhole camera model

huge dimensional space not efficient. This volume is never practically stored and the metric volume, which collapses all the projection information into a meaning measure of consistency between the views available, generally follows the projections. The metric volume is the volume that is stored to be passed onto further processing where the 3D scene is reconstructed.

3.1.1 Camera Calibration and the P-Matrix

Consider the projection of a 3D point in world space onto an image. Assuming a pinhole camera model and setting center of projection as the origin of a Euclidean coordinate system with the image plane placed at $z = f$, we obtain the configuration in figure 3.1. By similar triangles, we can see that a 3D point $P = (x, y, z)^\top$ is mapped onto the image at image coordinate p

$$(x, y, z)^\top \mapsto (fx/z, fy/z)^\top$$

Represented as homogeneous vectors, the mapping from Euclidean 3-space $\mathbb{R}^3$ to Euclidean 2-space $\mathbb{R}^2$ can be expressed in matrix multiplication as

$$\begin{pmatrix} fx \\ fy \\ z \end{pmatrix} = \begin{bmatrix} f & & & 0 \\ & f & & 0 \\ & & 1 & 0 \end{bmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

The above formulation assumes that the origin of coordinates in the image plane coincides with the principal point, to account for this offset, where the coordinate of the principal point occurs at $(x_0, y_0)^\top$, the mapping

$$(x, y, z)^\top \mapsto (fx/z + x_0, fy/z + y_0)^\top$$

can be rewritten as

$$\begin{pmatrix} fx + zx_0 \\ fy + zy_0 \\ z \end{pmatrix} = \begin{bmatrix} f & & x_0 & 0 \\ & f & y_0 & 0 \\ & & 1 & 0 \end{bmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

If we define a matrix $\boldsymbol{K}$, known as the intrinsic camera matrix since it describes the internal camera parameters,

$$\boldsymbol{K} = \begin{bmatrix} f & & x_0 \\ & f & y_0 \\ & & 1 \end{bmatrix}$$

then the mapping from $\mathbb{R}^3$ to $\mathbb{R}^2$ can be written as

$$p = \boldsymbol{K}[I|0]P$$

Furthermore in general, the camera coordinate system is embedded inside a world coordinate frame and the origin of the camera center, C, does not necessary coincide with the world coordinate origin. We realize the the P that we have been referring so far is expressed with respect to the camera coordinate system, and computations are generally performed with respect to the world coordinate system. The 3D point P relates to the world coordinate system by $P = \boldsymbol{R}(P_w - C)$, where $\boldsymbol{R}$ is the rotation matrix representing the orientation of the camera coordinate frame. In matrix form this would become

$$P = \begin{bmatrix} \boldsymbol{R} & -\boldsymbol{R}C \\ 0 & 1 \end{bmatrix} P_w$$

Combining this with the intrinsic camera matrix, we obtain

$$p = \boldsymbol{K}\boldsymbol{R}[I| - C]P_w$$

The term $\boldsymbol{K}\boldsymbol{R}[I| - C]$ is the camera projection matrix. It is however more convenient not to explicitly describe the camera center and represent the transformation between the coordinate system as a rotation followed by a translation, giving rise to the more common form of the projection matrix $\boldsymbol{P}$

$$\boldsymbol{P} = \boldsymbol{K}[\boldsymbol{R}|\boldsymbol{t}]$$

There are many methods that can be applied in order to recover the projection matrix. The choice of method mainly depends on the amount of information that is already known. For example, knowledge of the camera's internal parameters such as its focal length and its skew, or knowledge of the exact motion between different views such as its translation and rotation can dramatically change the accuracy and the technique used to compute the projection matrix.

Given only one image, computation of the projection matrix requires the exact 3D location of a number of points, ground truth. Since the projection matrix contains 12 entries and 11 degrees of freedom (for projection up to a scale factor), the most direct method is to obtain a minimal of $5\frac{1}{2}$ 3D point to 2D pixel image correspondences and solve the 11 equations obtained. Each point correspondence leads to two equations and the $\frac{1}{2}$ implies only either the x-coordinate or y-coordinate needs to be matched for the sixth image point. If data contains noise, an over-determined solution can be computed for n $\geq$ 6 by minimizing either an algebraic or geometric error.

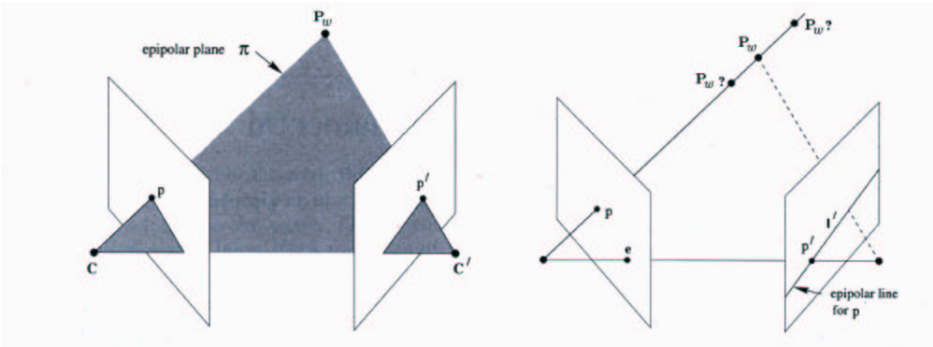


Figure 3.2: Epipolar Geometry. Image courtesy of [19].

However, we do not always and cannot assume that we will have knowledge of the exact 3D location of the points in the scene. In fact, the most common input is a set of images, either multiview or a single image in motion (such as a video sequence), and 3D reconstruction is to be performed based on these input image sequences. Although it is not possible with single images, it is possible to recover the projection matrix of multiple images without the prior knowledge of the scene, by exploiting the corresponding relationship between multiple views.

Let us consider the case of a pair of stereo images - two-view geometry. As depicted in figure 3.2, suppose a 3D point P_w is viewed from two images, at image point p in the first image and p' in the second image. The camera centers together with P and its images p and p' form a plane known as the epipolar plane. The epipoles, e and e' are defined as the image of the camera centers of one image in the other image, and is the point of intersection of the baseline (the line joining the two camera centers) with the image plane.

An important observation is that the image point p back-projects to a ray in 3D space defined by its camera center C. While the 3D point P_w must lie on this ray, this ray is also imaged in the second view as a line l'. Subsequently, the image of P_w in the second view, namely p' must therefore according to the epipolar geometry, lie on the line l'. This line, known as the epipolar line can be defined as the intersection of the epipolar plane with the image plane. All possible epipolar lines thus passes through the epipoles, since each possible epipolar plane which give rise to the epipolar lines must include the baseline. This property enables the derivation of the fundamental matrix which governs and encapsulates the relationship between stereo views.

