Bayesian Surface Estimation from Multiple Cameras Using a Prior Based on the Visual Hull and its Application to Image Based Rendering

Adam Bowen, Andrew Mullins, Roland Wilson and Nasir Rajpoot
Department of Computer Science
University of Warwick, UK
{fade,andy,rgw,nasir}@dcs.warwick.ac.uk

Abstract

The problem of visible surface estimation for image-based rendering is tackled using a new approach, which combines visual hull and surface estimation techniques. It is shown that the new method combines the best features of both approaches, being more robust than direct surface element estimation and more flexible than the visual hull. The new method uses an estimate of the visual hull as a prior on the ill-posed problem of surface element estimation. To improve the computational preformance of the algorithm, a multiresolution approach to surface patch estimation is used. The patches thus estimated can then be tracked over time, to provide an accurate model of the surface geometry, which is then used for estimation of the scene from arbitrary viewpoints. After a brief description of the algorithm, results are presented to show the improvement in performance which can be obtained using either technique alone. The paper concludes with a discussion of extensions to the work currently under investigation.

1 Introduction

The problem of image-based rendering has been much studied and ranges in complexity from simple interpolation techniques [2], [3], [8] to geometry estimation and space carving [4], [19], [16], [20], [5]. While simple interpolation approaches are adequate if a sufficiently dense set of images of the scene is available, in many practical situations, the number of cameras limits the accuracy of interpolations and leads to visible artefacts in the reconstructions. The alternative is an explicit computation of surface geometry, which allows more conventional, graphics based rendering to be used. Unfortunately, even with multiple cameras, estimation of visible surfaces remains an ill-posed problem, especially if the objects imaged are as complex as the human body. Two common approaches have been used to estimate scene geometry: surface estimation techniques [4], [19], [15], [18], and the so-called visual hull [7], [1], [16], [20]. Although the former is inherently more powerful, it suffers from the ill-posed nature of the problem: it depends heavily on the surface texture and smoothness of the 3-D shape of the objects in the scene; it is also highly demanding computationally. Visual hull techniques, on the other hand, require accurate segmentation of the projected images and make simplifying assumptions about the convexity of the object [10], require a large amount of memory, and suffer from quantization effects [9], but they lead to robust estimates; they are also reasonably fast computationally. A method which combines the features of the two seems like an obvious way to overcome the limitations of either.

Previous work in this area has used both determinstic [19] and simple stochastic methods [6] for hull refinement. In this paper, we present such a method, which uses a stochastic Bayesian framework for the computation of planar surface elements but employs a prior on those elements, derived from the visual hull. Such surface elements are a common respresentation of surface geometry [15], [18], [17], and provide a more general representation than similar surface descriptions such as wireframe meshes. The input to the algorithms are a set of video sequences together with camera calibration information for those sequences. The calibration information is obtained separately using the methods described in [22].

After a brief discussion of the surface representation and a description of the estimation algorithms, we present some results to show the performance of the new method, which significantly outperforms our earlier methods, giving more accurate surface estimates in a reduced computation time. The paper is concluded with a discussion of the power and limitations of the new approach and suggestions for further work.

2 Surface Estimation Method

First, a multiresolution segmentation algorithm is applied to the images captured from the scene to separate figure and background. From the silhouettes derived from the segmentation, a view-dependent visual hull is constructed following a similar method to that described in [11]. From the visual hull the depth and surface orientation along any ray in the scene may be derived using total least squares, a robust method of fitting planes to surface data [12]. These estimates are used as a prior in a multiscale particle filter, which provides the final surface estimates, in terms of a number of disjoint quadrilaterals, corresponding to square blocks in each image of the scene. The patches can then be tracked over time using a second particle filter [14], and used to reconstruct arbitrary views of the scene using an adaptation of a conventional graphics renderer, giving real-time reconstructions [13], [17].

2.1 Surface Representation

A view dependent surface model is defined as follows: partition each input image into pixel blocks and assume each block is the projection of a quadrilateral region of a surface in world space. For notational convenience it is assumed that every block has a unique index n. The class of quadrilaterals corresponding to image blocks will subsequently be refered to as ‘patches’.

