# Time Varying Volumetric Scene Reconstruction Using Scene Flow

Timothy Smith, David Redmill, Nishan Canagarajah, David Bull
Department of Electrical and Electronic Engineering,
University of Bristol, BS8 1UB, UK
timothy.smith@bristol.ac.uk

**Abstract**

Traditional volumetric scene reconstruction algorithms involve the evaluation of many millions of voxels which is highly time consuming. This paper presents an efficient algorithm based of future frame prediction that can dramatically reduce the number of voxels to be evaluated in time varying scenes. The new prediction method, combining scene flow and morphological dilations, is evaluated against a simple model dilation method. Results show the proposed method outperforms a simple dilation method and has the potential to improve the efficiency of volumetric scene reconstruction algorithms while retaining quality given accurate optical flows.

## 1 Introduction

Volumetric scene representations use a compact three dimensional grid to record colour and occupancy information at discrete points in space. Images taken from multiple calibrated cameras can be used to populate this voxel grid and many algorithms have been proposed [11, 7, 3, 16]. These algorithms have mainly been targeted at static scenes with their extension to video consisting of frame by frame processing. This results in useful temporal information being ignored which could otherwise aid the reconstruction process. This paper proposes a simple method of incorporating the observed optical flow from each camera into the reconstruction process with a view to dramatically reducing the number of voxels evaluated at each time frame. From per camera dense optical flows the per voxel scene flow is calculated and used to produce a volumetric frame prediction which is then dilated. This predicted model is then used to guide the voxel estimation of the next frame. An overview of voxel reconstruction algorithms is given in Section 2 with details of the proposed algorithm in Section 3. Section 4 shows results using the proposed technique and conclusions are drawn in Section 5.

## 2 Review of Volumetric Reconstruction

This section provides details of some of the main volumetric reconstruction algorithms relevant to this paper. A good overview of scene reconstruction techniques can be found in [12].

The voxel colouring algorithm was introduced in [11] as a method of constructing a set of voxels with associated colours from a set of calibrated colour images. If deemed

BMVC 2007 doi:10.5244/C.21.81

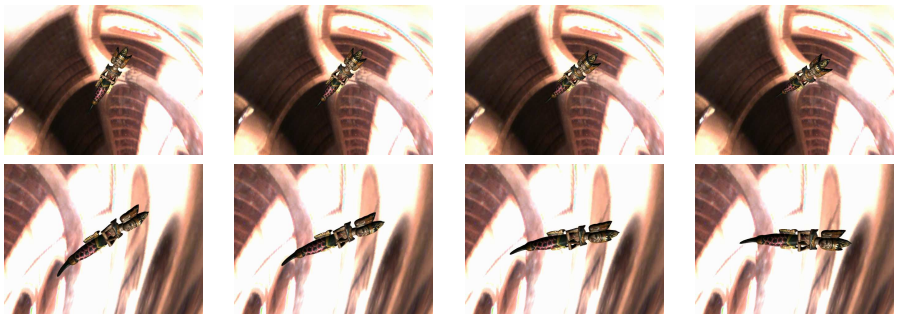


Figure 1: Original synthetic frames from 2 cameras. Left to right: Frame 1, 7, 13 and 19.

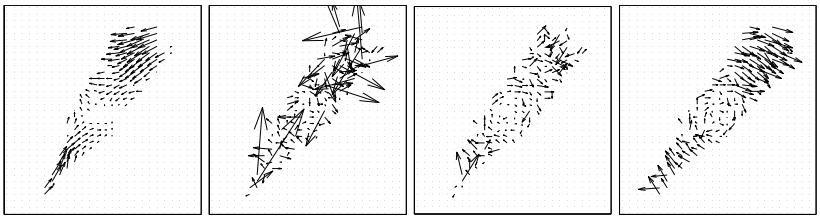
to be colour consistent across the images, a voxel is marked as occupied and assigned an average colour otherwise it is marked as transparent. Voxel occlusions are handled by restricting camera placement to satisfy the *ordinal visibility constraint*.