Given a point p', the epipolar line l' that passes through p' must also pass through e' such that $l' = e' \times p$. By expressing the cross product using the corresponding skew-symmetric matrix, we obtain

$$l' = [e']_{\times} p'.$$

Since p and p' are both images of the same 3D point P_w lying on the epipolar

plane, the set of all corresponding pair are projectively equivalent and thus exist a 2D homography H_π mapping each p to p'. Substituting $p' = H_\pi p$ into the previous equation, we obtain

$$l' = [e']_\times H_\pi p = \boldsymbol{F} p$$

where $\boldsymbol{F} = [e']_\times H_\pi$ is defined as the fundamental matrix. It is a 3×3 matrix of rank 2, and represents the mapping from the 2 dimensional projective plane of the first image to the pencil of epipolar lines through the epipole e'. Now since p' must lie on epipolar line l', $p'^\top l' = 0$. Substituting $l' = \boldsymbol{F} p$ into the equation, we obtain an important relation

$$p'^\top \boldsymbol{F} p = 0.$$

$\boldsymbol{F}$ is a 3×3 rank 2 homogeneous matrix with 7 degrees of freedom (ignoring common scaling and since $\det \boldsymbol{F} = 0$). Thus given 7 or more corresponding pairs, it is possible to compute the fundamental matrix $\boldsymbol{F}$.

The fundamental matrix is useful since it can be used to compute a pair of corresponding camera matrices. Given the fundamental matrix $\boldsymbol{F}$ for two images, the camera matrices may be chosen as

$$\boldsymbol{P} = [I|0] \quad \textit{and} \quad \boldsymbol{P'} = [[e']_\times \boldsymbol{F}|e'].$$

A proof of this formula can be found in [19] for interested readers.

The significance of this equation is that given two uncalibrated images, a reconstruction up to a scaling factor or similarity transformation is possible without the need for ground truth 3D data of the scene. Thus any scene, given two or more views, can have its camera projection matrices recovered for projective construction. Through point correspondences between the images, the fundamental matrix can be recovered allowing the computation of the projection matrix for each image. Notice however that by setting one of the projection matrix as $[I|0]$, it is essentially setting the camera center corresponding to that image as the base camera. All subsequent views are described relative to the base view.

3.1.2 Voxel Projections

The projection matrix, $\boldsymbol{P}$, describes the mapping from 3D scene space onto a 2D image. It projects a 3D point in the world coordinate frame onto an image plane relative to a specific camera. Since $\boldsymbol{P}$ only describes the transformation from Euclidean 3-space $\mathbb{R}^3$ to Euclidean 2-space $\mathbb{R}^2$, we need to include two extra matrices to describe the mapping onto the Euclidean 3-space and the mapping into pixel coordinates in the creation of our projective voxel volume. The projective volume contains a collection of voxels, thus a mapping is required to determine the world coordinate of each voxel before it is possible to compute its projection onto the images. After projecting the 3D location of the voxel onto the image plane, a further mapping is required to determine the pixel

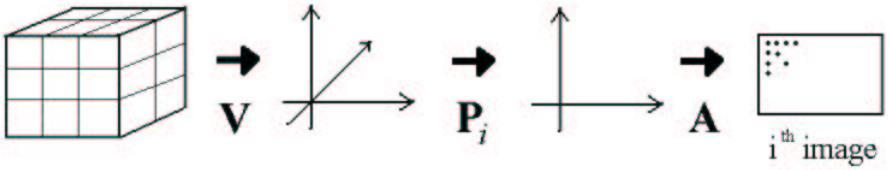


Figure 3.3: A flowchart of mappings from the Voxel Volume to Image Pixels

coordinates of the projection. This matrix that maps image coordinates onto pixel coordinates is similar in construction to the camera intrinsic matrix $\boldsymbol{K}$ described earlier. Figure 3.3 describes the mappings required to project a voxel in the projective volume onto a pixel in the image.

We briefly present the general framework of projective geometry required to construct the 3D voxel volume. A set of N input images $I_1, \ldots, I_n$ of a 3D scene are projected from N cameras $C_1, \ldots, C_n$, as depicted in Figure 3.4 with $N = 3$.

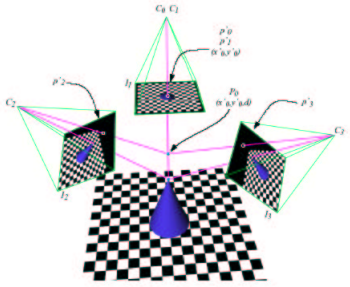


Figure 3.4: Multi-View Projection. 3D point P_0 is projected onto Images I_1, I_2, I_3. (Image courtesy of S. Roy and I. J. Cox [49])

In our formulation, we will assume a pinhole camera model and that all surfaces are Lambertian (i.e. the radiance observed of a 3D point is independent of viewing direction). The projective coordinate of a 3D point P_w in world space is expressed with homogeneous coordinates as

$$P_w = \begin{bmatrix} x_w & y_w & z_w & 1 \end{bmatrix}^T$$

while the projective image coordinate of a pixel in image i is

$$p_i = \begin{bmatrix} x_i & y_i & z_i \end{bmatrix}^T$$

such that the corresponding pixel coordinate p_i' of the projected point p_i can be

obtained by applying a homogenising function H where

$$H(\begin{bmatrix} x \\ y \\ z \end{bmatrix}) = \begin{bmatrix} x/z \\ y/z \end{bmatrix} \tag{3.1}$$

Given the volume of interest of the 3D space for reconstruction, we can obtain each voxel's feature vector by projecting every P_w in the 3D volume onto each of the N images available. With f features for every pixel in the image, each voxel will contain an $f \times N$ matrix, such that the collection of all voxels will contain all the information all the images. In other words, given the 3D voxel volume, all the processing and analysis can be achieved without the need of the original images. We define 4 matrices to describe the projection from a voxel in the volume to a pixel in the image.

Given a volume of M voxels, each voxel will be indexed by its voxel coordinates, $v_m = [v_a, v_b, v_c, 1]^T$, where v_a, v_b and v_c will range from 1 to the dimensions of the volume. The extra parameter appended at the end of the voxel coordinate is for consistency with the augmented homogeneous coordinate in projective space. To transform from voxel coordinates to 3D world coordinates, we compute

$$P_w = \boldsymbol{V} v_m$$

where

$$\boldsymbol{V} = \begin{bmatrix} \boldsymbol{I_3 k} & \boldsymbol{t_v} \\ \boldsymbol{0^T} & 1 \end{bmatrix}$$

and $\boldsymbol{I_3}$ is the 3×3 identity matrix, $\boldsymbol{t_v}$ the translation vector for specifying the world coordinate of the voxel volume's origin, and $\boldsymbol{k} = [k_x, k_y, k_z]^T$ is the stride in each voxel dimension, i.e. the number of units between each voxel in 3D world space. The choice of $\boldsymbol{k}$ and $\boldsymbol{t_v}$ is dependent on the resolution desired for the voxel volume and the origin of the volume of interest respectively.

From world coordinates, the classical 3×4 perspective projection matrix, $\boldsymbol{P}$, can be applied to obtain the projection of the 3D world point in image coordinates. In the case where we define the optical centre of the base camera, C_0, to coincide with the origin of the world coordinate system, the projection matrix will be simplified to be

$$\boldsymbol{P_0} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

subsequently, we can define a transformation matrix $\boldsymbol{W_i}$

$$\boldsymbol{W_i} = \begin{bmatrix} \boldsymbol{R_i} & \boldsymbol{t_i} \\ \boldsymbol{0^T} & 1 \end{bmatrix}$$

with rotation $\boldsymbol{R_i}$ and translation $\boldsymbol{t_i}$ to define the position and orientation of camera C_i relative to the base camera, C_0. The relative positions and orientations of each camera i is determined by a calibration procedure. Thus for C_i,

the projective projection matrix will be

$$\boldsymbol{P_i} = \boldsymbol{P_0 W_i}$$

From the image coordinates, the pixel coordinates of a projective point can be recovered up to a scaling factor, given knowledge of the internal parameters of the camera. Neglecting radial distortion from calibration, a matrix of intrinsic parameters can be computed such that

$$\boldsymbol{A} = \begin{bmatrix} -f_x & 0 & o_x \\ 0 & -f_x/\alpha & o_y \\ 0 & 0 & 1 \end{bmatrix}$$

where f_x is the focal length in effective horizontal pixel size units, α the aspect ratio (i.e. the vertical to horizontal pixel size), and (o_x, o_y) the image centre coordinates.