The centroid of a patch which corresponds to a block n, must lie along the line which passes through both the imaging camera, and the focal plane at the point which corresponds to the centre of the block (such lines will be refered to as ‘pixel rays’ for obvious reasons). A compact representation of the position of a patch is its distance d_n from the camera, along this pixel ray.

The orientation of a patch must lie between face-on, and almost perpendicular to, the imaging camera. In an appropriately chosen Cartesian frame where the z-axis aligns with the pixel ray for the centre of block n, the orientation of the patch may be represented using the first two components (i_n, j_n) of its surface normal, with the third component having unit length. This representation allows all allowable orientations, yet prevents the patch from becoming oriented perpendicular, or beyond perpendicular, to the imaging camera. The complete representation of a patch corresponding to block n is then

$$x_n = (d_n, i_n, j_n). \tag{1}$$

For convenience in the later discussion, $f_n(x_n)$ will denote the plane defined by x_n in some fixed Cartesian frame common to all patches.

2.2 Locally Adaptive Foreground-Background Segmentation

For a camera c in a data set, its image will be denoted I_c, and its Gaussian pyramid decomposition $I_c^{[0]} \dots I_c^{[M]}$, where $I_c^{[0]} = I_c$. For a pixel (u, v) in an image level $I_c^{[m]}$, its labelling $s(u, v)$ as either foreground or background is performed using a likelihood model derived from a segmentation of the previous image pyramid level $I_c^{[m+1]}$. One global approach using the above idea would be to evaluate the likelihood of a labelling $s(u, v) = l$ as

$$p_s(I_c^{[m+1]} | s(u, v) = l) = \frac{1}{K} \sum_{k=0}^{K} N(I_c^{[m]}(u, v)\ ; \mu_l^{[k]}, \sigma_l^{[k]}), \tag{2}$$

with the K Gaussian mixture components being derived from the colours of the pixels in class l at the previous image level. However, unless the number of components is very large, image details will not feature prominently in the mixture and may be misclassified.

Adopting a more local approach, our algorithm proceeds as follows: firstly each pixel is labelled using the classification of its 'parent' pixel from the previous image level, in quad-tree fashion. Then the image is partitioned into blocks and the segmentation refined with a two pass approach:

Pass one: For each block: if every pixel within the block and that of its neighbours is of the same class, mark the block as classified.

Pass two: The remaining unclassified blocks form a 'corridor of uncertainty' [21]. For each unclassified block,

i Search within an increasing radius until one or more blocks are found from each class which were classified in the first pass.

ii Form a Gaussian mixture colour likelihood model for both classes by clustering the pixel values within these classified blocks.

iii Classify each pixel within the current block as either foreground or background based on the local likelihood model.

As part of the third step, it is possible to use a Markov random field using the likelihood models and a smoothness prior as in [23], although this was found to not improve the results significantly given the extra computation required. The only remaining issue is how a classification is produced at the lowest resolution. It was found that a simple foreground-background model based on a background reference image provided a reasonable starting point.

2.3 Surface Prior Construction

Initially, a view dependent hull is constructed. For every pixel u, v in a particular image level $I_c^{[m]}$, it is possible to estimate a point which lies on the surface of the visual hull as follows:

i Generate a foreground-background segmentation for every camera.

ii Intersect the pixel ray for each pixel u, v labelled as foreground in image $I_c^{[m]}$ with the back-projected foreground regions in all other cameras $c' \neq c$ to produce zero or more lines in world space.

iii The closest line end point p_{uv} to camera c is the point on the visual hull which projects into pixel u, v.

The algorithm to construct a surface prior $\hat{x_n}$ for block n then proceeds as follows:

i Calculate $M = \frac{1}{|P_n|} \sum_{(u,v)\in P_n} (p_{uv} - \bar{p})(p_{uv} - \bar{p})^T$, where $\bar{p} = \frac{1}{|P_n|} \sum_{(u,v)\in P_n} p_{uv}$, and where P_n is the set of pixels in block n.

ii Compute the eigenvectors and eigenvalues of M.

iii The eigenvector corresponding to the smallest eigenvalue is the normal to the best-fit plane through the data, and together with $\bar{p}$ defines a plane y_n.