Space carving is described in [7] as a generalization of voxel colouring. In [7] it is shown that by starting with an overcomplete voxel representation of a scene then removing non-photoconsistent voxels, a *photo hull* can be produced. The space carving algorithm does not impose restrictions on camera placement by using a multi-sweep algorithm. Generalized voxel colouring [3] and multi-hypothesis reconstruction [4] attempt to solve the voxel visibility problem using slightly different techniques. In [3] each voxel is carved based on its simultaneous photoconsistency in all camera views in which it is visible while in [4] each voxel is assigned a number of colour hypotheses which are gradually removed if inconsistent. While space carving-type algorithms [4, 3, 7] are more general than voxel colouring [11] they still suffer from many of the same reconstruction artifacts, notably fattened reconstructions where surfaces bulge out towards the cameras.

Basing the voxel occupancy decision on a local threshold generally results in a non-optimal solution being reached. Rather, a globally optimum solution should be found. Vogiatzis et al. [16] extract a photoconsistent object surface using a minimum cut solution of a weighted graph representation of a photoconsistency cost function. In [6] this graph cut algorithm is enhanced by allowing adjacent voxels to contribute to the photo consistency function and implicitly providing a smoothing term. This technique is currently limited to closed, watertight objects.

A number of techniques to bring voxel colouring closer to real-time performance are suggested in [10]. Using temporal coherence to speed up voxel colouring of dynamic scenes is suggested with a simple extension that takes the previous frame in a sequence, dilates the occupied voxel set from that frame and then uses this set as the search space of the current frame. While a speed up of around two is shown, the method is unsuitable for fast moving scenes as no explicit motion parameters are calculated.

Some work has also been done with regard to modelling moving scenes using voxels. In [15] a six dimensional voxel representation of a scene is proposed where voxels are carved if they are inconsistent with images from two time instants or inconsistent with the flow between the two times. This method also recovers the scene flow between frames. The scene flow [14] can also be derived from per camera 2D optical flows which in [13] is applied to interpolating between two already carved scene frames. Rather than using two known scenes at consecutive times, this paper takes scene flow and uses it to project a known scene forward in time to guide the estimation of the following scene frame.



(a) H264 (b) Horn-Schunck (c) non-iterative L-K (d) hierarchical L-K

Figure 2: Optical flows for synthetic sequence, frame 1.

# 3 Methodology

A volumetric reconstruction method for moving scenes is proposed which consists of three steps: voxel occupancy estimation, voxel scene flow estimation and voxel occupancy prediction. A volumetric representation of the first frame in the sequence is estimated by evaluating all voxels in the scene volume. This gives an occupancy and colour for each voxel. The scene flow is then calculated from per camera optical flows and, combined with a dilation step, used to predict the occupancy in the next frame. When subsequent frames are processed only voxels which have been predicted as occupied from the previous frame are evaluated. More details are given below.

## 3.1 Voxel Occupancy Estimation

The voxel occupancy step can be any algorithm that takes calibrated input images and produces a colour and occupancy for each voxel in the scene, such as voxel colouring [11], voxel carving [3, 4, 7] or volumetric graph-cuts [16]. In the rest of this paper, voxel colouring is used for the voxel occupancy estimation due to its simplicity.

## 3.2 Optical Flow

Video sequences are often analyzed using optical flow. A very brief overview of the optical flow estimation techniques used in this paper is given below. More details can be found in the original papers [5, 8, 2] and a performance analysis of techniques in [1]. If optical flow is thought of as a simple translation, $\mathbf{v} = (\frac{\partial u}{\partial t}, \frac{\partial v}{\partial t})^T$, and intensity is assumed to be conserved then the gradient constraint equation may be written

$$\nabla I(\mathbf{x},t) \cdot \mathbf{v} + I_t(\mathbf{x},t) = 0 \tag{1}$$

where $I_t(\mathbf{x},t) = \frac{\partial I(\mathbf{x},t)}{\partial t}$. A second constraint [1] must be used to solve this equation.

