Bayesian Methods in Graphics

P.H.S. Torr*
with
A. Blake, R. Cipolla, A Criminisi, A. Dick, C. Rother, B. Stenger, J. Shotton, A.Thayananthan
Microsoft Research, 7 JJ Thomson Avenue,
Cambridge, CB3 0FB,UK

Abstract

Within this talk I shall describe some of the work I have been involved with at Microsoft Research Cambridge on Bayesian methods. In particular I will cover the application of Bayesian methods to certain problems relating to the field of Computer Graphics. Bayesian methods provide a rational way of making inference about problems; including learning parameters that are so often set by hand, and the incorporation of prior knowledge. The particular problems I shall address are (a) image cut out, (b) new view synthesis, (c) motion capture of articulated objects (e.g. hands) for driving avatars, (d) general 3D reconstruction.

In all of these problems I hope to demonstrate that the Bayesian approach leads to new insights and methodologies that improve on existing methods.

1 Introduction

The approach that follows is unashamedly Bayesian, as set out in his book [Jaynes 2003] the Bayesian method provides a consistent way of reasoning about the world that can be viewed as an extension of the Aristotelian calculus of logic to uncertainty.

The Bayesian approach was probably first proposed for vision (at least for segmentation) by Besag [Besag 1974] and then elaborated by Geman and Geman [Geman and Geman 1984]. Joe Mundy, now professor at Brown, being skeptical of Bayesian methods as "post hoc dressing up" [1] once asked what use are Bayesian methods in graphics? This paper is a small attempt to answer that question. Some will complain that to use Bayesian methods one must introduce arbitrary priors on the parameters. However, far from being a disadvantage, this a tremendous advantage as it forces open acknowledgment of what assumptions were used in designing the algorithm, which all too often are hidden away beneath the veneer of equations. Furthermore Bayesian methods provide a consistent way to learn the parameters of our models. Machine learning and Bayesian theory are closely intertwined To quote from his abstract of the talk of David Salesin to NIPS 2003 this year: "Machine learning has the potential to revolutionize the field of computer graphics. Until now, so many of the successes in computer graphics from realistic plant models to human animation to cinematographic effects have been achieved, painstakingly, through the creation of highly complex models by hand. Unfortunately, the process for creating these models does not scale: Whether for plants, animation, or cinematography, good models are hard to come by, with each model having to be crafted, individually, by an expert. Yet good examples of all of these things are all around us. Thus, for computer graphics to achieve its full potential, what we really need is for all of these highly complex models to be constructed automatically from the examples themselves in other words, what we really need is machine learning! ".

Within this talk I shall outline four areas in which Bayesian methods have proven useful to graphics applications. The methods have all been published elsewhere so this paper will simply give an outline of what each method does together with results and citation to the appropriate material.

The first application is image cut out, explained in Section 2, this can be done optimally be a Markov Random Field. Markov random fields are a powerful statistical model of spatial relationships within an image, and under certain assumptions, can be solved exactly. However the segmentation is very sensitive to choice of parameters. Bayesian learning of the parameters of the Markov Random field allow for a state of the art segmentation with minimal user inter actions.

Next in Section 3 the use of a weak prior model is shown to improve new view synthesis, again the parameters of the prior can be learned off line; reducing the reliance on human intervention.

Section 4 describes a Bayesian method for tracking, by maintaining a whole distribution of possibilities Bayesian filtering offers a way to overcome the ambiguity inherent in many tracking situations.

Section 5 reviews using high level priors for 3D reconstruction allowing a natural way to encode our knowledge about the shape of classes of objects. Bayesians methods provide a natural way to combine prior information with data to yield optimal estimates. Bayesian optimization methods such as MCMC furnish ways to test that our prior distribution is valid.

2 Bayesian Image Cut Out

Within this section we address the problem of interactive segmentation of images, e.g. to allow the user to cut out part of an image (such as a face) and paste it into another. This has found a variety of uses in medical imaging, movie special effects and the application we consider here: home editing of digital photographs. The wide variety of content within images and the differing desires of users precludes a fully autonomous system, so the goal is to make the interactive segmentation process as painless as possible by minimizing the amount of time (mouse clicks) required to achieve the desired segmentation. Many existing interactive techniques for segmentation require significant user input.

Previous work in this theme has included the Magic Wand of Adobe [Ado 1999], which computes a region of connected pixels similar to a pixel chosen by a mouse click, however it cannot work well in textured regions. Another class of approaches attempts to segment the image by finding a boundary that coincides with strong edges in the image, ignoring all other pixel information. These include (a) LIVEWIRE [Mortensen and Barrett 1998] in which the user clicks on two points on the boundary and dynamic programming is used to find the minimum cost path between these two points, on a graph formed on the image pixels with the arcs having weight inversely proportional to the edge strength be-

* e-mail: philtorr@microsoft.com New Address Oxford Brookes University, Dept Computing, Oxford,OX33 1HX

[1] P. Anandan, Microsoft Research has often claimed that many algorithms have a Bayesian interpretation added as a "post hoc dressing up"

tween pixels. (b) JETSTREAM [Perez et al. 2001] following a more probabilistic approach requires the user to click repeatedly on the boundary of the object and from this generates putative boundaries, along high contrast edges, around the object using a particle filter. (c) SNAKES [Kass et al. 1987] define an active contour around the object to be segmented and then iteratively minimize the contour's energy functional to achieve the optimal boundary, combining external forces such as gradient energy with internal forces like curvature to achieve an ideal segmentation. There are systems that then allow the user to interactively modify the energy landscape and thus nudge the snake [Daneels et al. 1993].

All of these method require many mouse clicks as in most typical images there are so many edges present that the cost surface is abundant with local minima. In particular they are often confused by T-junctions or other such edges in the background. In order to make the algorithm more effective a richer set of observations must be used. Where might these observations come from? Recently a new class of INTERACTIVE GRAPH CUT (ICG) methods have been proposed [Boykov and Jolly 2001] that consider the distribution of the colour of the foreground and background pixels in addition to the boundary pixels. Furthermore a key advantage of (ICG) is that it can also apply to $N-D$ images (volumes) opening it up to be applied for segmenting many types of medical volumetric images such as provided by computed tomography (CT), or video. It transpires that the combination of region and boundary information to assist interactive segmentation leads to far less effort for the user. The ICG allows for both terms to combined to form a posterior likelihood by using a Markov Random Field (MRF) formulation. In [**?**] we demonstrate a clear improvement to the (ICG) which has two parts, which we dub EMCUT. The first is that the initialization is in the form of a rectangle whose only constraint is that it bounds the object of interest (this is the 'one click' initialization). The second is the application of the EM algorithm to learn a mixture model for the colour of the foreground and background object. This involves exactly specifying the probabilistic formulation which was left somewhat hazy in [Boykov and Jolly 2001]. The algorithm proceeds by conditional maximization holding sets of the parameters constant whilst maximizing the posterior over the remainder. Bayesian methods are also used to automatically learn the parameters of the model as described in [Blake et al. 2004].

2.1 Image Cut Out Results

Our approach is demonstrated on the 'starfish' and 'donkey' images shown in Figure 1 column (a). In each case the user drags and clicks a rectangle, shown in column (b), over the object of interest (a one click operation). As can be seen the results, column (d) are close to the ground truth, column (c), which was obtained by careful hand labeling of the pixels. The starfish example took seven iterations to converge (an iteration being one maximization of $\mathbf{x}$ via graph cut followed by one maximization of Θ via EM) starting from a rectangle with 49159 mislabelled pixels. After the first iteration this was dramatically reduced to 6161 pixels, and after seven iterations convergence was achieved with 1190 mislabelled pixels. It should be noted that most of the mislabelled pixels are on the boundary, characterized by 'mixed pixels' where the ground truth supplied by hand is somewhat suspect. In each case it was found that three components for foreground and background distributions was used.