The prior can then be chosen for block n and has mean $\hat{x_n} = f_n^{-1}(y_n)$, and covariance $\hat{\sigma}_n$ which is chosen empirically.

2.4 Multiresolution Particle Filter

A particle filter is used which works across scale to produce surface estimates. For each patch at a given image level, the algorithm draws a set of samples $X = \{x^{[1]} \ldots x^{[S]}\}$ from an importance sampling function,

$$X \sim q(x_n | x_{h(n)}, I^{[m]} \ldots I^{[M]}), \tag{3}$$

where $h(n)$ is the parent block of block n.

The samples drawn from the importance sampling distribution are weighted according to

$$w_n^{[s]} \propto w_{h(n)}^{[s]} \frac{p(I^{[m]} | x_n^{[s]}) p(x_n^{[s]} | x_{h(n)}^{[s]})}{q(x_n | x_{h(n)}, I^{[m]} \ldots I^{[M]})}, \tag{4}$$

where the process model for the parameters of block n is simply

$$p(x_n | x_{h(n)}) = N(x_n;\ f_n^{-1}(f_{h(n)}(x_{h(n)})), \sigma_p) \tag{5}$$

i.e. the parameters corresponding to the same plane as that defined by its parent, but with some process noise with covariance σ_p added.

The measurement likelihood $p(I^{[m]} | x_n)$ is defined as follows: for any pixel (u, v) in block n, intersecting its pixel ray with the plane $f_n(x_n)$ produces a point in world space.

This point may be projected into any other camera's images and the colour of the original pixel compared to that in the projected location. Intuitively, the more similar the colours, the more likely that x_n are the correct parameters for the block. This mapping between a pixel (u,v) belonging to a block n in camera c, and the pixel (u',v') to which it projects in camera c' given that the block has parameters x_n is defined as $(u',v') = g^{c \to c'}(u,v,x_n)$.

Following this intuition we define a likelihood for a patch, given a set of multiscale image data from the cameras. The likelihood of x_n, given one camera's image level $I_{c'}^{[m]}$ is

$$p(I_{c'}^{[m]}|x_n) = \frac{1}{|P_n|} \sum_{(u,v) \in P_n} N(I_{c'}^{[m]}(g^{c \to c'}(u,v,x_n)) \; ; I_c^{[m]}(u,v), \sigma), \tag{6}$$

That is, the colour of the re-projected pixel is assumed to be normally distributed, with covariance σ about the colour of the pixel from the original block. Given all the images $I^{[m]} = \{I_1^{[m]} \ldots I_C^{[m]}\}$ from all cameras in the data set at level m, the likelihood of a patch is

$$p(I^{[m]}|x_n) = \frac{1}{|N_c|} \sum_{c' \in N_c} p(I_{c'}^{[m]}|x_n), \tag{7}$$

where N_c are the spatial neighbours of camera c.

3 Results

Our locally adaptive segmentation provides a clear advantage over the Markov random field approach in dealing with background regions of images onto which a shadow has been cast. Figure 4 highlights the class boundary for the two methods for an image taken from one of our data sets. Although the Markov random field has corrected the white regions on the t-shirt which are classified as background under using a naive approach, it has also misclassified several areas of shadow close to the body as foreground. The locally adaptive model as prevented this misclassification.

The results in table 3 indicate the mean squared error in colour across all patches when they are reprojected into neighbouring cameras, using different patch estimation algorithms for two data sets. The first contains a static person, whilst the second contains a person in motion which presents a significant challenge for the likelihood model due to motion blur. 'MSMC' is the multi-scale Monte-carlo algorithm which simply estimates patches using a particle filter across scale, but a weak prior. The 'VH' algorithm produces surface estimates derived from the visual hull alone. The 'VH + MC' algorithm estimates the patches using Monte-carlo simulation at the finest resolution using the visual hull to provide a prior. Finally, the 'VH + MSMC' algorithm is the full multiscale Monte-carlo algorithm using the visual hull as a prior, as described in section 2.4. This algorithm clearly outperforms all other approaches.