Combining all the described matrices, each voxel can be projected into each image i by computing

$$p_i = \boldsymbol{A P_i V} v_m$$

From the obtained projective pixel coordinates, the actual pixel coordinates can be obtained by applying the homogenising function described in Eq. 3.1.

3.1.3 Progress to Date

A projective volume have been built for two image sequences. The Povray sequence and the 'Head and Lamp' sequence from Tsukuba University. Camera calibration is a challenging problem in itself and currently, since my focus is on testing the validity of the proposed reconstruction method, work will begin initially with image sequences that have P-Matrices provided. The Povray sequence and the 'Head and Lamp' sequence is used because the motion between successive images is a pure translation, resulting in a simple derivation of the appropriate P-Matrices.

The volumes are present in Figure 3.6. Notice that each successive plane is a zoomed in version of the previous plane due to the projective nature of the volume.

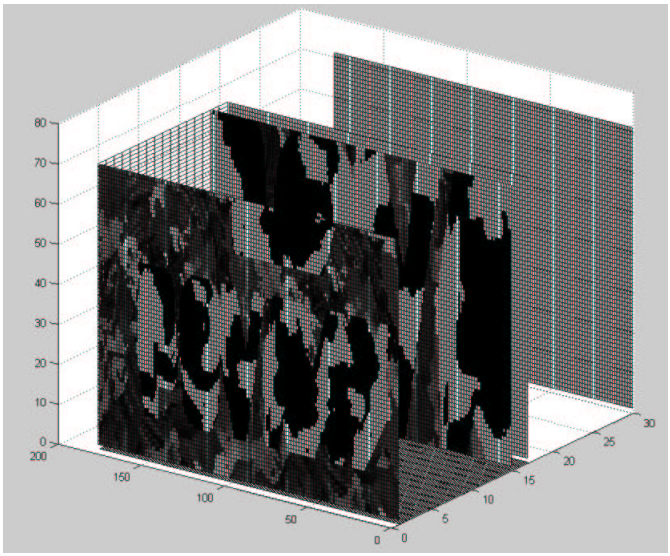


Figure 3.5: Projective volume projection for one of the image in the Povray sequence

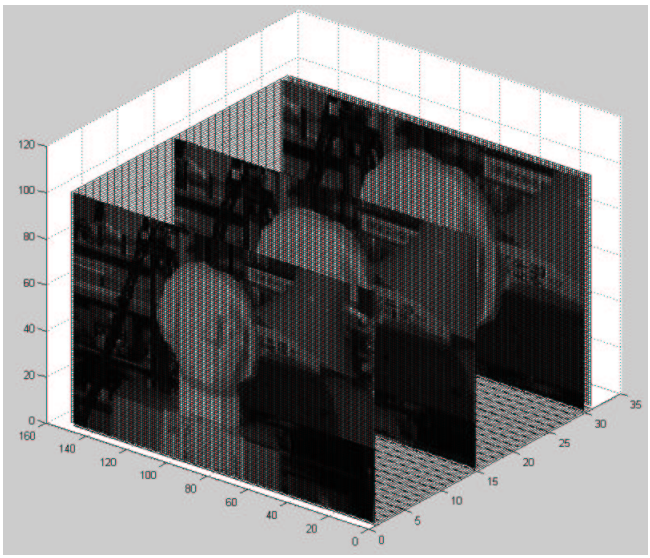


Figure 3.6: Projective volume projection for one of the image in the 'Head and Lamp' sequence

3.2 Metric Volume

After the projective volume is determined, a metric volume condensing the feature vectors of each voxel into a meaning measure or matching functional is required. While it is very important for the camera calibration process to determine the correct camera projection matrices so that the projective volume can be correct constructed, it is also very important to select an appropriate metric such that the correct measure can be computed. Accurate projections provide the necessary information for each voxel, so that each voxel can confidently locate the pixels from which it is back-projected to. Given all the information, feature vectors, it is up to the metric volume stage to analyse the obtained information and to decide whether a voxel is part of the 3D object scene. Volumetric scene reconstruction techniques recover depth information by labelling a voxel with either transparent or opaque, or in the case of a ternary decision, as either transparent, opaque or unseen. In the end, the most important task of the reconstruction is this decision. Thus the metric measure has the important task of making this decision through the analysis of the information available. Although our proposed reconstruction technique delays this decision making till the segmentation stage, since segmentation is accomplished through minimization of an energy function, it is still extremely important for the metric volume to derive a meaning measure of cost so that the 3D true scene surface is the global minima.

Shape from Voxel Occupancy and shape from photo-consistency techniques are a class of algorithm that uses the output of the metric volume as the reconstruction of the 3D scene. From the silhouette contours of each image, Voxel Occupancy algorithms uses a geometric consistency measure to determine whether a voxel is opaque or transparent. Voxel Coloring [51] uses the likelihood ratio test, $\lambda_V = \frac{(n-1)s^2}{\sigma_o^2}$ distributed as χ^2, to determine the color consistency between the projections, and a threshold corresponding to the maximum allowable correlation error is used to decide whether a voxel is transparent. Visibility and occlusions are also modelled in the metric volume implicitly. The ordinal visibility constraint [51] and Collins' plane sweep algorithm [9] are examples of ways in which the ordering of the voxel traversal in the analysis of the projections is used to determine the occupancy of the voxel. These methods are described in detail in Chapter 2.

The method chosen to analyse the metric volume and the volumetric reconstruction employed will also determine the most effective and efficient volumetric representation to use. The most common approach is to represent the volume as a regular tessellation of cubes, a set of voxels, in Euclidean 3-space. Other representations however are also available. Octrees provide a more space-efficient description of the scene and are most common in shape from silhouettes algorithms [8, 47, 59]. It is most useful when the volume contains large areas of transparent regions. Disparity space representations, although this 2.5D representation is mostly used by stereo matching algorithms, techniques which determines depth relative to one camera frame tends to use disparities to de-

scribe the reconstructed scene [49, 25, 61]. Without complete calibrations and using only fundamental matrices, projective space representations can also be used to describe the voxel space [23]. Other representations include ray space representation [28, 36] and polygonal surface representation [33].

Metric measures such as variance or a differnce measure or even a probability density function (pdf) can be used to derive the metric volume. In our formulation of the problem, our aim is to design a metric volume suitable for surface evolution where the aim is to minimize the sum of all metric along the evolved surface. Thus a photo-consistent voxel which is to be identified as opaque and lying on the surface, must return minimum measure, preferably zero. An inconsistent voxel which is to be labelled transparent and carved away should return a high metric value. Since the limit of the minimization is a sum of zero, negative measures are undefined.