Horn and Schunck [5] use a global smoothness term to provide this extra constraint and seek to iteratively minimize

$$E = \int_D (\nabla I \cdot \mathbf{v} + I_t)^2 + \lambda^2 (\|\nabla u\|^2 + \|\nabla v\|^2) d\mathbf{x} \tag{2}$$

Lucas and Kanade [8] assume optical flows are constant in a small neighbourhood and seek to minimize the error function

$$E = \sum_{\mathbf{x} \in \Omega} W^2(\mathbf{x}) \left[\nabla I(\mathbf{x},t) \cdot \mathbf{v} + I_t(\mathbf{x},t)\right]^2 \tag{3}$$

where $W^2(\mathbf{x})$ is a window function with decreasing weights from its centre. An iterative scheme may be applied where the source image is warped towards the target image after each minimization step using the current estimate of the optical flow. The estimated optical flow between the target image and warped source image is then computed and added into the overall optical flow.

Both of these gradient based techniques assume small (less than a pixel) optical flows. To combat this limitation a hierarchical scheme may be used [2]. A Gaussian pyramid is constructed from two temporally adjacent original images and the optical flow estimation run on the lowest resolution images in each of the pyramids. This flow information is then propagated to the next highest resolution to form the starting point for that resolution. This allows a straightforward integration into the iterative Lucas-Kanade algorithm.

Optical flow may also be solved directly using block matching techniques which are commonly performed by searching in $\mathbf{v}$ to minimize the sum of squared differences between the source and target block. The reader is referred to [1] for a fuller discussion. Such block matching algorithms are often found in motion based video compression schemes such as MPEG-2 and H264 [9] with the motion vectors embedded into the compressed video stream.

## 3.3 Scene Flow

In the scene flow estimation step the motion of each voxel is represented using scene flow as introduced in [14] as a 3D extension of optical flow in 2D. Let $\mathbf{x}(t) = (x, y, z)$ be the position of a 3D scene point (voxel centre) at time $t$ and $\mathbf{u}_n(t) = (u_n, v_n)$ be its projection in image $I_n$ then

$$\frac{d\mathbf{u}_n}{dt} = \frac{\partial \mathbf{u}_n}{\partial \mathbf{x}} \frac{d\mathbf{x}}{dt} \tag{4}$$

where $\frac{d\mathbf{u}_n}{dt}$ is the optical flow in image $n$ and $\frac{d\mathbf{x}}{dt}$ is the instantaneous scene flow. A system of equations $\mathbf{B}\frac{d\mathbf{x}_j}{dt} = \mathbf{A}$ can be set up with $N \geq 2$ cameras where

$$\mathbf{B} = \begin{bmatrix} \frac{\partial u_1}{\partial x} & \frac{\partial u_1}{\partial y} & \frac{\partial u_1}{\partial z} \\ \frac{\partial v_1}{\partial x} & \frac{\partial v_1}{\partial y} & \frac{\partial v_1}{\partial z} \\ \vdots & \vdots & \vdots \\ \frac{\partial u_N}{\partial x} & \frac{\partial u_N}{\partial y} & \frac{\partial u_N}{\partial z} \\ \frac{\partial v_N}{\partial x} & \frac{\partial v_N}{\partial y} & \frac{\partial v_N}{\partial z} \end{bmatrix}, \mathbf{A} = \begin{bmatrix} \frac{\partial u_1}{\partial t} \\ \frac{\partial v_1}{\partial t} \\ \vdots \\ \frac{\partial u_N}{\partial t} \\ \frac{\partial v_N}{\partial t} \end{bmatrix} \tag{5}$$