3 Bayesian Novel-view Synthesis

This section addresses the problem of novel-view synthesis from a pair of rectified images with specific emphasis on gaze correction for one-to-one teleconferencing. The work is to appear in [Criminisi et al. 2003]. The approach uses a Bayesian Markov Random Field prior to aid the matching; which has second order properties.

With the rise of instant messenging technologies[2], it is envisaged that the PC will increasingly be used for interactive visual communication. One pressing problem is that any camera used to capture images of one of the participants has to be positioned offset to his or her gaze (*cf.* fig. 2 and fig. 3). This can lead to lack of eye contact and hence undesirable effects on the interaction [Gemmell et al. 2000].

One may think that if it where possible to drill a hole in the middle of a computer screen and place a camera there, that would achieve the desired gaze correction. The first problem with this solution is that "porous" screens do not exist; but even if they did they would not solve our gaze correction problem. In fact, in order to achieve the correct eye contact the user should always be required to look at the centre of the monitor (where the extra camera was inserted). On the contrary, during a messenging session, the user looks at the communication window (where the other person's face appears) which can be displaced and moved around the screen at will (fig. 3a). Therefore, we would need the camera to be placed always behind the messenging window on the screen. This cannot be achieved by existing hardware and, therefore, a software solution must be sought.

Previously proposed approaches can be broadly categorized as *model-based* or *image-based*. One model-based technique is to use a detailed 3D head model, texture map it and reproject it into the required viewpoints; whilst this can be successful [Vetter 1998; Yang and Zhang 2002], it is limited to imaging heads (with no hair or neck because of the poor quality of current head models), and would not, for example, deal with occlusion events such as a hand in front of the face.

A more general approach (employed in this paper) is to use image-based rendering techniques (*IBR* [Chen and Williams 1993]) to synthesize novel views from the analysis of some input images. This approach would "correct" the entire input images (as opposed to the head only), thus avoiding the detection and modeling of heads with all the associated problems.

The basic camera system considered in this paper [Criminisi et al. 2003] is illustrated in fig. 3. Given two input cameras, the aim is to generate a view from a virtual camera that is located roughly where the image of the head will be displayed on the screen for each participant, thus achieving the desired eye contact.

In IBR a depth map is combined with input images to produce synthetic views. In order to generate a depth map a dense stereo algorithm is required, a substantial review of which can be found in [Scharstein and Szeliski 2002a]. In [Scharstein and Szeliski 2002a] the authors categorize and evaluate a number of existing dense-stereo techniques. But this evaluation may not be valid for our purposes as: (i) the range of disparities considered in [Scharstein and Szeliski 2002a] is much smaller than in our application (0-29 pixels there, whereas we typically consider 0-80 pixel disparities); (ii) we are primarily interested in new-view synthesis, thus it does not matter if the disparities are relatively inaccurate in texture-less image regions, all that matters is that the new view is well synthesized (as noted in [Scharstein 1999; Szeliski and Golland 1999]); (iii) we consider long video sequences, thus stability of estimation also plays an important part: a temporally flickery reconstruction is less desirable than a stable one.

Furthermore, in the past, research on dense stereo reconstruction has based all its efforts towards the accurate recovery of disparity maps, and the role of occlusion regions has been somehow underestimated. We have found that while incorrect disparities may still produce sufficiently good synthesized images in the matched regions, inconsistent occlusion maps make the process of synthesis near foreground objects very unreliable.

[2] *e.g.* `messenger.msn.co.uk/`, `messenger.yahoo.com/`, `www.aol.co.uk/aim/`

This paper [Criminisi et al. 2003] stresses the importance of accurate occlusion detection in situations where this problem is most important, *i.e.* for large disparity-range imaging conditions. It will be demonstrated that, in that case, inaccurate occlusion estimation may lead to a number of undesired artifacts in the synthetic virtual image.

According to the evaluation in [Scharstein and Szeliski 2002a], two of the most powerful approaches use graph cuts [Kolmogorov and Zabih 2002; Roy and Cox 1998] and loopy belief propagation [Sun et al. 2002] but both of these are currently too computationally intensive for real-time applications. In fact, one of the goals of this paper [Criminisi et al. 2003] is that of producing high-quality synthetic sequences as close as possible to frame rate.

One of the most computationally efficient algorithms for stereo is Epipolar-line Dynamic-Programming [Ohta and Kanade 1985], commonly referred to as DP.

In [Belhumeur and Mumford 1992] the authors acknowledge the importance of occlusions (or *half-occlusions* using their terminology) in depth understanding. They employ a Bayesian approach in a DP framework for recovering occlusions and depth maps. The use of a simulated annealing step makes the algorithm not-suitable to deal with real time input video data.

The DP algorithm described in [Cox et al. 1996] has previously been demonstrated for cyclopean view interpolation [Cox et al. 1993] in video[3]. In the basic form of the DP algorithm, in order to obtain computational efficiency observations consist of single-pixel intensities. This, together with the fact that pairs of corresponding scanlines are considered independently introduces a number of artifacts which corrupt the quality of the output reconstruction (especially for large disparity ranges).

In particular, DP-based algorithms for novel-view synthesis are characterized by three kinds of artifacts: (i) artifacts produced by mismatches (horizontal streaks); (ii) the "halo" in the regions where the background is visible in only one of the two input views (occlusions); and (iii) flickery synthetic pixels, caused by matching ambiguities.

The first kind of artifacts are visible in [Bobick and Intille 1999] where the authors introduce the *Disparity Space Image* and employ a DP algorithm for finding the optimal path through it.

This paper [Criminisi et al. 2003] sets out to address and solve all three kinds of artifacts while maintaining high computational efficiency. There are *two* parts to our method: the first one is about generating accurate disparity and occlusion maps. The second part is about representing and using the extracted information to efficiently generate new views. Within this paper [Criminisi et al. 2003] we present new contributions in both areas: for the generation of disparity and occlusion maps we propose a new type of dynamic-programming approach, as path finding through a *four*-plane graph (as opposed to the traditional single-plane DP), introducing new labels to help the explicit and correct identification of occlusions, and altering the cost function employed to favour: (a) correct grouping of occlusions, (b) formation of solid occlusion regions at the boundaries of foreground objects, and (c) inter-scanline consistency. Second, we introduce the geometry of *minimum-cost surface* projection as an efficient technique for generating synthetic views from arbitrary virtual cameras *directly* from the minimum-cost surface obtained during the DP process[4]. This technique avoids the explicit construction of a 3D mesh model or depth map and presents a number of advantages that will be described in [Criminisi et al. 2003].

[3]We refer to *cyclopean view* as the image generated from a virtual camera located in the mid-point between the two input cameras.

[4]The *minimum-cost surface* is defined to be the collection of all the minimum-cost paths estimated (independently) by the DP algorithm at each scanline.

3.1 Novel-view synthesis results

This section presents a number of concluding synthesis results achieved on real input sequences. In particular, we demonstrate: gaze correction, cyclopean view generation, three-dimensional translation of the virtual camera, simple augmented-reality effects such as object insertion and replacement.

