On New View Synthesis Using Multiview Stereo

O. J. Woodford[1], I. D. Reid[1], P. H. S. Torr[2] and A. W. Fitzgibbon[3]
[1]Department of Engineering Science, University of Oxford
[2]Department of Computing, Oxford Brookes University
[3]Microsoft Research, Cambridge

Abstract

We show that application of modern multiview stereo techniques to the new-view synthesis (NVS) problem introduces a number of non-trivial complexities. By simultaneously solving for the colour and depth of the new-view pixels we can eliminate the visual artefacts that conventional NVS-via-stereo suffers. The global occlusion reasoning which has led to considerable improvements in recent stereo algorithms can easily be included in the new algorithm, using a recently improved graph-cut-based optimizer for general multi-label conditional random fields (CRFs). However, the CRF priors that are important to success in stereo cannot be easily applied if the reconstruction is to be computed in the reference frame of the novel view. We address this problem by extending recent work on the fast optimization of texture priors in NVS to model the image edge structure, yielding a synthesis of the two approaches which yields good results on difficult image sequences.

1 Introduction

The problem addressed in this paper is new view synthesis (NVS): given multiple images of a 3D scene captured by a set of cameras, or by a single moving camera, generate a synthetic view of the scene, as it would appear from a new viewpoint. Such new views can be used in teleconferencing [1] or in 3-dimensionalizing monocular film footage.

Algorithms to solve this problem can be subdivided into two categories: scene reconstruction, and image-based rendering. Reconstruction methods form a representation of the 3D scene, for example as a 3D depth map [9], volumetric grid [11] or plenoptic function [7], from which the new view can be rendered. Stereo methods in particular can produce extremely accurate reconstructions, with only sparse input images, as occlusion between pixels is explicitly modelled [5, 12], and the smoothness prior can encourage depth discontinuities in the reconstruction to coincide with intensity edges in the input images—a conditional random field (CRF) prior [3]. However, the considerable, and universal, disadvantage of these methods is the generation of artefacts when the new view is finally rendered, such as "tearing" [9], distortion of fine features, and general aliasing caused by the change in reference frame.

In contrast, image-based rendering (IBR) methods solve directly for colour in the *new* view, thus avoiding these pitfalls. IBR methods can be further categorized into implicit and explicit geometry methods. Of these, implicit geometry methods [2, 13] marginalize out the depth, solving only for the colour of new-view pixels. Such methods generally employ image-based priors, working well on fine scene features. Explicit geometry methods [10] generate a depth map for the new view, much in the same way as traditional stereo

methods, but the important link between input image edges and depth discontinuities provided by the CRF is lost. IBR methods have also accounted for occlusion between pixels using occlusion models based on robust statistics [10, 13] rather than geometry, so do not enjoy the global occlusion reasoning of the stereo methods.

In this paper we combine these strands of stereo reconstruction and IBR. We take a recently introduced stereo algorithm [12] and adapt it to the NVS domain, requiring that a number of nontrivial problems be addressed. The primary contributions are (1) simultaneously solving for the new view and depth, with occlusion modelling, and (2) replacing the CRF with an efficient texture prior [13]. While stereo literature sometimes alludes to its potential application in NVS, the conversion process and the challenges it produces have not been addressed until now. This is, to our knowledge, the first IBR method to use a geometrical occlusion model in a global optimization framework, and is certainly the first to combine this with a texture term.

The paper proceeds in these stages: formal statement of the problem; definition of the energy function to be minimized; description of the graph-cut based optimization strategy; and evaluation of the results.

2 Problem statement

The task of NVS is to generate a new view, $\mathcal{V}$, of a scene, given a set of calibrated input views, $\mathcal{I}_1, .., \mathcal{I}_N$. A 2D vector, $\mathbf{x}$, denotes a pixel location in $\mathcal{V}$, the colour of which is written as $V(\mathbf{x})$. A projection function $\pi_i(\mathbf{x}, z)$ computes the 2D projection in image i of the 3D point at depth z in front of pixel $\mathbf{x}$ in the novel view. This function is easily computed from the images using commercial camera calibration software. The colour of this pixel projected into image $\mathcal{I}_i$ is written $I_i(\mathbf{x}, z)$, shorthand for $I_i\left(\pi_i(\mathbf{x}, Z(\mathbf{x}))\right)$, with $Z(\mathbf{x})$ (and z) being the estimated depth of the pixel. Pixel colours at non-integer locations are linearly interpolated from the image; locations outside the image boundaries are given a value of ∞.