In our initial implementation, we have chosen to use a variance measure and thresholding the variance to determine whether a voxel is transparent. In our initial stage, occlusion has not been considered in the construction of the metric volume and has been left for the task of the segmentation stage to iteratively model the occluded regions. The variance is computed for all projected rays onto each image. For N images ranging from $i = 1 \dots N$ and M voxels ranging from $m = 1 \dots M$, each voxel contains i projections. A feature vector Q for each voxel can be defined which contains intensity values from each image i. The mth voxel contains a corresponding feature vector Q^m. The variance of the feature vector can therefore be defined

$$\begin{aligned} VAR(Q^m) &= \mathbb{E}((Q^m - \mathbb{E}(Q^m))^2) \\ &= \mathbb{E}((Q^m)^2 - 2Q^m\mathbb{E}(Q^m) + (\mathbb{E}(Q^m))^2) \\ &= \mathbb{E}((Q^m)^2) - 2\mathbb{E}(Q^m)\mathbb{E}(\mathbb{E}(Q^m)) + \mathbb{E}((\mathbb{E}(Q^m))^2) \\ &= \mathbb{E}((Q^m)^2) - (\mathbb{E}(Q^m))^2 \\ &= \frac{\sum_i (Q^m)^2}{N} - \frac{(\sum_i Q^m)^2}{N^2} \end{aligned}$$

3.2.1 Progress to Date

A metric volume, based on thresholding the variance of the feature vector computed for each voxel, has been constructed for the two image sequences (Figure 3.8. Adjusting the threshold level alters the error measure allowing more mismatches in the volume but providing more data for the segmentation stage to follow. Notice also that in order to register depth information and accurate matches for the full scene, a larger volume needs to be constructed. An alternative which will be investigated is by beginning with the furthermost plane and zoom out rather than being with the closest plane and zooming in. In order to produce a useful projection volume, accurate P-Matrices are also important, so that matches occur within the volume of interest.

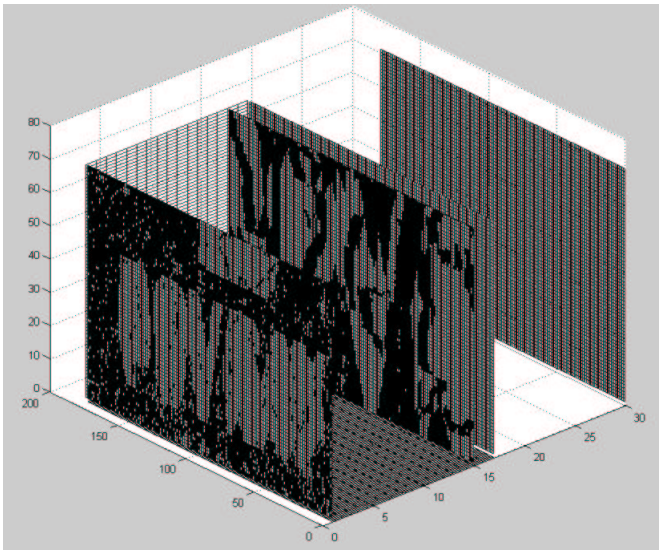


Figure 3.7: Metric volume computed for the Povray sequence

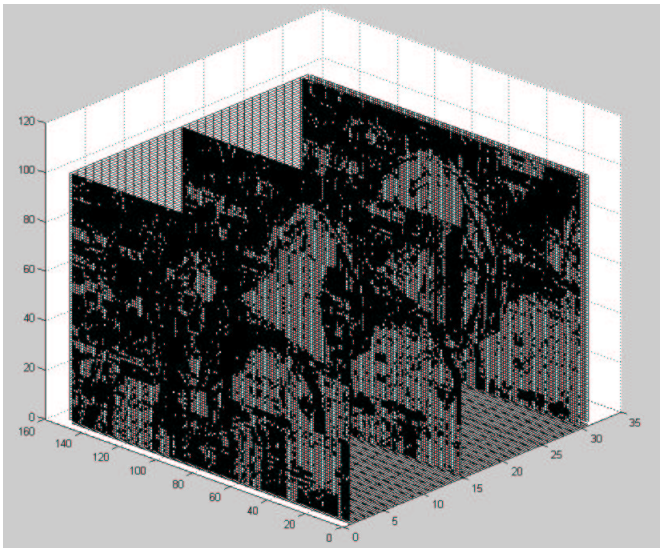


Figure 3.8: Metric volume computed for the 'Head and Lamp' sequence. Notice how the lamp has matched well in the foreground, while the head matches well in the back planes of the projection

3.3 Surface Evolution

Classical active contours such as snakes [22] and level sets [1] have mainly been applied to the segmentation problem in image processing. The recent introduction of fast implicit active contour models [27], which uses the semi-implicit additive operator splitting (AOS) scheme introduced by Weickert et al. [65], is an improved version of the geodesic active contour framework [7]. Given such advancements in active contour analysis, multi-dimensional segmentation are becoming not only more robust and accurate, but computationally feasible. The application of surface evolution and level set methods to the stereo problem was pioneered by Faugeras and Keriven [15]. Although Faugeras' approach is limited to binocular stereo and the epipolar geometry, their novel geometric approach to the stereo problem has laid the foundation for a new set of algorithms that can be used to solve the 3D reconstruction problem. Our proposal is to approach the 3D scene reconstruction problem by applying surface evolution to volumetric scene modelling.

3.3.1 Energy Minimization Overview

One of the common methods for solving a segmentation problem is to by defining the problem as an energy minimization problem. If the problem is correctly modelled, energy minimization can theoretically provide the optimal solution. By redefining the reconstruction of a 3D scene as a segmentation problem, energy minimization techniques can be applied to recover the 3D structure from a set of images.

The energy minimization problem can be viewed as two separate steps, the definition of an energy function and the minimization of the function. In order for energy minimization to occur, a cost function where its minimum represents the optimal solution needs to be constructed. Terms to consider in the energy function includes parameters such as smoothness or regularization, data fidelity, prior knowledge, visibility or geometric information.

Once the energy function has been derived, a variety of methods can be applied in order to solve the problem. Traditionally, simulated annealing has been used [2, 18]. Computational flow and fluid dynamic analysis have also been applied to solve the energy function. The bottleneck restricting the maximum water flow have been demonstrated to produce the global minima. A third class of method involves the discretization of the energy function into a graph. Graph embedding transform the segmentation problem in a maximum flow minimum cut problem, such that geodesic shortest path algorithms can be applied in order to determine the segmentation. A fourth class of method involves the use of variational calculus to derive a partial differential equation (PDE) which can be used to evolving a surface towards the minimum energy.

The simulation of a PDE can be accomplished in a number of ways. Cellular Automata has been demonstrated to be able to simulate the evolution of a PDE [46]. Snakes can be applied in order to evolve a curve governed by a PDE towards a minimum energy [22]. Volume of fluids has also been introduced to describe

the evolution of a curve [43]. More recently, level sets have also introduced capable of evolving a surface described by a PDE, by representing the surface as the zero level set of a function [1].

Although energy minimization has been widely applied to segmentation problems, it has only been recently that energy minimization has been used to reconstruct 3D scenes. A variety of energy functions have been proposed and graph cuts have been a popular method applied to solve for the 3D scene [25, 49]. 3D reconstruction through surface evolution have also been explored [14]. Many of these methods however have approach the 3D reconstruction problem by redefining the stereo correspondence problem into an energy minimization problem. Our proposed method is to combine the advantages of both volumetric scene modelling techniques and energy minimization in order to reconstruct the 3D scene space.