Gaze correction by cyclopean view synthesis for still stereo images. Figure 4 shows an example where the input left and right images have been used to generate the cyclopean view the proposed algorithm. In the output image (fig. 4b) the gaze has been corrected. Another example of gaze correction is illustrated in fig. 5.

3D translation of the virtual camera. Figure 6 shows an example of translating the virtual camera towards and away from the visualized scene. Notice that this is different from simple zooming or cropping of the output image. In fact, parallax effect may be noticed in the boundary between the head and the background, thus providing the correct three-dimensional feeling.

Figure 7 shows an example of in-plane translation (with $\mathbf{O}_v$ on the $X-Y$ plane) of the virtual camera. Notice the relative displacement of the head with respect to the background.

Cyclopean view generation in long sequences. Figure 8 demonstrates the effectiveness of the proposed algorithm for reconstructing cyclopean views of extended temporal sequences. Notice that most of the spatial artefacts (*e.g.* streaks, halo) and temporal artefacts (*e.g.* flickering) are removed.

Background replacement. The proposed algorithm generates novel, virtual views, but also a 3D representation of the observed scene. The latter can be advantageous for 3D editing of the visualized scene.

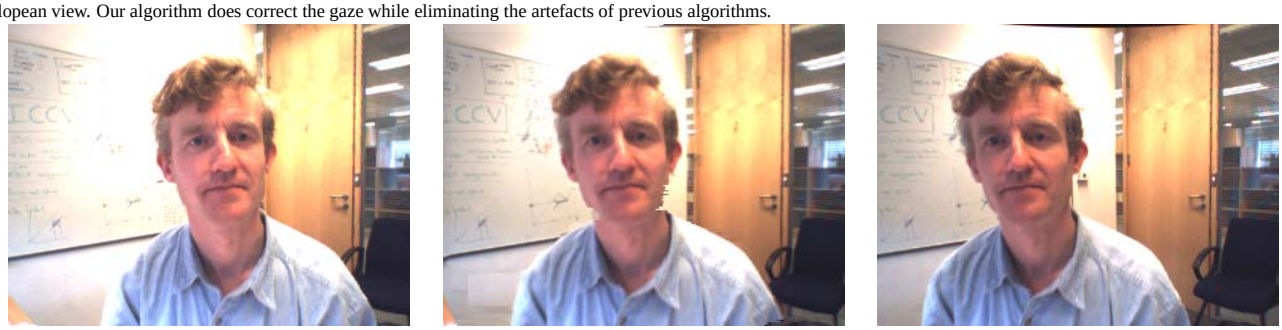
Figure 9 demonstrates the possibility of replacing the original background with a different one, either taken from real photographs or artificially generated. This is made possible thanks to the foreground/background segmentation step described in [Criminisi et al. 2003].

4 Bayesian Motion Capture/Tracking

One of the fundamental problems in vision is that of tracking objects through sequences of images. In [Stenger et al. 2003] we present a generic Bayesian algorithm for tracking the 3D position and orientation of rigid or non-rigid objects (in our application hands) in monocular video sequences. Great strides have been made in the theory and practice of tracking, e.g. the development of particle filters recognized that a key aspect in tracking was a better representation of the posterior distribution of model parameters [Gordon et al. 1993; Isard and Blake 1996]. Going beyond the uni-modal Gaussian assumption of the Kalman filter, a set of random samples is used to approximate arbitrary distributions. The advantage is that the filter can deal with clutter and ambiguous situations more effectively, by not placing its bet on just one hypothesis. However, a major concern is that the number of particles required increases exponentially with the dimension of the state space [Choo and Fleet 2001; MacCormick and Isard 2000]. Worse still, even for low dimensional spaces there is a tendency for particles to become concentrated in a single mode of the distribution [Doucet 1998]. Within this paper we wish to track a 27 DOF articulated hand (21 DOF for the joint angles and 6 for orientation and location) in cluttered images, without the use of markers. In [Wu et al. 2001] it is suggested that due to the strong correlation of joint angles, the state

a **b** **c**

Figure 4: **Example of gaze correction.** (a,c) Input left and right views, respectively; (a) looking slightly towards the right; (b) looking slightly towards the left; (b) Gaze-corrected cyclopean view. Our algorithm does correct the gaze while eliminating the artefacts of previous algorithms.



a **b** **c**

Figure 5: Yet another example of **gaze correction** from still images. The central image, (b) has been generated from the two input views (a,b) and shows correct gaze (the person is looking at us). The background (*e.g.* the door frame) has been reconstructed correctly despite its occlusion in the left input view. The "halo" effect and streaky artefacts have been removed.

space for the joints can be reduced to 7 DOF by applying PCA, with "loss of only 5 percent of information". Tracking is demonstrated for a fixed view with no clutter. However, if the hand is also to move, the state space has 13 dimensions, which is still considered high.

There are several possible strategies for estimation in this high dimensional space. One way is to use a hierarchical search, in which some parameters are estimated first, and then others, assuming that the initial set of parameters is correctly estimated. This strategy may seem suitable for articulated objects. For example, Gavrila [Gavrila and Davis 1996] suggests, in the context of human body tracking, first locating the torso and then using this information to search for the limbs. Unfortunately, this approach is in general not robust to different view points and self-occlusion. MacCormick and Isard [MacCormick and Isard 2000] propose a particle filtering framework for this type of method in the context of hand tracking, factoring the posterior into a product of conditionally independent variables. This assumption is essentially the same as that of Gavrila, and tracking has been demonstrated only for a single view point with no self-occlusion.

The development of particle filters was primarily motivated by the need to overcome ambiguous frames in a video sequence so that the tracker is able to recover. Another way to overcome the problem of losing lock is to treat tracking as object detection at each frame. Thus if the target is lost in one frame, this does not affect any subsequent frame. Template based methods have yielded good results for locating deformable objects in a scene with no prior knowledge, e.g. for hands or pedestrians [Athitsos and Sclaroff 2002; Gavrila 2000]. These methods are made robust and efficient by the use of distance transforms such as the chamfer or Hausdorff distance between template and image [Barrow et al. 1977; Huttenlocher D.P. et al. 1993]. These methods were developed for matching one template modulo projective transformations (e.g. translation, rotation etc.). A key suggestion was that multiple templates could be dealt with efficiently by building a tree of templates [Gavrila 2000; Olson and Huttenlocher 1997]. Given the success of these methods, it is natural to consider whether or not tracking might not be best effected by template matching using exhaustive or guided search at each frame. The answer to this question may be yes in some cases, but it depends strongly on the localizing power of the used feature set, otherwise a jerky motion will result. One approach to embed template matching in a probabilistic framework was proposed by Toyama and Blake [Toyama and Blake 2002]. However, it is acknowledged that "one problem with exemplar sets is that they can grow exponentially with object complexity. Tree structures appear to be an effective way to deal with this problem, and we would like to find effective ways of using them in a probabilistic setting" [Toyama and Blake 2002]. Within [Stenger et al. 2003] paper we address this problem.

4.1 Tree-Based Detection

When matching many similar templates to an image, a significant speed up can be achieved by forming a template hierarchy and using a coarse to fine search [Gavrila 2000; Olson and Huttenlocher 1997]. The idea is to group similar templates together and represent them with a single prototype template together with an estimate of the variance of the error within the cluster which is used to define a matching threshold. The prototype is first compared to the image; only if the the error is below the threshold are the templates within the cluster compared to the image. This clustering is done at various levels, resulting in a hierarchy, with the templates at the leaf level covering the space of all possible templates. Gavrila [Gavrila 2000] suggests forming the hierarchy by recursive (off-line) clustering, the goal being efficient on-line evaluation. When the exemplar templates are clustered using a cost function based on chamfer dis-

tance, a guarantee can be given that no better match is found in sub-trees, and objects are not missed when pruning sub-trees during the search. However, it is not straightforward how to give such guarantees when incorporating prior information for each template.