The problem is poorly constrained—many candidate solutions $\mathcal{V}$ can explain the data equally well—so a powerful prior is needed to select good solutions. Following many current NVS [10] and stereo [5, 12] approaches, we cast our problem in a CRF energy minimization framework explicitly over depth (as well as colour), in contrast to methods which marginalize out depth, optimizing solely over colour [2, 13]. Our objective function contains costs over pixels and cliques of pixels, of the form

$$E(\mathcal{V}, \mathcal{Z}) = \underbrace{E_{\text{photo}}(\mathcal{V}, \mathcal{Z})}_{\text{data costs}} + \underbrace{E_{\text{smooth}}(\mathcal{V}, \mathcal{Z})}_{\text{surface smoothness}}, \tag{1}$$

for which we can use a powerful global optimizer based on graph cuts to compute strong local optima of the energy.

2.1 Data costs

The data cost is a term that ensures that each pixel in $\mathcal{V}$ is photo consistent with the input views. It enforces the constraint that the colour of output pixels which are visible (not occluded) in a given input view should match the colour of their projected location in that

view. We use a standard truncated SSD data cost. E_{photo} is the sum of data costs over all pixels in the novel view, denoted by the set $\mathcal{X}$, averaged over input views, thus:

$$E_{\text{photo}}(\mathcal{V},\mathcal{Z}) = \frac{1}{N}\sum_{i=1}^{N}\sum_{\mathbf{x}\in\mathcal{X}} O_i(\mathbf{x},\mathcal{Z})\min(\|V(\mathbf{x})-I_i(\mathbf{x},z)\|^2,\kappa)+(1-O_i(\mathbf{x},\mathcal{Z}))\nu \quad (2)$$

where κ is a robustness threshold, ν is a penalty cost for occluded pixels, and $O_i(\mathbf{x},\mathcal{Z})$ indicates whether pixel $\mathbf{x}$ is occluded in $\mathcal{I}_i$; 1 means visible, 0 means occluded. We must have $\nu > \kappa$ in order to avoid our objective function encouraging self-occlusions.

We use the asymmetrical occlusion model of Wei and Quan [12] to evaluate the visibility of pixels—the value of $O_i(\mathbf{x},\mathcal{Z})$ is determined entirely from our single depth map, $\mathcal{Z}$. It is defined to be 0 if there exists another pixel, $\mathbf{p}$, which projects to the same point[1] in $\mathcal{I}_i$ as pixel $\mathbf{x}$, and for which the projected depth is less than that of $\mathbf{x}$, otherwise it is 1.

2.2 Surface smoothness

Surface smoothness priors regularize out uncertainties in depth, especially in untextured regions, by placing a cost, $S()$, on a neighbourhood, $\mathcal{N}$, of pixels, which encourages smoothness. E_{smooth} is the sum of smoothness costs over a defined set of pixel neighbourhoods, $\mathbb{N}$, commonly defined as:

$$E_{\text{smooth}}(\mathcal{Z}) = \sum_{\mathcal{N}\in\mathbb{N}} \lambda_{\text{s}}\min\Big(S(\mathcal{N},\mathcal{Z}),\delta_{\text{s}}\Big) \quad (3)$$

where λ_{s} weights the smoothness prior, and δ_{s} is a discontinuity preserving threshold. This is a truncated linear kernel, which approaches the Potts model kernel as $\delta_{\text{s}} \to 0$.