3.3.2 3D Reconstruction via Segmentation

The development of algorithms that can provide globally optimal solutions to segmentation problems has made its application to image processing problems very attractive. By designing a volume appropriate for maximum-flow analysis, the minimum-cut associated with the maximum flow can be viewed as an optimal segmentation. While Roy and Cox have demonstrated a version of maximum-flow to analyse stereo images, a more computationally feasible method was recently proposed by Sun [58]. A two-stage dynamic programming (TSDP) technique was introduced to obtain efficiently a 3D maximum-surface, which enables the computation of a dense disparity map.

In our voxel volume formulation, since our projected volume enables us to work directly in true 3D coordinates, we aim to output a 3D surface representative of the complete 3D scene rather than using a disparity map to obtain a 2.5-D sketch of the scene [34]. Formulating the 3D reconstruction problem as a segmentation problem has many advantages over the use of the classical dynamic programming technique. In segmentation, optimisation is performed along a surface rather than along a line. This subsequently provides segmentation methods with the advantage of outputting contours that wrap back on themselves, while dynamic programming will have difficulty following these concave surfaces. Rather than reformulating dynamic programming or similar techniques in order to model occlusions and concavity, we propose the use of segmentation to approach 3D reconstruction from a new point of view.

Active contours have been demonstrated to be a useful tool in the segmentation problem. Geodesic active contours that use a variational framework have been shown to obtain locally minimal contours [7]. Fast implicit active contour models, that use the semi-implicit additive operator splitting (AOS) scheme introduced by Weickert et al. [27, 65], and shortest path algorithms [6], have been used to avoid the variational framework producing optimal active contours. By formulating the metric volume such that the 3D segmentation is the 3D scene surface, we can approach volumetric scene reconstruction through the evolution of surfaces. By choosing a positive scalar metric, g, such that g can be assured

to be always greater than zero, the minimisation of the energy functional E, can be formulated to describe the 3D scene structure

$$E(C) = \int_C g(C(s))ds$$

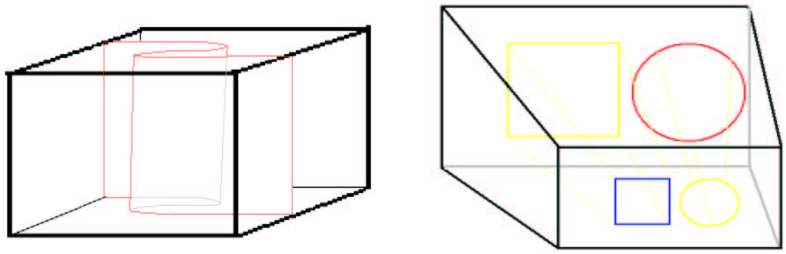
where C is the segmentation contour.

Level set theory can be applied to evolve the surface contour towards the minimum energy. Beginning with an initial surface that encloses the scene to be reconstructed, the surface can at successive time steps move inwardly along the surface normal towards the 3D objects in the scene. A PDE derived from an appropriate energy function can be used to determine the speed of the evolution such that the speed decreases gradually as the surface approaches the true 3D object. Kolmogorov's approach involves an energy function that includes a data term that is the SSD (Sum of Squared Difference) of the pair of images, a smoothness term that favours similar depths for adjacent pixels, and a visibility term which determines whether a particular pixel has been occluded. Their method however is limited to the comparison of pairs of images and they have chosen to use graph cuts to solve the minimization. Although Faugeras uses the Level Set method, his energy function also involves pairwise matching and a cross correlation term has been used to determine the speed function of the evolution. Our formulation aims to design an energy function that would operate on the metric volume. Similar to Space Carving, we propose to approach the reconstruction problem from a volumetric modelling point of view. Contrary to Space Carving, we propose to improve on volumetric scene modelling techniques by exploiting spatial coherence through the use of segmentation and curve evolution.

3.3.3 Occlusion Modelling

Algorithms that depend on dense feature correspondence have much difficulty modelling occlusions. Our proposed method overcomes this problem by redefining the problem, using a new approach that does not depend on pixel to pixel feature correspondence. One of the advantages of our approach is that occlusion does not need to be explicitly modelled. Occluded regions visible in a limited number of images are still projected validly into 3D space for analysis, with the major difference being that less images project to that region. This scheme subsequently also allows for occluded region to be iteratively improved as more images of the occluded scene are available.

Similar to the use of item buffers in Weghorst's variation of Voxel Coloring [12], Kolmogorov included in the energy function an occlusion term that analyzes the visibility of a 3D projection through comparisons of the 3D point with depths of other 3D points. That is if a 3D point projects within a certain range of another 3D point at a different depth, then occlusion must occur. The occlusion analysis of each pair of possible 3D points is termed interaction. Faugeras approach allows a level set to evolve such that the evolution is driven by a cost functional measure that is evaluated by projecting each current point



(a) (b)

Figure 3.9: (a) Dynamic program outputs a 2.5D sketch, thus will not allow the surface to fold back onto itself. (b) The reconstruction of the rectangle (blue in the front plane) might seem occluding the circle (red in the back plane) if the volume is analyzed as a Euclidean volume. Following the ray-tracing (yellow lines), a projective nature shows that the rectangle does not occlude the circle.

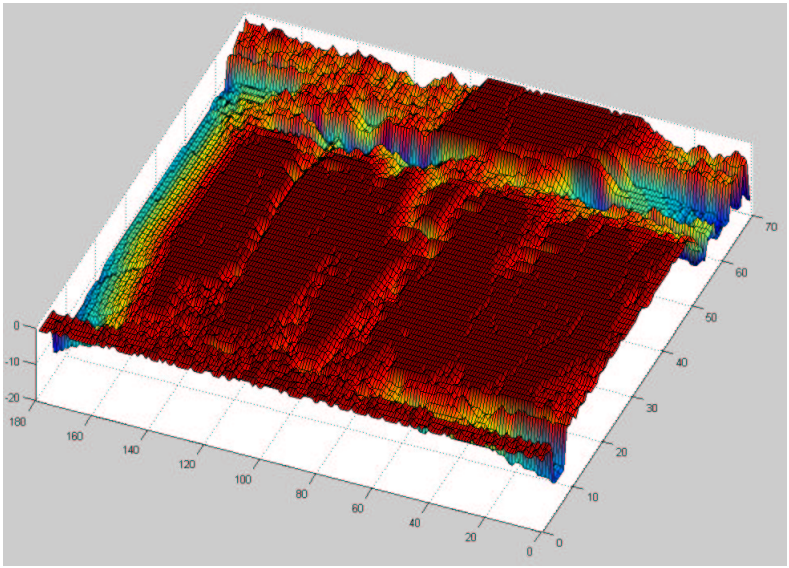
on the surface, S. Since each correlation term in the integral is defined only for points on S that are visible from both pair of images under consideration, those views which do not see the 3D point thus theoretically does not contribute to the decision making of the next evolution of the curve. Our proposed approach differs from Faugeras since we do not back-project each 3D point at each time step and we do not perform pair-wise comparisons.