If a parametric object model is available, another option to build the tree is by partitioning the state space. Let this tree have L levels, each level l defines a partition $\mathscr{P}_l$ of the state space into N_l distinct sets $l = 1,\ldots,L$, such that $\mathscr{P}_l = \{\mathscr{S}^{li} : i = 1,\ldots,N_l\}$. The leaves of the tree define the finest partition of the state space $\mathscr{P}_L = \{\mathscr{S}^{Li} : i = 1,\ldots,N_L\}$. Such a tree is depicted schematically in figure 11(a), for a single rotation parameter. This tree representation has the advantage that prior information is encoded efficiently, as templates with large distance in parameter space are likely to be in different tree branches.

The hierarchical detector works well for locating hands in the images [Thayananthan et al. 2003] , and yet often there are ambiguous situations that could be solved by using temporal information. The next section describes the Bayesian framework for filtering. Filtering is the problem of estimating the state (hidden variables) of a system given a history of observations how to combine tree based detection and tracking is laid out in [Stenger et al. 2003].

4.2 Tracking Results

In the first sequence we track the global 3D motion of the hand without finger articulation. The 3D rotations are limited to a hemisphere. At the leaf level, the tree has the following resolutions: 15 degrees in two 3D rotations, 10 degrees in image restoration and 5 different scales. These 12960 templates are then combined with a search at 2-pixel resolution in the image translation space. Figures 12 and show results from tracking a pointing hand.

In the second sequence (figure 13) tracking is demonstrated for global hand motion together with finger articulation. The articulation parameters for the thumb and fingers are approximated by 7 and 5 divisions, respectively. For this sequence the range of global hand motion is restricted to a smaller region, but it still has 6 DOF. In total 35000 templates are used at the leaf level.

Note that in all three cases, the hand model was automatically initialized by searching the complete tree in the first frame of the sequence.

5 A Bayesian Estimation of Building Shape Using MCMC

Many algorithms have been developed for inferring 3D structure from a set of 2D images. A review of the state of the art in this area can be found at [Scharstein and Szeliski 2002b]. However there are often cases in which image information is ambiguous or misleading, such as in areas of homogeneous or repeated texture. In such cases extra information is required to obtain a model of the scene.

In the past, dense stereo algorithms have used heuristics favouring "likely" scenarios such as regularization or smoothing in an attempt to resolve these ambiguities, but in general these are unsatisfactory (for instance, the smooth surface assumption is violated at occlusion boundaries). It is our belief that maximum likelihood estimates (even regularized) of structure have progressed as much as they are able, and that further research in this area will yield negligible or arguable benefit. Our approach to structure from motion is to develop generic methods to exploit domain-specific knowledge to overcome these ambiguities. This has been successfully done for other 3D reconstruction domains, e.g. heads [Fua 1999; Shan et al. 2001], bodies [Plänkers and Fua 2001].

Within this paper [Dick et al. 2002] we explore the reconstruction of generic buildings from images, using strong prior knowledge of building form provided by architects, this is most naturally done in a Bayesian framework. The Bayesian framework provides a rational method for incorporating prior information into the estimation process. However in complicated scenarios such as the modelling of architecture, there still remains two problems to be resolved. Whilst Bayes provides the basic laws for manipulating probabilities, we still need to resolve the problem of parameterization, and once the problem is parameterized choose the best algorithm to optimize the parameters.

Structure is represented as a collection of planes (corresponding to walls) and primitives (representing windows, doors and so on). Each primitive is defined by several parameters. The advantages of this model-based approach are that it enables the inference of scene structure and geometry where evidence from the images is weak, such as in occluded regions or areas of homogeneous texture, and that it provides an interpretation of the scene as well as its geometry and texture [Dick et al. 2001]. The representation of the scene as a set of planes and primitives is useful for reasoning about the scene during reconstruction and for subsequent rendering and manipulation of the model. The compactness of the representation also makes recovery of structure and motion more reliable.

In previous work [Dick et al. 2001] a framework was defined for model based structure from motion for buildings. In this framework an algorithm for estimating a maximum a posteriori (MAP) estimate of the model based on priors and image likelihood measures was proposed. However the spatial prior used in this work applied only to the parameters of each individual primitive, thus ignoring information about their spatial juxtaposition (for instance, that windows are likely to occur in rows and columns). In this paper the spatial prior is expanded to include this sort of information.

The form which the spatial prior should take is far from obvious. Ideally it should admit all plausible buildings while excluding those which are for practical or aesthetic reasons implausible. However the plausibility of a structure can in general only be verified by manual inspection. Thus a crucial step in the formulation of the prior is to test it by drawing sample buildings from it and checking that they appear reasonable. However even with expert knowledge, it is very difficult to explicitly represent the probability density function (pdf) of a suitable prior. What is somewhat easier to do is to express the prior as a scoring function that favours particular configurations, such as windows in rows. One approach is to use a scoring function suggested by an expert and then draw samples from the implicitly defined pdf using an MCMC algorithm. If the samples drawn look like reasonable buildings then the prior must be close to the true prior.

This raises the question of just how close to the true prior our estimate must be to generate reasonable looking models. To answer this empirically the scoring function is varied both on a small scale and a large scale, and the effect on models generated from the prior, and reconstructions obtained using the prior and an image sequence, is observed.

5.1 Building Results

When jump types involving wall plane parameters are included in the MCMC algorithm, closure of the building is enforced and the reconstruction converges to a symmetric model such as that shown in Figure 14. The texture for this model is cut and pasted from areas of the image identified as a wall, window, columns and so on, and the same texture sample is used for every instance of a type of primitive. Another feature of using an MCMC algorithm to sample the posterior is that as well as having a MAP model estimate, other probable samples can also be examined. This is useful for identifying ambiguities in the reconstruction. Four of the more marked ambiguities present in this model are shown in Figure 14 (i)-(l).