Stereo methods in this graph cut optimized framework generally use, as a smoothness cost, a prior on the first order of disparity of two pixel neighbourhoods:

$$S(\{\mathbf{p},\mathbf{q}\},\mathcal{Z}) = \left|\frac{1}{Z(\mathbf{p})} - \frac{1}{Z(\mathbf{q})}\right|. \quad (4)$$

Many stereo methods locally vary λ_{s} and/or δ_{s} as a function of the reference image, in order to encourage occlusion boundaries to fit to image contours. Since, in NVS, the reference image (the new view, $\mathcal{V}$) is unknown, this approach is not possible here. However, Woodford *et al*. [13] recently introduced a pairwise texture prior which discourages discontinuities only where there is no supporting evidence from the input sequence. We therefore define a new E_{smooth} which incorporates this prior, thus:

$$E_{\text{smooth}}(\mathcal{V},\mathcal{Z}) = \sum_{\mathcal{N}\in\mathbb{N}} E_{\text{texture}}(\mathcal{V},\mathcal{N})\lambda_{\text{s}}\min\Big(S(\mathcal{N},\mathcal{Z}),\delta_{\text{s}}\Big), \quad (5)$$

$$E_{\text{texture}}(\mathcal{V},\mathcal{N}) = 1+\lambda_{\text{t}}\min\Big(\min_{\mathbf{T}\in\mathbb{T}_{\mathcal{N}}}\|\mathbf{T}-\mathbf{V}(\mathcal{N},\mathcal{Z})\|^2,\delta_{\text{t}}\Big) \quad (6)$$

where $\mathbf{V}(\mathcal{N},\mathcal{Z})$ represents the vector of colours of the pixels in $\mathcal{N}$, defined by $\{V(x,\mathcal{Z})|x \in \mathcal{N}\}$, $\mathbb{T}_{\mathcal{N}}$ represents a library of patches specific to the $\mathcal{N}$, constructed as described in [13], and λ_{t} and δ_{t} are a further two model parameters.

[1] We define 'same point' to mean within half a pixel in both directions. This measure is an approximation, as different pixels have different projected footprints. While a more accurate definition could be employed, we found ours to work suitably well.

2.3 Computing colour

NVS differs from stereo in that one is optimizing over both colour and depth, as opposed to just depth. However, by making colour a function of depth, we can reuse the stereo optimization framework. We define the colour of pixel $\mathbf{x}$ to be the mean of visible input image samples of $\mathbf{x}$, thus:

$$V(\mathbf{x}, \mathcal{Z}) = \frac{\sum_{i=1}^{N} O_i(\mathbf{x}, \mathcal{Z}) I_i(\mathbf{x}, z)}{\sum_{i=1}^{N} O_i(\mathbf{x}, \mathcal{Z})}. \tag{7}$$

While the truncation term, κ, means that equation (2) is not necessarily unimodal in $\mathcal{V}$, given $\mathcal{Z}$, if we assume that all visible samples are a good match (as they should be for the correct solution), then equation (7) gives the colour that minimizes the E_{photo} term. Therefore, we can rewrite all the above energies in terms of $\mathcal{Z}$ only, and, by discretizing depth, we can now optimize this energy using a recently introduced method to obtain high-quality solutions, as we now describe.

3 Optimization

Despite the apparent complexity of the energy in fig 1, it ultimately boils down to an energy of the form

$$E(\mathcal{Z}) = \underbrace{\sum_{\mathbf{x} \in \mathcal{X}} u_{\mathbf{x}}(Z(\mathbf{x}))}_{\text{unary terms}} + \underbrace{\sum_{\mathcal{N} \in \mathbb{N}} c_{\mathcal{N}}(Z(\mathcal{N}))}_{\text{clique terms}} \tag{8}$$

where the cliques include 2-cliques of pixels which may be a long way apart, defining the occlusion term O (for which the reader is referred to [12]). A recent study of optimization algorithms [4] showed that such long range and irregularly connected terms are only effectively optimized using graph cut algorithms.