Since the surface evolution occurs in the metric volume space, modelling of occlusion depends on the metric measure. By considering data fidelity, whether an occluded regions can be reconstructed would depend on the number of views that can see the occluded 3D point. Since the creation of the metric volume is not only a decision on the likelihood and the probability of the visibility of a particular voxel, it is also a vote of confidence on the most likely radiance of a particular voxel. An occluded region with many views viewing it from another angle will drive the segmentation to correctly model the occluded region. While an occluded region without enough views supporting the decision making will have the segmentation smooth out or wrap over the occluded region. However, after an initial segmentation has been computed, it is possible to back-project each point on the surface to the available images to reevaluate the photo-consistency of the reconstruction. It is also possible to determine which image in fact cannot view the occluded region, and with two or more views of the occluded region, reconstruction in that region can be reprocessed.

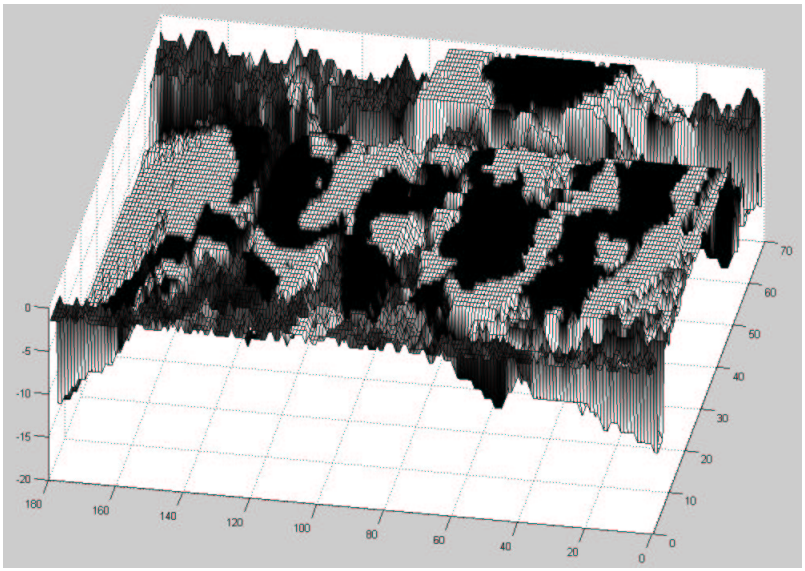
3.3.4 Progress to Date

Given the metric volume, a dynamic programming (DP) algorithm has been applied to the metric volume in the attempt to segment the volume without surface evolution computations (Figure 3.10 and Figure 3.11). It has been ob-

served however that a DP analysis on the metric volume is meaningless. This is because the application of DP implies a disparity solution and since a disparity solution is a 2.5D sketch, it does not allow for a multi-valued solution, that is the curve wrapping backwards onto itself, as illustrated in figure 3.9a. The DP algorithm operates under the assumption that the volume is Euclidean. But since the volume constructed is a projective volume, a pinhole camera model, objects which are not occluding each other under projective analysis might seem occluded if the volume is treated as a Euclidean volume (which is why the DP output is erroneous). As depicted figure 3.9b, if the volume is not interpreted as a projective volume, it might seem that the rectangle (which is in the front plane) is occluding the circle (which is in the back plane). Since the rectangle is directly in front of the circle. However, if the projective nature of the volume is taken into account, it can be seen that the rectangle does not occlude the circle. In order to overcome this problem, surface evolution techniques can be applied to recover the object surface. Unlike snakes, level sets have the advantage that surfaces can split since it is the solution to the zero level set of an implicit volume.

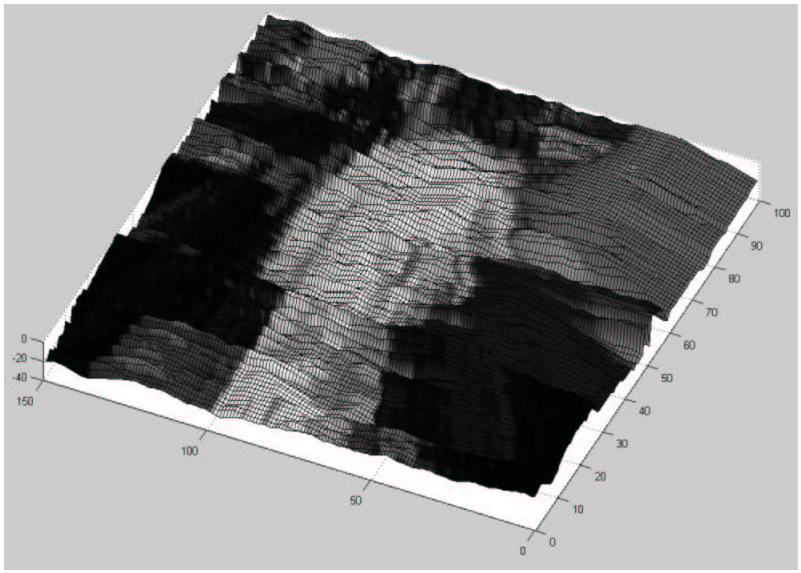


(a)

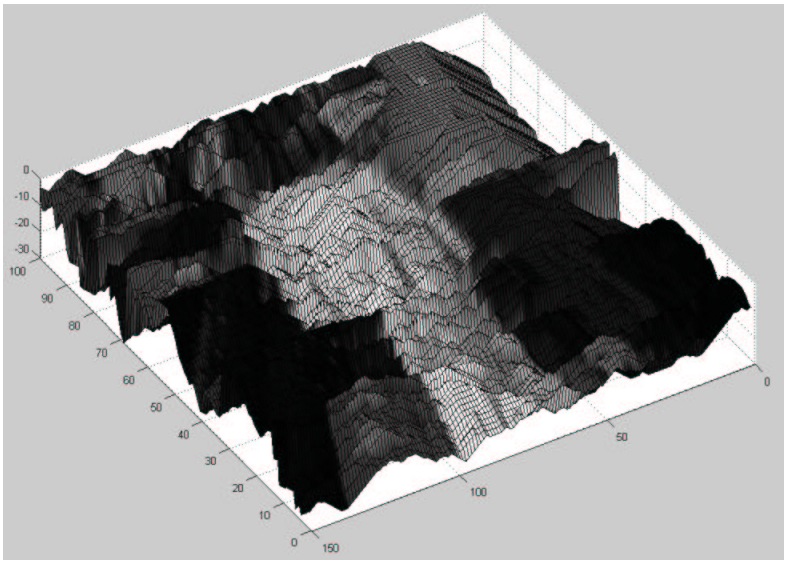


(b)

Figure 3.10: 3D reconstruction of a section of the Povray sequence. (a) untextured. (b) textured.



(a)



(b)

Figure 3.11: 3D reconstruction of a section of the 'Head and Lamp' sequence from different viewing angles. Notice the lamp is reconstructed to be in front of the head model.

Chapter 4

Future Work

While much research have been conducted in 3D reconstruction and reconstructions are becoming increasingly photorealistic, improvements are still needed in order to accurately and efficiently recover the 3D scene from images. Geometric accuracy, realistic surface reflectance and voxel visibility models, methods to account for complex large-scale dynamic environments, and real-time reconstruction remain areas of research and development in the field of 3D scene reconstruction. My research aims at exploring new techniques that can accurately reconstruct a dynamic 3D scene from multi-view image sequences.

4.1 3D

Having designed a framework to model an arbitrary 3D scene from images, the next task will be to implement the idea. Initial design of the projection volume have been accomplished and the next step will be to develop expertise in the segmentation stage and surface evolution. Level sets methods will be explored and its application to volumetric scene reconstruction tested. Apart from the Povray and 'Head and Lamp' sequence, I also plan to perform reconstruction on the famous dinosaur sequence. This will allow comparisons with existing algorithms.