The operation of this algorithm is shown in Figure 15 for the

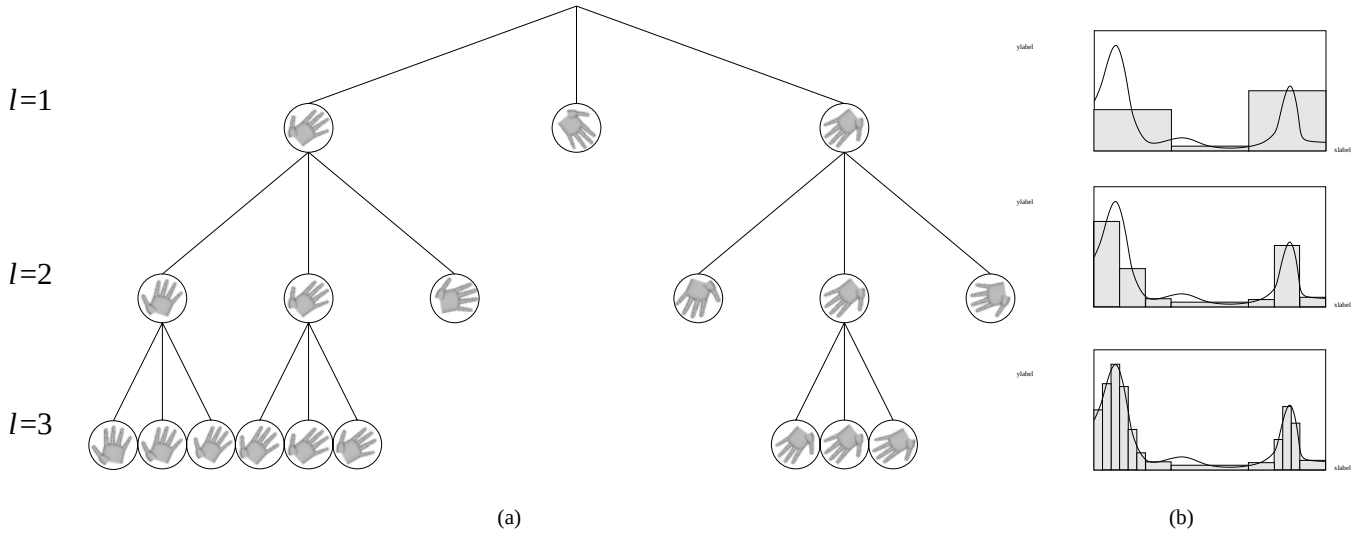


Figure 11: **Tree-based estimation of the posterior density.** *(a) Associated with the nodes at each level is a non-overlapping set in the state space, defining a partition of the state space (here rotation angle). The posterior pdf for each node is evaluated using the center of each set, depicted by a hand rotated by a specific angle. Sub-trees of nodes with low posterior probability are not further evaluated. (b) Corresponding posterior density (continuous) and the piece-wise constant approximation using tree-based estimation. The modes of the distribution are approximated with higher precision at each level.*

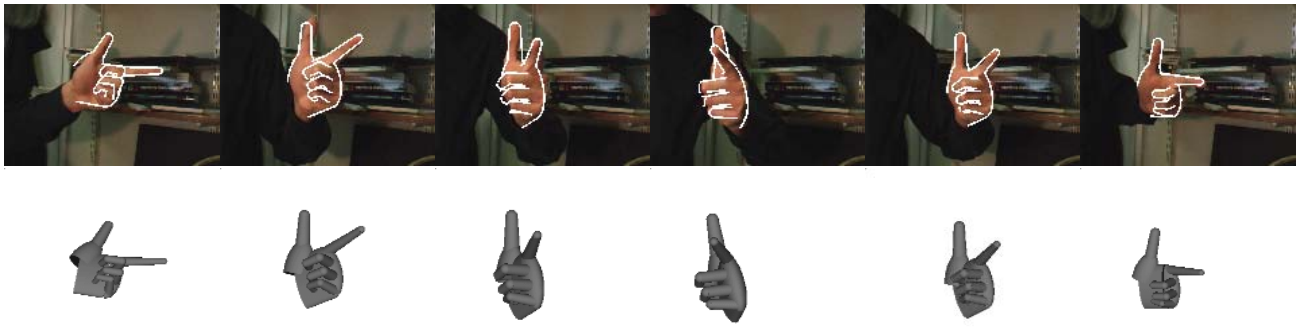


Figure 12: **Tracking a pointing hand in front of clutter**. *The images are shown with projected contours superimposed (top) and corresponding 3D model (bottom).*

Trinity Chapel sequence. Note that the entire model is obtained from only 3 images. Although the model is not completely accurate in areas which are not visible in the images, it is a plausible structure, and is obtained automatically except for the prior specification of the structure as being Gothic, and the restriction of the variety and shape of primitives this entails. The width of each part of the building is obtained from the average size of the window or door primitives on visible walls—each segment of the building is made wide enough to accommodate one window of height and width equal to the average height and width of the visible windows, with spacing to either side equal to half the window width. In the absence of image information, this seems a reasonable assumption to make and produces generally plausible architectural models.

6 Conclusion

Within this paper several Bayesian approaches to solve some problems in Graphics have been outlined. Due to lack of space the details have been omitted and only a problem description together with result shown, in that the hope that, if the reader is interested he will contact me or seek the references. However in general the results should help convince the reader that their is more benefit to a Bayesian analysis than a post hoc dressing up in many problem areas.

The power in these approaches is that they provide a systematic way to perform estimation of parameters, to encode high level prior information and to encode uncertainty in the estimates. Automatic estimation of parameters is particularly important in the case of image cut out and dense stereo for new view synthesis as the results can be very sensitive to the wrong choice of parameters. Encoding of uncertainty can help a tracker recover from ambiguous situations. Indeed the case of articulated hand tracking is often highly ambiguous with many interpretations for the hand pose at any given time instances. The use of high level prior information turns out to be very useful in the reconstruction of architectural scenes, and again the Bayesian method yields a consistent way to use learn and use this prior information to make an estimate of 3D structure.

Bayesian methods are not appropriate for all classes of problems, however problems involving uncertainty or learning will certainly benefit from a Bayesian analysis and it is certain that a understand-

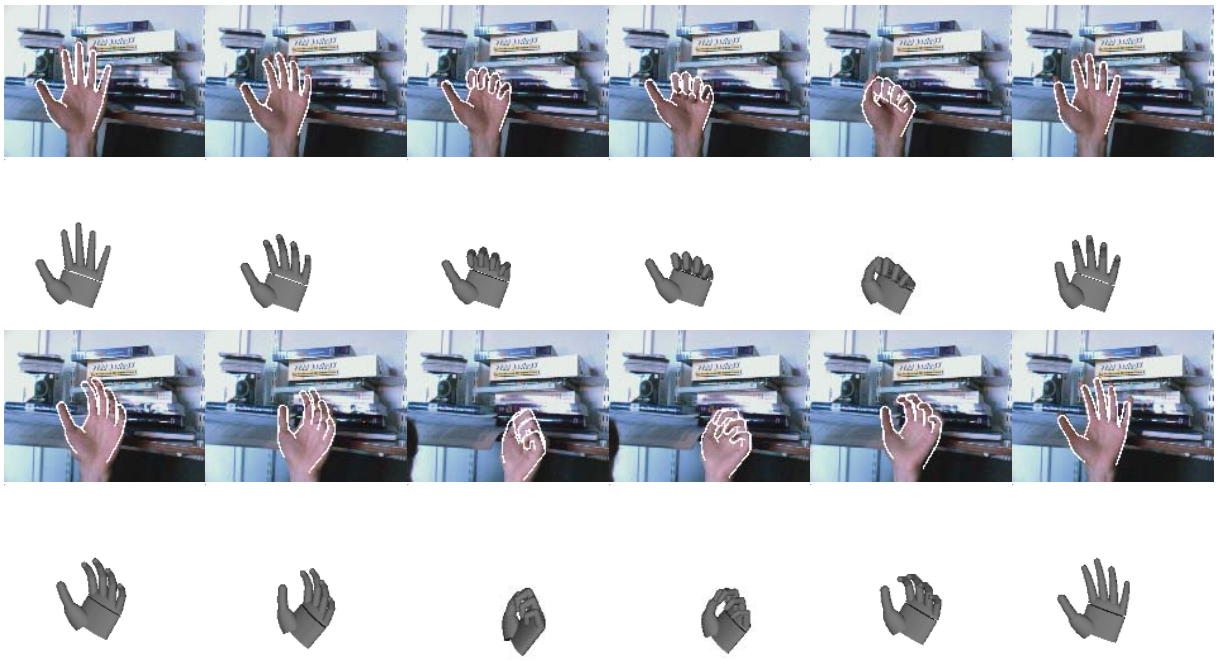


Figure 13: **Tracking a hand opening and closing with rigid body motion in front of clutter.** *In this sequence 6 DOF for rigid body motion plus 2 DOF for finger flexion and extension are tracked successfully.*

ing of these methods will benefit many Graphics researchers.