In order to optimize the energy, we therefore follow recent work [8], and reduce it to a sequence of binary problems as follows. Suppose we have a current estimate of the depth, $\mathcal{Z}_t$, and a *proposal* depth map $\mathcal{Z}_p$. The goal is to optimally combine ("fuse") the proposal and current depth maps to generate a new depth map $\mathcal{Z}_{t+1}$ for which the energy $E(\mathcal{Z}_{t+1})$ is lower than $\mathcal{Z}_t$. This is achieved by taking each pixel in $\mathcal{Z}_{t+1}$ from one of $\mathcal{Z}_t, \mathcal{Z}_p$, as controlled by a binary indicator image $\mathcal{B}$ with elements $B(\mathbf{x})$:

$$\mathcal{Z}(\mathcal{B}) = \mathcal{B} \cdot \mathcal{Z}_t + (1 - \mathcal{B}) \cdot \mathcal{Z}_p, \tag{9}$$

where dot indicates elementwise multiplication. Then the energy $E(\mathcal{Z})$ is a function only of the indicator image $\mathcal{B}$, so we may define

$$\mathcal{Z}_{t+1} = \mathcal{Z}\left(\operatorname*{argmin}_{\mathcal{B}} E(\mathcal{B} \cdot \mathcal{Z}_t + (1 - \mathcal{B}) \cdot \mathcal{Z}_p) \right). \tag{10}$$

If this binary optimization problem leads to a submodular[2] graph then a globally optimal $\mathcal{B}$ can be found using graph cuts. However, as Wei and Quan [12] explain, the occlusion term O is not guaranteed to fulfil the submodularity constraint.

[2] A submodular pairwise energy graph is one for which every pairwise energy term, $\phi_{pq}(l_p, l_q)$, $l_p, l_q \in \{0, 1\}$, satisfies the submodularity constraint: $\phi_{pq}(0,0) + \phi_{pq}(1,1) \leq \phi_{pq}(0,1) + \phi_{pq}(1,0)$.

$$
\begin{aligned}
V(\mathbf{x}, \mathcal{Z}) &:= \frac{\sum_{i=1}^{N} O_i(\mathbf{x}, \mathcal{Z}) I_i(\mathbf{x}, z)}{\sum_{i=1}^{N} O_i(\mathbf{x}, \mathcal{Z})} \\
E_{\text{photo}}(\mathcal{Z}) &:= \frac{1}{N} \sum_{i=1}^{N} \sum_{\mathbf{x} \in \mathcal{X}} \begin{pmatrix} O_i(\mathbf{x}, \mathcal{Z}) \min(\|V(\mathbf{x}, \mathcal{Z}) - I_i(\mathbf{x}, z)\|^2, \kappa) \\ + \ (1 - O_i(\mathbf{x}, \mathcal{Z}))\, \nu \end{pmatrix} \\
S(\{\mathbf{p}, \mathbf{q}\}, \mathcal{Z}) &:= \left| \frac{1}{Z(\mathbf{p})} - \frac{1}{Z(\mathbf{q})} \right| \\
E_{\text{texture}}(\mathcal{V}, \mathcal{N}) &:= 1 + \lambda_{\text{t}} \min\left(\min_{\mathbf{T} \in \mathbb{T}_{\mathcal{N}}} \|\mathbf{T} - \mathbf{V}(\mathcal{N}, \mathcal{Z})\|^2, \delta_{\text{t}} \right) \\
E_{\text{smooth}}(\mathcal{Z}) &:= \sum_{\mathcal{N} \in \mathbb{N}} E_{\text{texture}}(\mathcal{V}, \mathcal{N}) \lambda_{\text{s}} \min\left(S(\mathcal{N}, \mathcal{Z}), \delta_{\text{s}} \right) \\
E(\mathcal{Z}) &:= \underbrace{E_{\text{photo}}(\mathcal{Z})}_{\text{data costs}} + \underbrace{E_{\text{smooth}}(\mathcal{Z})}_{\text{surface smoothness}}
\end{aligned}
$$

Figure 1: **Energy function.** The energy $E(\mathcal{Z})$ minimized as a function of the new-view depth map $\mathcal{Z}$. Note that although complex, with many terms, this function can be effectively reduced to a sequence of binary optimization problems, for which the QPBO algorithm finds either a global optimum, or a local optimum with an indication of how far from the global optimum it is.