By taking the singular value decomposition of $\mathbf{B}$ such that $\mathbf{B} = \mathbf{U}.\mathbf{w}.\mathbf{V}^T$, a solution can be found for the $j^{th}$ voxel as $\frac{d\mathbf{x_j}}{dt} = \mathbf{V}.diag(\frac{1}{w_i}).\mathbf{U}^T.\mathbf{A}$. This solution minimizes the squared error between the reprojected scene flow and the optical flow in each camera image. If more cameras are used, a more robust scene flow estimation can be calculated. $\mathbf{B}$ is calculated from the camera projection matrix at $\mathbf{x}(t)$. The scene flow is calculated for each voxel marked as occupied to create a 3D volumetric model that includes per voxel motion.

| Voxel evaluation method | Average PSNR (dB) |
|---|---|
| All voxels | 14.0272 |
| 4 dilations only | 13.7315 |
| Hierarchical Lucas-Kande + 3 dilations | 13.6506 |
| Lucas-Kanade + 3 dilations | 13.5858 |
| H264 + 3 dilations | 13.2917 |
| Horn-Schunck + 3 dilations | 13.2564 |
| 3 dilations only | 12.3454 |

Table 1: Average PSNR for different frame prediction methods for synthetic sequence.

### 3.4 Voxel Occupancy Predication

The next step is to predict the next scene frame based on the current frame. To do this each voxel in the current scene frame is moved based on the scene flow vector assigned to it with voxels which would be moved out of the volume being clamped to lie on the edge of the volume. Using scene flow on its own is not sufficient for successful prediction therefore this paper proposes that the predicted model is then expanded using a 3D version of the morphological dilation operator which dilates based on each voxel's six-face-connected binary occupancies. There are three motivations to doing this:

- Each predicted voxel is forced to lie on integer voxel coordinates whereas in reality it lies between integer voxels and has an influence on the surrounding voxels.
- An unknown error is associated with each scene flow vector leading to voxels possibly being moved incorrectly.
- The forward flowed voxel model may have holes which would affect subsequent reconstructions as voxels are only removed, never added, during voxel carving.

The number of dilations is found empirically based on the granularity of the voxel model compared with the input images and the error associated with the optical flow field. With a fine voxel model, a number of dilations may correspond to a single pixel change in the input images meaning more dilations are needed.

## 4 Results and Discussion

For the evaluation of the proposed technique a synthetic 20 frame sequence[1] (Figure 1) and 20 frame natural sequence[2] (Figure 7a) from 8 fully calibrated cameras were used.

In the synthetic sequence the highly textured figure rotates with its extremities moving at 5 to 8 pixels per frame while its centre of rotation remains fixed. Optical flows were recovered for every frame in every camera using three methods: Horn-Schunck, non-iterative Lucas-Kanade (as in [1]) and hierarchical iterative Lucas-Kanade. In addition, the block motion vectors from an H264 encoding of each camera sequence found using the exhaustive search strategy [9] were extracted and upsampled to produce an H264 optical flow. For the spatial image derivatives a 5 point central difference kernel was used

[1]Original 3D model Copyright ©Andrew Kator, http://www.katorlegaz.com/

[2]Dataset from Interactive Visual Media Group, Microsoft Research [17]