References

1999. *Adobe PhotoShop Version 5.5 (Chapter 7: Selecting)*. Adobe Systems Inc.

ATHITSOS, V., AND SCLAROFF, S. 2002. An appearance-based framework for 3D hand shape classification and camera viewpoint estimation. In *IEEE Conference on Face and Gesture Recognition*, 45–50.

BARROW, H. G., TENENBAUM, J. M., BOLLES, R. C., AND WOLF, H. C. 1977. Parametric correspondence and chamfer matching: Two new techniques for image matching. In *Proc. 5th Int. Joint Conf. Artificial Intelligence*, 659–663.

BELHUMEUR, P., AND MUMFORD, D. 1992. A Bayesian treatment of the stereo correspondence problem using half-occluded regions. In *IEEE Comp. Soc. Conf. on Comp. Vision and Pattern Recognition*, 506–512.

BESAG, J. 1974. Spatial interaction and statistical analysis of lattice systems. *J. Roy. Stat. Soc. Lond. B. 36*, 192–225.

BLAKE, A., ROTHER, C., AND TORR, P. H. S. 2004. GRABCUT an interactive segmentation tool. To appear.

BOBICK, A., AND INTILLE, S. 1999. Large occlusion stereo. *IJCV 33*, 3 (September), 1–20.

BOYKOV, Y., AND JOLLY, M. 2001. Interactive graph cuts for optimal boundary and region segmentation of objects in n-d images. In *ICCV01*, I: 105–112.

CHEN, E., AND WILLIAMS, L. 1993. View interpolation for image synthesis. In *SIGGRAPH*, 279–288.

CHOO, K., AND FLEET, D. J. 2001. People tracking using hybrid monte carlo filtering. In *Proc. 8th Int. Conf. on Computer Vision*, vol. II, 321–328.

COX, I., OTT, M., AND LEWIS, J. 1993. Videoconference system using a virtual camera image. *US Patent 5,359,362*.

COX, I., HINGORANI, S., AND RAO, S. 1996. A maximum likelihood stereo algorithm. *Computer vision and image understanding 63*, 3, 542–567.

CRIMINISI, A., SHOTTON, J., BLAKE, A., AND TORR, P. H. S. 2003. Efficient dense-stereo and novel-view synthesis for gaze manipulation in one-to-one video teleconferencing. To appear ICCV 2003.

DANEELS, D., CAMPENHOUT, D., NIBLACK, W., EQUITZ, W., BARBER, R., BELLON, E., AND FIERENS, F. 1993. Interactive outlining: An improved approach using active contours. storage and retrieval for image and video databases. In *SPIE93*, 226–233.

DICK, A., TORR, P. H. S., RUFFLE, S., AND CIPOLLA, R. 2001. Combining single view recognition and multiple view stereo for architectural scenes. In *ICCV01*, 268–274.

DICK, A., TORR, P. H. S., AND CIPOLLA, R. 2002. A bayesian estimation of building shape using mcmc. In *ECCV2002*, 852–866.

DOUCET, A. 1998. On sequential simulation-based methods for bayesian filtering. Tech. Rep. CUED/F-INFENG/TR310, Dept. of Engineering, University of Cambridge, Cambridge, UK.

FUA, P. 1999. Regularized Bundle-Adjustment to Model Heads from Image Sequences without Calibration Data. *International Journal of Computer Vision 38*, 2 (July).

GAVRILA, D. M., AND DAVIS, L. S. 1996. 3-D model-based tracking of humans in action: a multi-view approach. In *Proc. Conf. Computer Vision and Pattern Recognition*, 73–80.

GAVRILA, D. M. 2000. Pedestrian detection from a moving vehicle. In *Proc. 6th European Conf. on Computer Vision*, vol. II, 37–49.

GEMAN, S., AND GEMAN, D. 1984. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Trans. on Pattern Analysis and Machine Intelligence 6*, 6, 721–741.

GEMMELL, J., TOYAMA, K., ZITNICK, C., KANG, T., AND SEITZ, S. 2000. Gaze awareness for video-conferencing: A software approach. *IEEE Multimedia 7*, 4.

GORDON, N. J., SALMOND, D. J., AND SMITH, A. F. M. 1993. Novel approach to non-linear and non-gaussian bayesian state estimation. *IEE Proceedings-F 140*, 107–113.

HUTTENLOCHER D.P., NOH J.J., AND RUCKLIDGE W. J. 1993. Tracking non-rigid objects in complex scenes. In *Proc. 4th International Conference on Computer Vision, Berlin*, IEEE Computer Society Press, Los Alamitos, CA, 93–101.

ISARD, M., AND BLAKE, A. 1996. Visual tracking by stochastic propagation of conditional density. In *Proc. 4th European Conf. on Computer Vision*, 343–356.

JAYNES, E. T. 2003. *Probability Theory, The Logic of Science*. Cambridge University Press.

KASS, M., WITKIN, A., AND TERZOPOULOS, D. 1987. Snakes: Active contour models. In *Proc. 1st Int. Conf. on Computer Vision*, 259–268.

KOLMOGOROV, V., AND ZABIH, R. 2002. Multi-camera scene reconstruction via graph cuts. In *Proc. Europ. Conf. Computer Vision*, 82–96.

MACCORMICK, J., AND ISARD, M. 2000. Partitioned sampling, articulated objects, and interface-quality hand tracking. In *Proc. 6th European Conf. on Computer Vision*, vol. 2, 3–19.

MORTENSEN, E., AND BARRETT, W. 1998. Interactive segmentation with intelligent scissors. *GMIP 60*, 5 (September), 349–384.

OHTA, Y., AND KANADE, T. 1985. Stereo by intra- and inter-scan line search using dynamic programming. *IEEE Trans. on Pattern Analysis and Machine Intelligence 7*, 2, 139–154.

OLSON, C. F., AND HUTTENLOCHER, D. P. 1997. Automatic target recognition by matching oriented edge pixels. *Transactions on Image Processing 6*, 1 (January), 103–113.

PEREZ, P., BLAKE, A., AND GANGNET, M. 2001. Jetstream: Probabilistic contour extraction with particles. In *ICCV01*, II: 524–531.

PLÄNKERS, R., AND FUA, P. 2001. Articulated Soft Objects for Video-based Body Modeling. In *International Conference on Computer Vision*.

ROY, S., AND COX, I. 1998. A maximum-flow formulation of the n-camera stereo correspondence problem. In *ICCV6*, Narosa Publishing House, U. Desai, Ed., 492–499.

SCHARSTEIN, D., AND SZELISKI, R. 2002. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *Int. J. Computer Vision 47*, 1–3, 7–42.

SCHARSTEIN, D., AND SZELISKI, R. 2002. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *IJCV 47*, 1, 7–42. Evaluation page http://www.middlebury.edu/stereo/eval/.

SCHARSTEIN, D. 1999. *View Synthesis Using Stereo Vision*, vol. 1583 of *Lecture Notes in Computer Science (LNCS)*. Springer-Verlag.

SHAN, Y., LIU, Z., AND ZHANG, Z. 2001. Model based bundle adjustment with applications to face modeling. In *ICCV Vol 2*, IEEE, 644–651.

STENGER, B., THAYANANTHAN, A., TORR, P., AND CIPOLLA, R. 2003. Bayesian tracking using tree-based density estimation. To appear ICCV 2003.