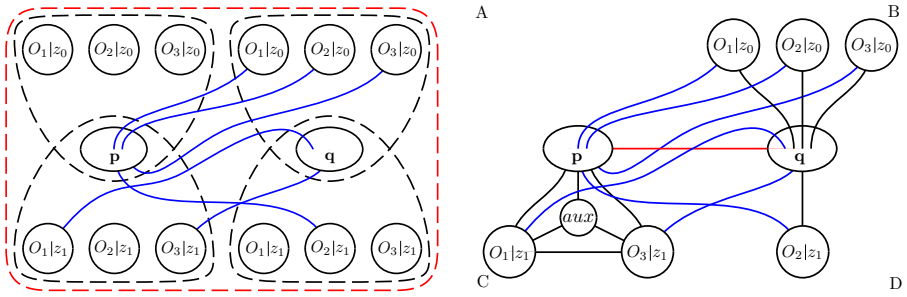
Rather, we can now use the Quadratic Pseudo-Boolean Optimization (QPBO) strategy introduced to computer vision in [8]. QPBO is an extension of graph cuts that can be used to optimize non-submodular energies. Unlike the globally optimal submodular case, QPBO returns a partial solution to $\mathcal{B}$ and an associated mask $\mathcal{M}$, with the guarantee that at pixels $\mathbf{x}$ where $M(\mathbf{x}) = 1$, the value $B(\mathbf{x})$ is at the value it would have at the global minimum, but pixels where $M(\mathbf{x}) = 0$ have "unlabelled" values. A further guarantee of QPBO is that, after forcing $B(\mathbf{x}) = 1$ at those unlabelled pixels, $E(Z_{t+1}) \leq E(Z_t)$, thus ensuring a convergent optimization. In practice, we find that, while there may be many unlabelled pixels at each fusion step, those pixels for which the proposal depth is optimal tend to be labelled, so the energy is minimized quite effectively.

In principle our choice of proposal depth map is not constrained when using QPBO, but we emulate the simple approach of [12] in setting the proposal at each fusion step to be a fronto-parallel plane at one of a discrete set of depths.

3.1 Graph construction

NVS has the additional complexity over stereo that the colour of pixel $\mathbf{x}$, as given by equation (7) depends not only on its depth (current or proposed), but also on the binary visibilities $O_1(\mathbf{x}, \mathcal{Z}), .., O_N(\mathbf{x}, \mathcal{Z})$. Therefore, in order to accurately model the energy of equation (2) our graph requires cliques of size $N + 1$, as shown in figure 2(a), while equation (6) requires cliques of size $4N + 2$.

QPBO, like all graph cut algorithms, can only solve graphs with cliques up to size two. Energy terms of any order can always be decomposed into a set of pairwise energy terms, with additional, latent nodes, but this set grows exponentially with the clique size. In order to avoid an explosion in graph complexity, we limit our maximum clique size to three. This requires approximations to be made in our graph structure, the details of



(a) Exact energy representation. (b) Approximate, soluble construction.

Figure 2: **Graph construction**. Graphical representations of (a) our objective function, and (b) the approximate energy graph we construct, for a 2×1 pixel image, with $N = 3$. Ellipses (including circles) represent nodes of the graph (and associated unary terms); lines (edges) represent pairwise energy terms. Nodes $\mathbf{p}$ and $\mathbf{q}$ encode the depth labels of the two image pixels. The nodes $O_1|z_0$, *etc*. encode whether (by way of example) pixel $\mathbf{p}$ is occluded at depth label 0 (i.e. depth z_0) in $\mathcal{I}_1$, given the depth labels of all other pixels also. The blue lines are infinite edge costs which set these visibilities, as per [12]. The list of occlusion interactions (*i.e*. blue lines) is computed prior to solving the graph, and it should be noted that not every occlusion node has such an interaction, while others may have more than one. The dashed lines in (a) encircle nodes in higher order cliques, which accurately model the data costs (black lines) of equation (2), and surface smoothness cost (red line) of equation (5). However, since graph cut optimizers can only solve graphs with pairwise and unary terms, we approximate these cliques to generate graph (b) as follows. First, we approximate the surface smoothness cost with a single pairwise edge (red line), by using a fixed approximation of pixel colour in equation (6). Then we remove all occlusion nodes with no occlusion interactions—the image samples associated with those nodes will always be visible—reducing some of the cliques in size. Cliques of size 1, 2 and 3 can then be modelled exactly using unary and pairwise terms (black lines), as shown by the graph structures in corners A, D and C of (b) respectively. In particular, the triple clique energy is decomposed into 6 pairwise terms according to [6], which also generates an additional, latent node, aux. Cliques of size 4 (corner B) or larger are approximated using a set of pairwise edges, as described in §3.1.