I also plan to explore P-Matrix computations. It is always difficult to assume that the image sequences obtained are provided with the P-matrices. Even with calibration information provided, processing is still required to compute the required projection matrices. Techniques developed to accurately recover P-matrices from uncalibrated image sequences will be extremely useful.

Knowledge of the camera geometry not only enables the construction of the projection matrix, but also allows the computation of the polyhedral intersection of the camera views. Thus rather than assuming a large initial volume that would hopefully encapsulate the 3D scene to be reconstructed, by intersection each camera's projection pyramid, a volume of interest can theoretically be computed. The computation of this polyhedral however is an extremely challenging

problem when the number of views is large.

4.2 4D

The fusion of stereo and motion has been recognised by many researchers as a means of providing additional information that were not previously obtainable through their independent analysis. Waxman and Duncan pioneered the analysis of stereo motion by considering binocular image flow [63]. Many studies have subsequently tackled this problem through the use of Kalman filtering, optical flow and feature tracking [24, 40]. While these methods have demonstrated reasonable success, they are limited by problems inherent in the correspondence problem. Similar to our proposed approach in the use of segmentation, rather than reformulating optical flow or Kalman filtering to model stereo motion, we propose the use of a 4D voxel volume in order to analyse the stereo dynamics in stereoscopic image sequences.

By embedding the design of our 3D voxel feature volume into a 4-D feature space, we present a novel approach to the analysis of stereo motion for the reconstruction of dynamic 3D scenes. Given a set of images captured over different time frames, we can compute the camera geometry and projection parameters for each image through the use of the many calibration techniques developed, such as Zhang's four point algorithm for stereo rig analysis [67]. From the set of projection matrices computed, we can construct a voxel volume for each time frame. Since geodesic active contours can be applied to segment multi-dimensional volumes, similar to the analysis of our 3D voxel volume, we can compute a segmentation in 4D in order to produce a 3D surface in time. The use of this 4D voxel volume also has the advantage of not only recovering the 3D and motion information from stereoscopic image sequences, but is capable of processing multi-view image sequences. The computational feasibility of multi-dimensional segmentation makes this 4D approach to stereo motion an attractive alternative to the analysis of dynamic, complex 3D scenes.

I have a strong and keen interest in exploring 3D reconstruction of dynamic scenes. Rather than solving the problem of multi-view stereo (of views greater than two), I have a greater interest in stereo motion since it is like a natural pair of eyes. Stereo motion analysis can potentially become extremely useful for robotics.

4.3 Research Plan - Timeline

A research plan for the rest of the PhD studies is presented.

March - June 2003	Learn to evolve surfaces using Level Sets Method Design an Energy Function and derive suitable PDE such that segmentation outputs 3D scene Scene Reconstruction from available multi-view images
July - September 2003	Obtain arbitrary multi-view images Manually obtain face and arbitrary scene images Computation of P-Matrices for multi-view images
October - December 2003	Scene reconstruction for arbitrary multi-view images Occlusion modelling
January - June 2004	4D scene reconstruction Stereo motion analysis from multi-view image sequences

Bibliography

[1] D. Adalsteinsson and J. A. Sethian. A fast level set method for propagating interfaces. *Journal of Computational Physics*, 118(2):269–277, 1995.

[2] Stephen Barnard. Stochastic stereo matching over scale. *International Journal of Computer Vision*, 3(1):17–32, 1989.

[3] J. S. De Bonet and P. Viola. Poxels: Probabilistic voxelized volume reconstruction. In *Proceedings of ICCV*, pages 418–425, 1999.

[4] Yuri Boykov, Olga Veksler, and Ramin Zabih. Fast approximate energy minimization via graph cuts. In *ICCV (1)*, pages 377–384, 1999.

[5] Alexander M. Bronstein, Michael M. Bronstein, and Ron Kimmel. 3d face - fast and accurate face recognition. http://www.cs.technion.ac.il/ gip/faces/faces.htm, September 2002.

[6] M. Buckley and J. Yang. Regularised shortest-path extraction. *Pattern Recognition Letters*, 18(7):621–629, 1997.

[7] Vincent Caselles, Ron Kimmel, and Guillermo Sapiro. Geodesic active contours. *International Journal of Computer Vision*, 22(1):61–79, 1997.

[8] C.H. Chien and J.K. Aggarwal. Volume/surface octrees for the representation of three-dimensional objects. *CVGIP*, 36:100–113, 1986.

[9] R. T. Collins. A space-sweep approach to true multi-image matching. In *CVPR*, pages 358–363, 1996.

[10] I. Cox, S. Hingorani, S. Rao, and B. Maggs. A maximum likelihood stereo algorithm. *Computer Vision and Image Understanding*, 63(3):542–567, 1996.

[11] G. Cross. *Surface Reconstruction from Image Sequences: Texture and Apparent Contour Constraints*. PhD thesis, University of Oxford, 2000.

[12] Bruce Culbertson, Thomas Malzbender, and Greg Slabaugh. Generalized voxel coloring. In B. Triggs, A. Zisserman, and R. Szeliski, editors, *Vision Algorithms: Theory and Practice*, number 1883 in LNCS, pages 100–115, Corfu, Greece, Sept 1999. Springer-Verlag.

[13] Charles R. Dyer. *Volumetric Scene Reconstruction From Multiple Views*, chapter 16, pages 469–489. in Foundations of Image Understanding. Kluwer, Boston, 2001.

[14] Olivier Faugeras and Renaud Keriven. Complete dense stereovision using level set methods. *Lecture Notes in Computer Science*, 1406:379–393, 1998.

[15] Olivier Faugeras and Renaud Keriven. Variational principles, surface evolution, pde's, level set methods and the stereo problem. *Special Issue on Partial Differential Equations and Geometry-Driven Diffusion in Image Processing and Analysis of the IEEE Transactions on Image Processing*, 7(3):336–344, March 1998.

[16] P. Fua. From multiple stereo views to multiple 3d surfaces. *International Journal of Computer Vision*, 24(1):19–35, 1997.

[17] D. Geiger, B. Ladendorf, and A. Yuille. Occlusions and binocular stereo. *International Journal of Computer Vision*, 14:211–226, 1995.

[18] S. Geman and D. Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6:721–741, 1984.

[19] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 1998.

[20] H. Ishikawa and D. Geiger. Occlusions, discontinuities, and epipolar lines in stereo. In *ECCV*, pages 232–248, Freiburg, Germany, June 1998.

[21] S. Kang, R. Szeliski, and J. Chai. Handling occlusions in dense multiview stereo.

[22] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, 1(4):321–331, 1998.

[23] M. Kimura, H. Saito, and T. Kanade. 3d voxel construction based on epipolar geometry. In *ICIP*, volume 3, pages 135–139, October 1999.

[24] R. Koch. 3-d surface reconstruction from stereoscopic image sequences. In *ICCV*, pages 109–114, Cambridge, MA., USA, June 1995.

[25] Vladimir Kolmogorov and Ramin Zabih. Multi-camera scene reconstruction via graph cuts. In *ECCV*, volume 3, pages 82–96, 2002.

[26] Vladimir Kolmogorov and Ramin Zabih. What energy functions can be minimized via graph cuts? In *ECCV (3)*, pages 65–81, 2002.