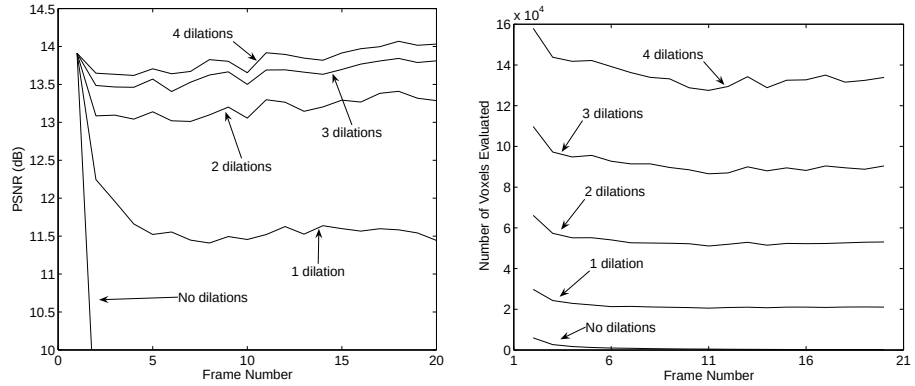



Figure 3: PSNR and voxel count using proposed technique on synthetic sequence with hierarchical Lucas-Kanade optical flow and a varying number of dilations.

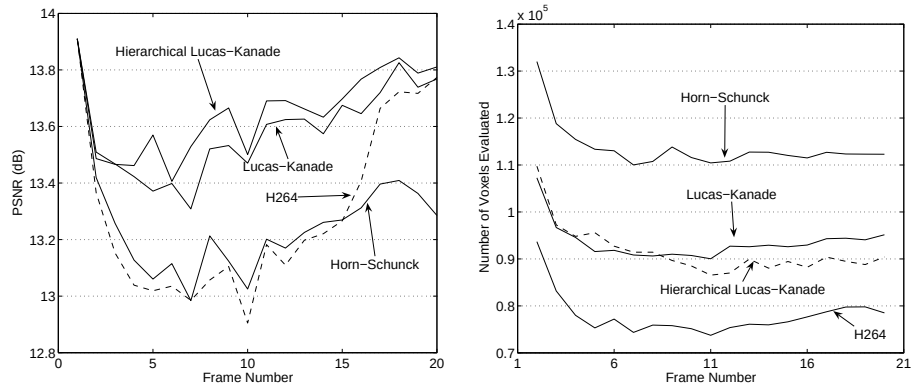



Figure 4: PSNR and voxel count using proposed technique on synthetic sequence with different optical flow estimation algorithms and 3 dilations.

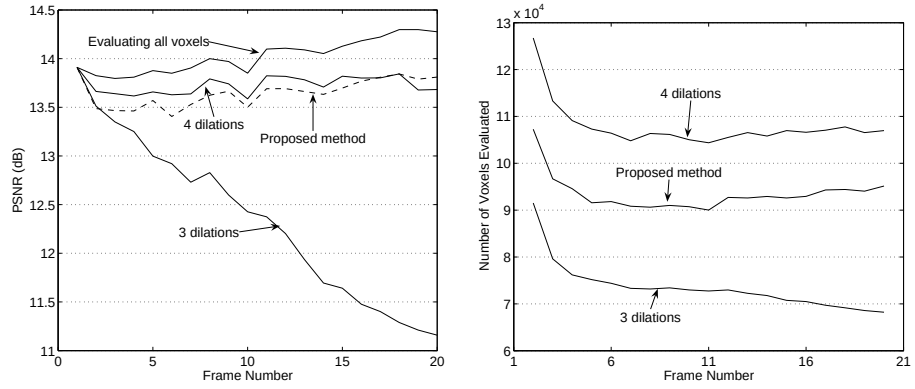
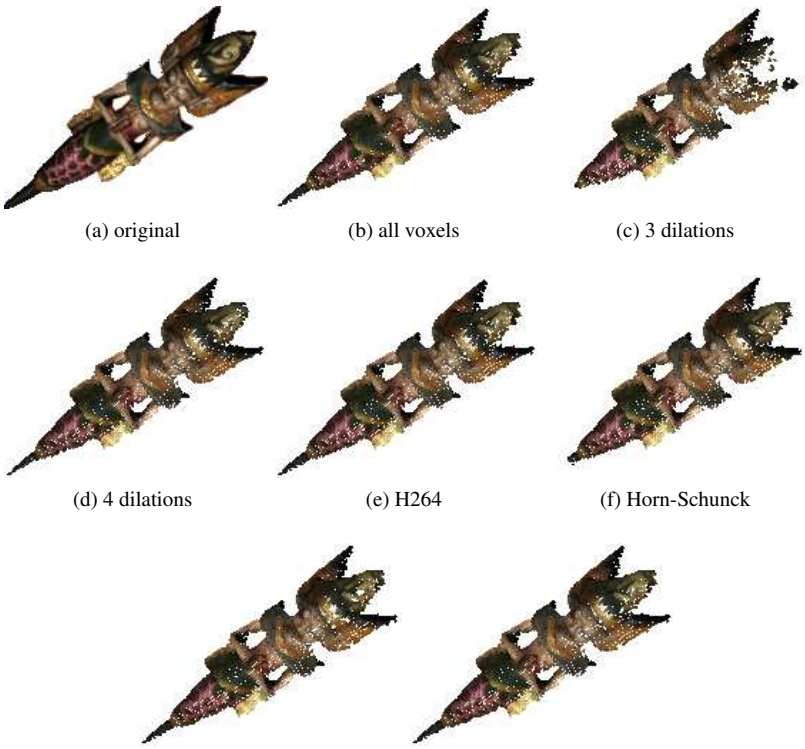