which are one of the main contributions of this paper.

To remove the complexity generated by the variability of colour in equation (6), we simply fix the colour of each pixel $\mathbf{x}$ at a given depth z to $V'(\mathbf{x}, \mathcal{Z}_t, z)$, *i.e*. we assume all pixels other than $\mathbf{x}$ to be at the depth output by the previous fusion, thus:

$$V'(\mathbf{x}, \mathcal{Z}_t, z) = \frac{\sum_{i=1}^{N} O_i(\mathbf{x}, \mathcal{Z}_t) I_i(\mathbf{x}, z)}{\sum_{i=1}^{N} O_i(\mathbf{x}, \mathcal{Z}_t)}. \tag{11}$$

Rather than use this approximation as standard in equation (2) as well, we prefer to model the data costs as accurately as possible, as they have a much greater impact on the quality of the solution. Figure 2(b) shows that, once unnecessary occlusion nodes have been removed from the graph, pixels at a given depth with up to two possibly occluded input samples can be modelled exactly with a single unary, pairwise or triple clique term for the data cost over all input images. We can therefore model all data costs exactly when $N = 2$. However, in the case of larger cliques we use the fixed colour, V', in evaluating E_{photo}. Potentially occluded image samples therefore generate a pairwise edge, as in stereo [5, 12], while data costs for unoccluded samples are simply added to the correct unary term of the node representing the pixel in question.

The approximation of equation (11) means we no longer model the true value of our objective function in our graph. When we evaluate the true value of colour, $V(\mathbf{x}, \mathcal{Z}_{t+1})$, given by equation (7), after the fusion operation, some of the pixels will change colour due to the visibilities of the input image samples changing with the new depth map. The result is that the objective energy $E(\mathcal{Z}_{t+1})$ may increase, such that the guarantee of convergence given by the stereo framework is lost. However, we have found this to be rare and negligible in practice.

4 Experiments

In all our experiments we use the parameter set given in table 1, which we chose after a grid search over parameter space and qualitative inspection of the results. We make two passes through the set of depth proposals, which is dependent on the sequence, but numbers of the order of 100 depths spaced equally in disparity space (1/depth); the passes run through the set in order, from near to far. The first pass fixes most pixels, with the second making only a few corrections. While additional passes improve the result further, returns on computation time diminish rapidly. We ran experiments on a range of standard NVS and stereo image sequences, and compared our results with other methods.

Figures 3 & 4 show images synthesized from a viewpoint halfway between the two rectified input views. The former compares our method with warping a known view with a depth map [9] (here we use ground truth[3]). Warped stereo leaves holes sometimes (cyan pixels), but also sets a single depth for mixed depth pixels, which then causes artefacts (*e.g*. around depth discontinuities) when rendered in the new view. By rendering directly into the new view we avoid this rerendering step and its associated artefacts.

Figure 3 also demonstrates the impacts on our synthesis framework, our main contribution, of two further contributions of our work—employing a texture prior to weight the surface smoothness cost, and sensibly approximating data costs in our graph. In image (d) (no texture prior), some of the cone tips are truncated. The aim of our texture prior is to encourage depth discontinuities to fit to the edges of objects, and we can see in (c) that these cone tips have been corrected, as desired. Comparing (c) with (e) demonstrates that accurately modelling data costs in cliques with less than three potentially occluded pixels produces far fewer rendering artefacts, though this improvement becomes less pronounced as N increases.