The ultimate objective of this work is to render images from a new point of view not in the original data set. Figure 2 shows three reconstructions produced from the patches using the 'MSMC' algorithm, the 'VH' algorithm and the 'VH + MSMC' algorithm.

Data set	MSMC	VH	VH + MC	VH + MSMC
1	1571.9	2563.4	2318.5	1374.3
2	n/a	9179.8	9268.7	5585.8

Table 1: MSE of estimated patch projections.

4 Conclusions and Further Work

We have shown the power of a combining hull and surface estimation within a Bayesian framework. It significantly increases the accuracy and speed of object geometry estimation. In addition, we have proposed a novel multiscale method for producing the foreground-background segmentations which are used to generate the visual hull. As a part of future work we would like to investigate the possibility of using hull information for the estimation of frame-to-frame motion.

References

[1] Adrian Broadhurst, Tom Drummond, and Roberto Cipolla. A probabilistic framework for space carving. pages 388–393.

[2] E. Chen and L. Williams. View interpolation for image synthesis, 1993.

[3] S. Chen. Quicktime vr — an image-based approach to virtual environment navigation, 1995.

[4] S. Gortler, R. Grzeszczuk, R. Szeliski, and M. Cohen. The lumigraph, 1996.

[5] A. Hilton and J. Starck. Multiple view reconstruction of people.

[6] J. Isidoro and S. Sclaroff. Stochastic refinement of the visual hull to satisfy photometric and silhouette consistency constraints.

[7] Kiriakos N. Kutulakos and Steven M. Seitz. A theory of shape by space carving. Technical Report TR692, 1998.

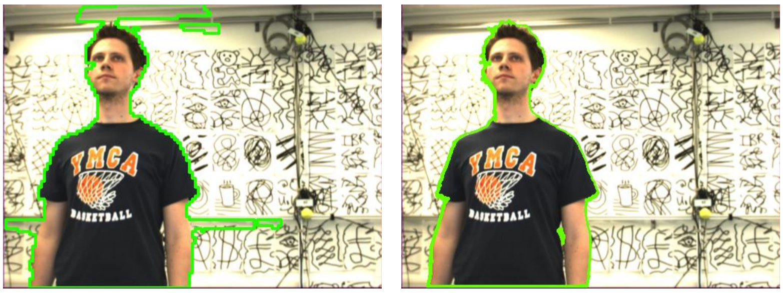
[8] Marc Levoy and Pat Hanrahan. Light field rendering. *Computer Graphics*, 30(Annual Conference Series):31–42, 1996.

[9] M. Li, M. Magnor, and H. Seidel. Improved hardware-accelerated visual hull rendering, 2003.

[10] Ming Li, Hartmut Schirmacher, Marcus Magnor, and Hans-Peter Seidel. Combining stereo and visual hull information for on-line reconstruction and rendering of dynamic scenes.