Figure 5: PSNR and voxel count comparing only using dilations, evaluating all voxels and using proposed technique (Lucas-Kanade plus 3 dilations) for synthetic sequence.



(a) original (b) all voxels (c) 3 dilations

(d) 4 dilations (e) H264 (f) Horn-Schunck

(g) Lucas-Kanade (h) hierarchical Lucas-Kanade

Figure 6: Frame 19 reconstructions for synthetic sequence.

(as in [1]) while the temporal derivative was calculated from a simple frame difference. The original images were not pre-smoothed. Example flows for each method for frame 1 are shown in Figure 2. As expected the hierarchical iterative Lucas-Kanade (Figure 2d) retrieves the most accurate optical flows with large motions correctly recovered. All the optical flow algorithms struggled to obtain accurate flows for the 'tail' of the figure which is near homogeneous in colour and in many images is the same colour as the background.

These four sets of optical flows were then used in the proposed reconstruction algorithm using 3 dilations, chosen to give a balance between scene reconstruction quality and voxel count. Figure 3 shows the sensitivity of the proposed algorithm to varying the number of dilations. To assess the quality of the estimated model each frame was reconstructed for each camera and a peak signal-to-noise ratio (PSNR) calculated between the reconstruction and the original camera frame image. This PSNR was calculated over the actual image area of the original 3D model based on a segmentation of the background and foreground generated when the original scene was being rendered. The mean frame PSNR over all cameras and the number of voxels evaluated at each frame, excluding frame 1, are shown in Figure 4. Frame 1 has $256^3 \approx 10^7$ voxels evaluated.

It is clear that the best performance is achieved using the hierarchical iterative Lucas-Kanade optical flows. Notably there is a sharp drop in PSNR after the first frame indicating the shift away from evaluating all voxels to evaluating only predicted voxels.