SUN, J., SHUM, H. Y., AND ZHENG, N. N. 2002. Stereo matching using belief propagation. In *Proc. Europ. Conf. Computer Vision*.

SZELISKI, R., AND GOLLAND, P. 1999. Stereo matching with transparency and matting. *IJCV 32*, 1, 7–27.

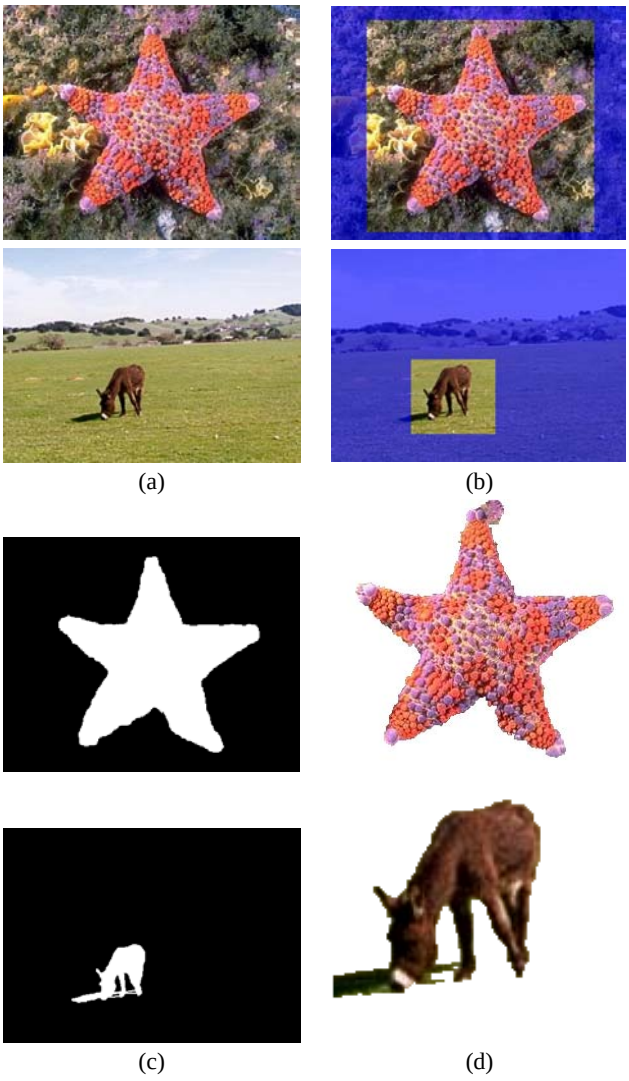
THAYANANTHAN, A., STENGER, B., TORR, P. H. S., AND CIPOLLA, R. 2003. Shape context and chamfer matching in cluttered scenes. In *Proc. Conf. Computer Vision and Pattern Recognition*. to appear.

TOYAMA, K., AND BLAKE, A. 2002. Probabilistic tracking with exemplars in a metric space. *Int. Journal of Computer Vision* (June), 9–19.

VETTER, T. 1998. Synthesis of novel views from a single face image. *Int. J. Computer Vision 28*, 2, 103–116.

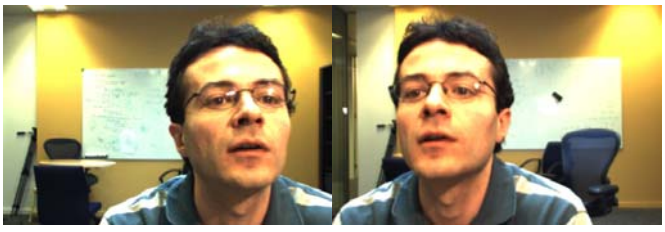
WU, Y., LIN, J. Y., AND HUANG, T. S. 2001. Capturing natural hand articulation. In *Proc. 8th Int. Conf. on Computer Vision*, vol. II, 426–432.

YANG, R., AND ZHANG, Z. 2002. Eye gaze correction with stereovision for video tele-conferencing. In *Proc. Europ. Conf. Computer Vision*, vol. 2, 479–494.



(a) (b) (c) (d)

Figure 1: *(a) The 'starfish' and 'donkey' image. (b) shows the rectangle selected by the user, drawn so as to loosely bound the object of interest. (c) The ground truth, generated by hand, there are 49159 mislabelled pixels in the starfish case, 6455 in the donkey. (d) The resulting cut out obtained after iterating to convergence, with 1190 and 357 mislabelled pixels respectively. In each case it was found that three components for foreground and background distributions was used.*



a **b**

Figure 2: During one-to-one video-conferencing, cameras located on the frame of the computer monitor fail to capture the correct gaze. In this example the person is looking at the centre of the screen and does not appear to be looking at us. The aim of the proposed algorithm is that of correcting the distorted gaze and make the person appear as if he/she was naturally looking at us.

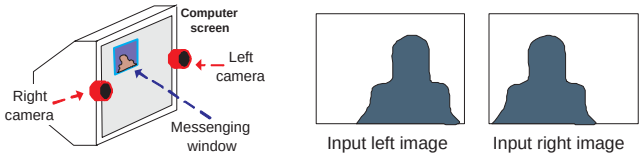


Figure 3: **Camera-computer configuration.** The basic setup considers two cameras placed on the frame of the computer monitor. The window for the one-to-one teleconferencing application is marked in blue on the computer screen. The goal of this work is that of achieving the correct eye contact by efficient processing of the two input images to generate high-quality views for virtual cameras placed behind the messenging application window. the technique described in this work achieves gaze correction in an efficient and compelling way.

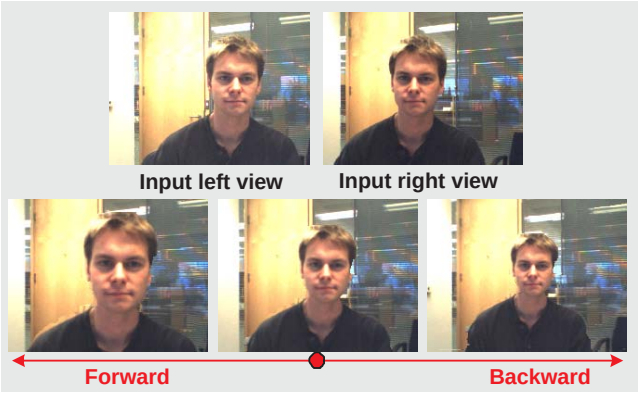


Figure 6: **Forward/backward translation of virtual camera.** The bottom row shows the synthesized cyclopean views with (left) forward virtual camera translation, (centre) no virtual camera translation, (right) backward virtual camera translation. Notice the *parallax* effect around the head.

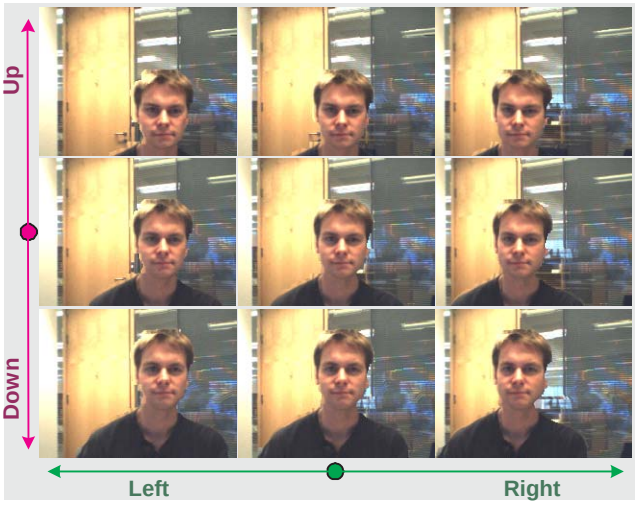


Figure 7: **In-plane translation of virtual camera.** The left and right input images are the same as in fig. 6. This table shows the synthesized images corresponding to translation of the virtual camera along the x and y axes. Notice the *parallax* effect around the head. Also, the door frame is reconstructed nicely despite it being partially occluded in the right input view.