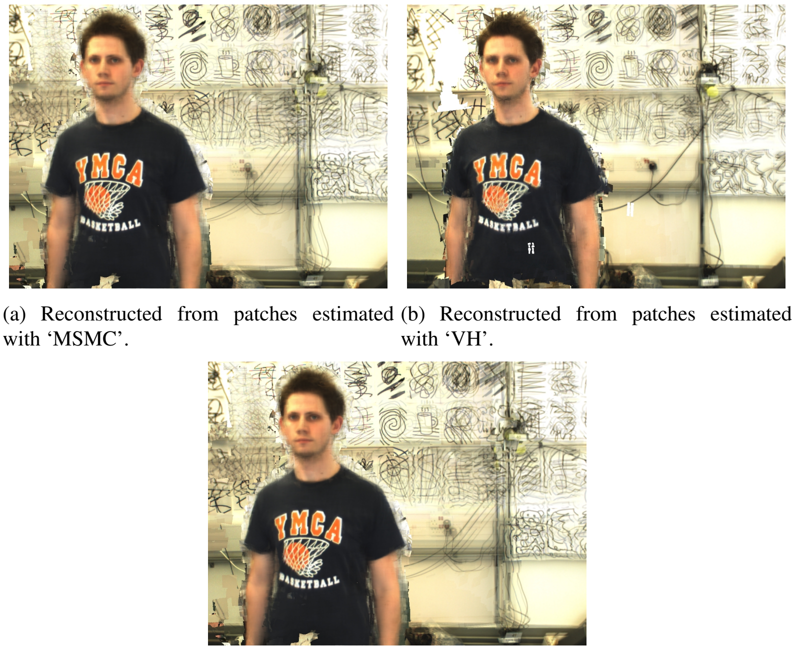
[11] Wojciech Matusik. Image-based visual hulls. In *Master of Science Thesis*, 2001.

[12] Niloy J. Mitra and An Nguyen. Estimating surface normals in noisy point cloud data. In *Proceedings of the nineteenth annual symposium on Computational geometry*, pages 322–328, 2003.



(a) Fixed foreground-background model. (b) Adaptive foreground-background model.

Figure 1: An input image with the border between foreground and background image overlayed for two different algorithms.



(a) Reconstructed from patches estimated with 'MSMC'.

(b) Reconstructed from patches estimated with 'VH'.

(c) Reconstructed from patches estimated with 'VH + MSMC'.

Figure 2: Reconstructions obtained after using various patch estimation algorithms.

[13] Andrew Mullins, Adam Bowen, Roland Wilson, and Nasir Rajpoot. Estimating planar patches for light field reconstruction. In *The Proceedings of the British Machine Vision Conference*, 2005.

[14] Andrew Mullins, Adam Bowen, Roland Wilson, and Nasir Rajpoot. Surace estimation and tracking using sequential mcmc methods for video based rendering. In *To be published in: The Proceedings of the International Conference on Image Processing ICIP07*, 2007.

[15] Don Murray and James J. Little. Patchlets: Representing stereo vision data with surface elements. In *Proceedings of the Seventh IEEE Workshops on Application of Computer Vision.*, volume 1, pages 192–199, 2005.

[16] P J Narayanan and Takeo Kanade. Virtual worlds using computer vision. In *Proceedings of the 1998 IEEE and ATR Workshop on Computer Vision for Virtual Reality Based Human Communications*, pages 2 – 13, January 1998.

[17] Hanspeter Pfister, Matthias Zwicker, Jeroen van Baar, and Markus Gross. Surfels: Surface elements as rendering primitives. In Kurt Akeley, editor, *Siggraph 2000, Computer Graphics Proceedings*, pages 335–342. ACM Press / ACM SIGGRAPH / Addison Wesley Longman, 2000.

[18] Tobias Pietzsch and Axel Gromann. A method of estimating oriented surface elements from stereo images. In *Proceedings of the British Machine Vision Conference*, volume 1, 2005.

[19] P. Ramanathan, E. Steinbach, P. Eisert, and B. Girod. Geometry refinement for light field compression.

[20] S. Seitz and C. Dyer. Photorealistic scene reconstruction by voxel coloring, 1997.

[21] Michael Spann and Roland Wilson. A quad-tree approach to image segmentation which combines statistical and spatial information. In *Pattern Recognition 18*, volume (3-4), pages 257–269, 1985.

[22] Tomáš Svoboda, Daniel Martinec, and Tomáš Pajdla. A convenient multi-camera self-calibration for virtual environments. *PRESENCE: Teleoperators and Virtual Environments*, 14(4):407–422, August 2005.

[23] Yang Wang, Kia-Fock Loe, Tele Tan, and Jian-Kang Wu. A dynamic hidden markov random field model for foreground and shadow segmentation. In *Proceedings of the Seventh IEEE Workshops on Application of Computer Vision*, volume 1, pages 474–480, 2005.

This work was supported by the UK EPSRC.