(a) original (b) all voxels evaluated (c) proposed method

Figure 7: Frame 8 reconstructions for natural sequence

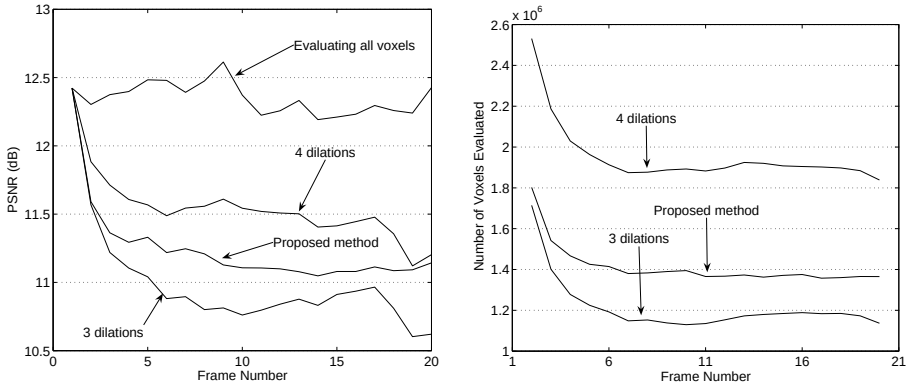



Figure 8: PSNR and voxel count comparing only using dilations, evaluating all voxels and using proposed technique (Lucas-Kanade plus 3 dilations) for natural sequence.

Interestingly, the H264 PSNR continues to drop until frame 10 whereupon it starts to rise again. This can be explained by looking at the optical flows estimated for each frame from the block motion vectors and noting that for the first 10 frames the direction of movement at the head of the figure is incorrect[3]. In the later frames this block is estimated correctly. Despite causing the most voxels to be evaluated (Figure 4), the Horn-Schunck optical flows produce the worst reconstruction results due to the badly estimated optical flow in areas with large pixel displacements (Figure 2b).

Taking the best performing optical flow (hierarchical Lucas-Kanade) allows a comparison to be made to a simple dilation method [10] as well as to evaluating all voxels in the scene. As the motion predicted model based on scene flow is dilated three times before being used, it is important to establish that the same effect could not be achieved simply using 3 dilations alone. Figure 5 clearly shows that 3 dilations is insufficient to track the object whereas incorporating optical flow brings the PSNR back up to a stable state. Related to this PSNR drop is the drop in the number of voxels (Figure 5) being evaluated meaning that voxels are continuously being lost from the constant volume object. Increasing the number of dilations to 4 brings the PSNR to a value similar to that obtained using scene flow at the expense of an increase in the number of voxels evaluated. The dilations must expand the previous model enough to capture the largest motion in the scene meaning a scene with large motions needs more dilations to capture the voxels associated

[3]Incorrect in terms of optical flow. The H264 estimated motion is actually that which minimizes the coding cost and is most probably correct in this sense.

with these motions. Using scene flow to guide the prediction allows large motions to be present in the scene and still keep the number of voxels evaluated to a minimum.

Evaluating all voxels in a scene produces the best quality reconstruction but using the proposed scene flow guided prediction model as a hypothesis for the next scene frame leads to only a small drop in quality with a dramatic reduction in voxel evaluations performed. As reconstruction time is directly proportional to the number of voxels evaluated and the optical flow calculations are relatively fast, a significant processing time decrease is achieved. A summary of the overall average frame PSNRs is shown in Table 1 while Figure 6 shows the synthesis results for frame 19 from a single camera. The low overall PSNR could be improved by using a more sophisticated voxel occupancy algorithm [16].

Results for the natural sequence are shown if Figures 7 and 8 with PSNR evaluated over the entire image. The extremities of the dancer move upto 80 pixels between frames leading to very poor optical flow estimation for these areas which leads to the observable poor synthesis in these regions (Figure 7c). Even so, including optical flow still produces higher quality syntheses than using only 3 dilations and is comparable to the synthesis quality obtained when evaluating all voxels.

# 5 Conclusions and Future Work

Volumetric scene reconstruction algorithms usually focus on static scenes and do not take into account temporal information when reconstructions are performed on moving scenes. To address this weakness, this paper has suggested using a combination of scene flow and morphological dilations applied to a standard voxel colouring algorithm. A number of optical flow algorithms [5, 8, 2] have been used to obtain dense scene flow which has then been applied to the prediction of future scene frames. Basing voxel occupancy estimation on the predicted occupancy allows a dramatic decrease in the number of voxels which need to be evaluated leading to a substantial computational speed gain with only a small decrease in reconstruction quality. The proposed technique also improves on previous model dilation techniques [10].