Figure 8: **Cyclopean image synthesis for long sequences.** Frames extracted from a reconstructed cyclopean sequence (over 10 sec long). The input images are not shown here.

Figure 9: **Background replacement.** The techniques developed in this paper allow, amongst other things, for the foreground to be segmented from the background. This, in turn, allows the real background to be replaced by more interesting and prettier images, or video-textures as shown in this case.

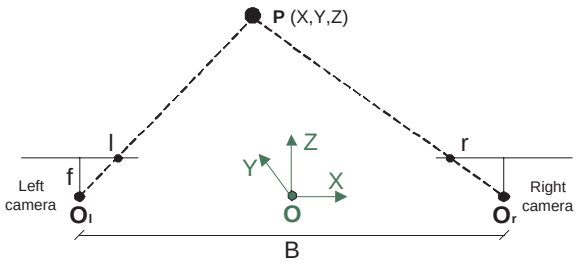


Figure 10: **Basic camera configuration and notation.** $\mathbf{O}_l$ and $\mathbf{O}_r$ are the optical centres of left and right cameras respectively, f is the focal length of the cameras (identical for both cameras) and B is the baseline between the two optical centres. The origin of the reference coordinate system X,Y,Z is denoted $\mathbf{O}$.

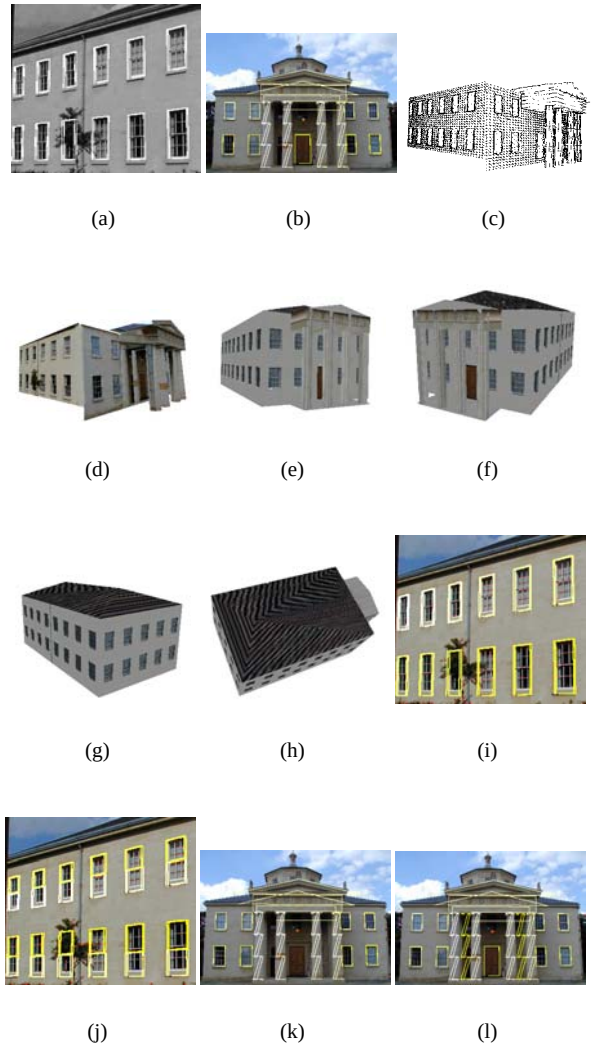


Figure 14: *(a) MAP model of side wall of Downing library, after 2000 MCMC iterations using the Classical prior. (b) Front wall. Both front and back faces of primitives are drawn, hence the pair of triangles and rectangles for the pediment and entablature. (c)-(d) 3D rendering of MAP Downing model, obtained without using add/remove/delete wall jumps. The textures shown on the model are automatically extracted from the images which are most front-on to each plane. (e)-(h) Four views of the completed model of Downing library, with extra walls added. Even though only two walls are visible, a complete building has been modelled using symmetry. Wall, window, roof and column textures are sampled from the images and applied to the appropriate primitives. (i)-(l) Some ambiguities in the Downing model, chosen from the 20 most probable models visited by the MCMC process. (i) Window sills are included in the window primitives. (j) Windows are represented using two primitives each. (k) The door is omitted. (l) Extra columns are added in between the existing ones.*

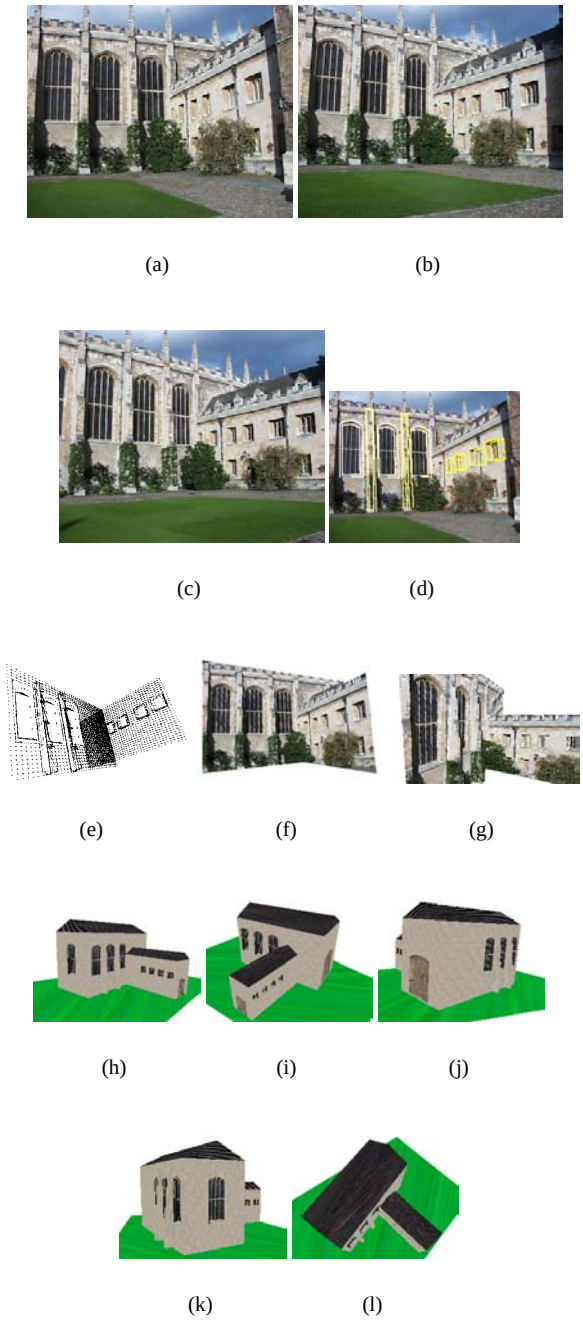


Figure 15: *(a)-(c) 3 original images of Trinity college courtyard. (d) MAP model primitives, superimposed on image. (e) Wireframe MAP model. (f)-(g) 3D model with texture taken from images. (i)-(m) Five views of the completed model of the north-east corner of Great Court, Trinity College. Only two of the walls are visible in the images.*