[27] G. Kühne, J. Weickert, M. Beier, and W. Effelsberg. Fast implicit active contour models. In L. Van Gool, editor, *Pattern Recognition*, Lecture Notes in Computer Science. Springer, Berlin, 2002. To appear.

[28] K. N. Kutulakos. Shape from the light field boundary. In *CVPR*, pages 53–59, 1997.

[29] Kiriakos N. Kutulakos. Approximate n-view stereo. In D. Vernon, editor, *ECCV*, volume 1842 of *Lecture Notes in Computer Science*, pages 67–83. Springer-Verlag, 2000.

[30] Kiriakos N. Kutulakos and Steven M. Seitz. A theory of shape by space carving. Technical Report TR692, Computer Science Dept., U. Rochester, 1998.

[31] A. Lanitis, C.J. Taylor, and T.F. Cootes. Automatic face identification system using flexible appearance models. *IVC*, 13(5):393–401, June 1995.

[32] A. Laurentini. The visual hull concept for silhouette-based image understanding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(2):150–162, February 1994.

[33] W. E. Lorensen and H. E. Cline. Marching cubes: A high resolution 3d surface construction algorithm. In *SIGGRAPH*, pages 163–169, 1987.

[34] D. Marr. *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information.* W.H. Freeman and Co., 1982.

[35] W.N. Martin and J.K. Aggarwal. Volumetric descriptions of objects from multiple views. *Trans. on Pattern Analysis and Machine Intelligence*, 5(2):150–158, Mar 1983.

[36] Wojciech Matusik, Christopher Buehler, Ramesh Raskar, Leonard McMillan, and Steven J. Gortler. Image-based visual hulls. In *SIGGRAPH*, pages 369–374, 2000.

[37] Gerard Medioni. Binocular and multiple-view stereo using tensor voting. Technical report, USC IMSC, 2001.

[38] Saied Moezzi, Arun Katkere, Don Y. Kuramura, and Ramesh Jain. Reality modeling and visualization from multiple video sequences. *IEEE Computer Graphics and Applications*, 16(6):58–63, Nov. 1996.

[39] Saied Moezzi, Li-Cheng Tai, and P. Gerard. Virtual view generation for 3d digital video. *IEEE Multimedia*, 4(1):18–26, Jan. - Mar. 1997.

[40] T. Moyung and P. Fieguth. Incremental shape reconstruction using stereo image sequences. In *ICIP*, 2000.

[41] Reuters CNN News. Twins crack face recognition puzzle. http://www.cnn.com/2003/TECH/ptech/03/10/israel.twins.reut/index.html, 10 March 2003.

[42] W. Niem. Robust and fast modelling of 3d natural objects from multiple views. In *SPIE Proceedings*.

[43] W. Noh and P. Woodward. A simple line interface calculation. In A.I. van de Vooran and P.J. Zandberger, editors, *Fifth International Conference on Fluid Dynamics*. Springer-Verlag, 1976.

[44] M. Okutomi and T. Kanade. A multiple-baseline stereo. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 15(4):353–363, 1993.

[45] Stanley Osher and James A Sethian. Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations. *Journal of Computational Physics*, 79:12–49, 1988.

[46] N.H. Packard and S. Wolfram. Two-dimensional cellular automata. *Journal of Statistical Physics*, 38:901–946, Mar 1985.

[47] M. Potmesil. Generating octree models of 3d objects from their silhouettes in a sequence of images. *Computer Vision Graphics Image Processing*, 40:1–29, 1987.

[48] S. Roy. Stereo without epipolar lines: A maximum flow formulation. *International Journal of Computer Vision*, 1(2):1–15, 1999.

[49] S. Roy and I. J. Cox. A maximum-flow formulation of the n-camera correspondence problem. In *ICCV*, pages 492–499, 1998.

[50] Daniel Scharstein and Richard Szeliski. A taxonomy and evaluation of dence two-frame stereo correspondence algorithms. *International Journal of Computer Vision*, 47(1/2/3):7–42, April-June 2002.

[51] S. M. Seitz and C. R. Dyer. Photorealistic scene reconstruction by voxel coloring. In *CVPR*, pages 1067–1073, 1997.

[52] Jonathan W. Shade, Steven J. Gortler, Li-Wei He, and Richard Szeliski. Layered depth images. *Computer Graphics*, 32(Annual Conference Series):231–242, 1998.

[53] M. Shneier, E. Kent, and P. Mansbach. Representing workspace and model knowledge for a robot with mobile sensors. In *ICPR*, pages 199–202, Montreal, Canada, July 1984.

[54] G. Slabaugh, W. B. Culbertson, T. Malzbender, and R. Schafer. A survey of volumetric scene reconstruction methods from photographs. In K. Mueller and A. Kaufman, editors, *Volume Graphics 2001, Proc. of Joint IEEE TCVG and Eurographics Workshop*, pages 81–100, Stony Brook, New York, USA, June 2001. Springer Computer Science.

[55] D. Snow, P. Viola, and R. Zabih. Exact voxel occupancy with graph cuts. In *CVPR*, pages 345–352, 2000.

[56] P. Srinivasan, Ping Liang, and S. Hackwood. Computational geometric methods in volumetric intersection for 3d reconstruction. *Pattern Recognition*, 23(8):843–857, 1990.

[57] S. K. Srivastava and N. Ahuja. Octree generation from object silhouettes in perspective views. *Computer Vision, Graphics and Image Processing*, 49:68–84, 1990.

[58] Changming Sun. Fast stereo matching using rectangular subregioning and 3d maximum-surface techniques. *International Journal of Computer Vision*, 47(1/2/3):99–117, May 2002.

[59] R. Szeliski. Rapid octree construction from image sequences. *CVGIP: Image Understanding*, 58(1):23–32, July 1993.

[60] R. Szeliski and R. Zabih. An experimental comparison of stereo algorithms. In *International Workshop on Vision Algorithms*, pages 1–19, Kerkyra, Greece, September 1999. Springer.

[61] Richard Szeliski and Polina Golland. Stereo matching with transparency and matting. In *ICCV*, pages 517–526, 1998.

[62] Sundar Vedula, Simon Baker, Steven Seitz, and Takeo Kanade. Shape and motion carving in 6d. In *Computer Vision and Pattern Recognition*, volume 2, pages 592–598, June 2000.

[63] Allen M. Waxman and James H. Duncan. Binocular image flows: Steps toward stereo-motion fusion. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 8(6):715–729, Nov. 1986.

[64] Hank Weghorst, Gary Hooper, and Donald P. Greenberg. Improved computational methods for ray tracing. *ACM Transactions on Graphics*, 3(1):52–69, Jan 1984.

[65] J. Weickert, B. ter Haar Romeny, and M. A. Viergever. Efficient and reliable schemes for nonlinear diffusion filtering. *IEEE Trans. Image Proc.*, 7(3):398–410, 1998.

[66] A. L. Yuille and T. Poggio. A generalized ordering constraint for stereo correspondence. AI Memo 777, MIT, AI Lab, 1984.

[67] Z. Zhang. Motion and structure of four points from one motion of a stereo rig with unknown extrinsic parameters. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 17(12):1222–1227, December 1995.

[68] Z. Zhang. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1330–1334, 2000.

[69] Z. Zhang, R. Deriche, L. T. Luong, and O. Faugeras. A robust approach to image matching: Recovery of the epipolar geometry. In *ECCV*, pages 179–186, 1994.