At present, evaluating all voxels in the scene for photoconsistency produces the highest quality reconstructions. In the future, techniques for increasing the reconstruction quality based on scene flow will be explored, such as dynamically varying the number of dilations based on optical flow confidences. The same scene flow based algorithm will also be integrated with more advanced voxel occupancy estimation algorithms.

# References


[1] J. L. Barron, D. J. Fleet, and S. S. Beauchemin. Performance of Optical Flow Techniques. *International Journal of Computer Vision*, 12(1):43–77, 1994.

[2] James R. Bergen, P. Anandan, Keith J. Hanna, and Rajesh Hingorani. Hierarchical Model-Based Motion Estimation. In *Proceedings of the European Conference on Computer Vision*, volume LNCS 588, pages 237–252, 1992.

[3] W. Bruce Culbertson, Thomas Malzbender, and Gregory G. Slabaugh. Generalized Voxel Coloring. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 100–115, 2000.

[4] P. Eisert, E. Steinbach, and B. Girod. Multi-Hypothesis, Volumetric Reconstruction of 3-D Objects from Multiple Calibrated Camera Views. In *International Conference on Acoustics Speech and Signal Processing*, pages 3509–3512, March 1999.

[5] Berthold K. P. Horn and Brian G. Schunck. Determining Optical Flow. *Artificial Intelligence*, 17:185–203, 1981.

[6] Alexander Hornung and Leif Kobbelt. Robust and Efficient Photo-Consistency Estimation for Volumetric 3D Reconstruction. In *Proceedings of the European Conference on Computer Vision*, volume LNCS 3952, pages 179–190, May 2006.

[7] Kiriakos N. Kutulakos and Steven M. Seitz. A Theory of Shape by Space Carving. *International Journal of Computer Vision*, 38(3):199–218, 2000.

[8] Bruce D. Lucas and Takeo Kanade. An Iterative Image Registration Technique with an Application to Stereo Vision. In *Proceedings of the 7th International Joint Conference on Artificial Intelligence (IJCAI '81)*, pages 674–679, April 1981.

[9] J. Ostermann, J. Bormans, P. List, D. Marpe, M. Narroschke, F. Pereira, T. Stockhammer, and T. Wedi. Video coding with H.264/AVC: Tools, Performance, and Complexity. *IEEE Circuits and Systems Magazine*, 4(1):7–28, 2004.

[10] C. Prock and A. Dyer. Towards Real-Time Voxel Coloring. In *DARPA Image Understanding Workshop*, pages 315–321, 1998.

[11] S. M. Seitz and C. R. Dyer. Photorealistic Scene Reconstruction by Voxel Coloring. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1067–1073, June 1997.

[12] G. Slabaugh, B. Culbertson, T. Malzbender, and R. Schafer. A Survey of Methods for Volumetric Scene Reconstruction from Photographs. In *International Workshop on Volume Graphics*, pages 81–100, June 2001.

[13] Sundar Vedula, Simon Baker, and Takeo Kanade. Image-Based Spatio-Temporal Modeling and View Interpolation of Dynamic Events. *ACM Transactions on Graphics*, 24(2):240–261, April 2005.

[14] Sundar Vedula, Simon Baker, Peter Rander, Robert Collins, and Takeo Kanade. Three-Dimensional Scene Flow. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(3):475–480, March 2005.

[15] Sundar Vedula, Simon Baker, Steven Seitz, and Takeo Kanade. Shape and Motion Carving in 6D. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 592–598, June 2000.

[16] G. Vogiatzis, P.H.S. Torr, and R. Cipolla. Multi-View Stereo via Volumetric Graph-Cuts. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 391–398, June 2005.

[17] C. Lawrence Zitnick, Sing Bing Kang, Matthew Uyttendaele, Simon Winder, and Richard Szeliski. High-quality video view interpolation using a layered representation. *ACM Transactions on Graphics*, 23(3):600–608, 2004.