Figure 4 shows a comparison of our method with the DP method of Criminisi *et al*. [1]. While producing similar results on this sequence, and in real-time, their method enforces the less general "ordering" constraint in modelling occlusions. Our approach is therefore preferable in scenes with complex foreground objects and wide baseline input views. Note that the input images have different exposures—this is handled by equalizing the mean and variance of the two images.

Figure 5 shows a new view of a challenging sequence, with many occlusions, synthesized from 8 input images. Our method is able to reconstruct the colour in occluded regions (*e.g*. wall above nose, and between ribs) well, in contrast to the implicit depth method of Woodford *et al*. The explicit depth model and smoothness prior allows us to extract the correct depth of the wall, and the geometric occlusion model the correct

[3] Sequence and ground truth depth maps downloaded from www.middlebury.edu/stereo.

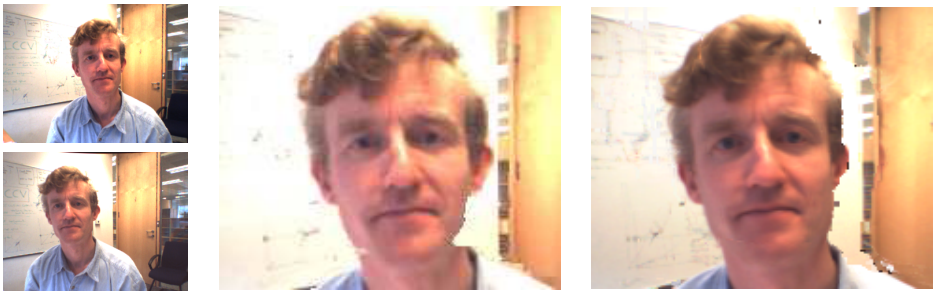
Parameter	κ	ν	δ_s	λ_s	δ_t	λ_t
Value	$c(12.5N/(N-1))^2$	$\kappa+1$	$1.9d$	$0.24\kappa/\delta_s$	$5000c$	$6/\delta_t$

Table 1: **Parameter settings**. Values of the constant parameters in our objective function, where c is the number of colour channels in the input sequence, and d is the constant disparity spacing between the discrete proposal depths, which varies between input sequences.

(a) Ground truth (b) Warped stereo views

(c) Our result (d) $\lambda_t = 0$ (e) Always fix to V'

Figure 3: **Cones sequence**. (a) A ground truth central view, and (b) a view synthesized by warping (in a manner similar to that of [9]) two outer images into the central view using ground truth depth maps—blue pixels are unknown due to holes in the depth maps, while cyan pixels are regions occluded in both input views. Our result (c), and our results (d) removing the texture prior and (e) using the approximate colour of equation (11) in all data cost calculations.



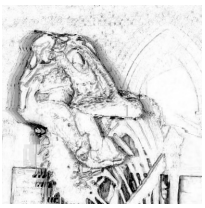
(a) Input views (b) Result of [1] (b) Our result

Figure 4: **Teleconferencing**. Rendering a centre view (c) from 2 rectified input views, for direct gaze teleconferencing. Sequence taken from [1], with the result from the same paper (b).

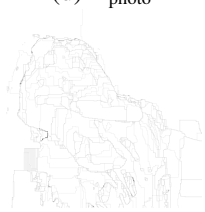
(a) Our result

(b) Method of [13]

(c) $1/\mathcal{Z}$



(d) E_{photo}



(e) E_{smooth}

Figure 5: **Edmontosaurus sequence**. (a) New view of a sequence from [13], and the result of the method of the same paper (b). (c)–(e) show other outputs of our method, as labelled. $N = 8$.

texture. Some artefacts, such as shadows and jaggedness, exist around the edges of the foreground object.

Figure 6 demonstrates the results of our algorithm on a further two difficult sequences. While fine details such as fur and feathers are accurately rendered, some areas (*e.g.* under the forearm and upper arm in (a), to left of head in (b)) appear blurred; this is due to the wrong depth being chosen in these regions.

Artefacts in our results are generated by a combination of two processes: (1) the optimal solution to our objective function not accurately representing the scene, and (2) nodes being unlabelled in each fusion step when the optimal solution would select the proposed new depth. We found that optimizing parameters for a particular sequence or view often produced better results than with the standard parameter set—future work may involving developing methods to automatically evaluate the optimal settings. We expect the performance of QPBO, an algorithm relatively new to the field, to improve significantly in the future, further reducing the appearance of artefacts.

(a) Our result

(b) Our result

(c) Ground truth

Figure 6: **Monkey and plant & toy sequences**. (a) A new view of the monkey sequence (from [2]). $N = 8$. (b) A leave-one-out test on the plant & toy sequence (from [13]). (c) The ground truth view of (b). $N = 8$.

5 Conclusion

We have confirmed the common suggestion that graph-cut stereo methods can be applied to the task of new-view synthesis. While straightforward in principle, this repurposing presents a number of technical difficulties, the solutions to which are the main contributions of this paper. The results improve on the current state of the art NVS methods, demonstrating the power of an explicit depth model with global, geometric occlusion reasoning in determining colour in partially occluded regions, as well as showing that rendering directly into the new view avoids artefacts generated by scene reconstruction methods. While the texture prior which we apply is not in principle as powerful as the stereo CRF prior (which cannot be applied), we show that it acts similarly in improving rendering at discontinuity boundaries.

Acknowledgements We are extremely grateful to Carsten Rother and Vladimir Kolmogorov for providing us with a preliminary version of [8], QPBO software and for generous assistance in using the latter. We also thank Vladimir for discussing graph cut stereo with us. Research funded by EPSRC and Sharp.

References

[1] A. Criminisi, J. Shotton, A. Blake, C. Rother, and P. H. S. Torr. Efficient dense stereo with occlusions for new view-synthesis by four-state dynamic programming. *IJCV*, 71(1):89–110, Jan 2007.

[2] A. Fitzgibbon, Y. Wexler, and A. Zisserman. Image-based rendering using image-based priors. In *Proc. ICCV*, volume 2, pages 1176–1183, Oct 2003.

[3] V. Kolmogorov, A. Criminisi, A. Blake, C. Cross, and C. Rother. Probabilistic fusion of stereo with color and contrast for bi-layer segmentation. *IEEE PAMI*, 28(9):1480–1492, Sep 2006.

[4] V. Kolmogorov and C. Rother. Comparison of energy minimization algorithms for highly connected graphs. In *Proc. ECCV*, volume 2, pages 1–15, 2006.

[5] V. Kolmogorov and R. Zabih. Multi-camera scene reconstruction via graph cuts. In *Proc. ECCV*, volume 3, page 82, 2002.

[6] V. Kolmogorov and R. Zabih. What energy functions can be minimized via graph cuts? *IEEE PAMI*, 26(2):147–159, 2004.

[7] M. Levoy and P. Hanrahan. Light field rendering. In *SIGGRAPH96*, 1996.

[8] C. Rother, V. Kolmogorov, V. Lempitsky, and M. Szummer. Optimizing binary MRFs via extended roof duality. In *Proc. CVPR*, 2007.

[9] D. Scharstein. Stereo vision for view synthesis. In *Proc. CVPR*, pages 852–858, 1996.

[10] C. Strecha, R. Fransens, and L. Van Gool. Wide-baseline stereo from multiple views: a probabilistic account. In *Proc. CVPR*, volume 1, pages 552–559, Jun 2004.

[11] G. Vogiatzis, P. H. S. Torr, and R. Cipolla. Multi-view stereo via volumetric graph-cuts. In *Proc. CVPR*, pages 391–398, 2005.

[12] Y. Wei and L. Quan. Asymmetrical occlusion handling using graph cut for multi-view stereo. In *Proc. CVPR*, volume 2, pages 902–909, 2005.

[13] O. J. Woodford, I. D. Reid, and A. W. Fitzgibbon. Efficient new view synthesis using pairwise dictionary priors. In *Proc. CVPR*, 2007.