A Convex Multi-View Stereo Formulation with Robustness to Occlusions and Time-Varying Clutter

Qingxu Dou*
Heriot-Watt University
Edinburgh, UK
qd5@hw.ac.uk

Paolo Favaro
Heriot-Watt University
Edingurgh, UK
P.Favaro@hw.ac.uk

ABSTRACT

We present a novel framework for multi-view stereo that poses the problem of recovering a 3D surface in the scene as a regularized minimal partition problem of the visibility function in the presence of clutter. We introduce a simple and robust method to integrate estimates from several views that tolerates both static and time-varying clutter. Our formulation does not rely on the visual hull, 2D silhouettes, or make use of initial surface estimates. Furthermore, we use a globally optimal framework, so that the solution does not depend on initialization and computationally efficient numerical methods can be used to find the solution. We also strive for simplicity so that more general models of image formation can be used without compromising the estimation process. Experimental results on synthetic and publicly available real data show that our method performs on a par with state-of-the-art methods that have been used on clutter-free data.

1. INTRODUCTION

In this paper we present a novel solution to calibrated multi-view stereo (MVS), i.e., the problem of estimating 3D surfaces from a collection of 2D views with known pose, in the presence of clutter. Research in MVS is very active in the field of computer vision and a wide variety of methods have been proposed to address the recovery of the surface of 3D objects in very challenging scenarios, e.g., for wide-baseline datasets [8], with non Lambertian objects [3, 17], dealing with illumination [9], or in the presence of clutter [6]. We focus on the challenge posed by clutter and propose a general solution that does not require knowledge of the visual hull, silhouettes or approximate depth maps either implicitly or explicitly.

As pointed out by Furukawa and Ponce [6], MVS algorithms can be associated to the datasets (and their implicit assumptions) that they can handle: Objects with clear sil-

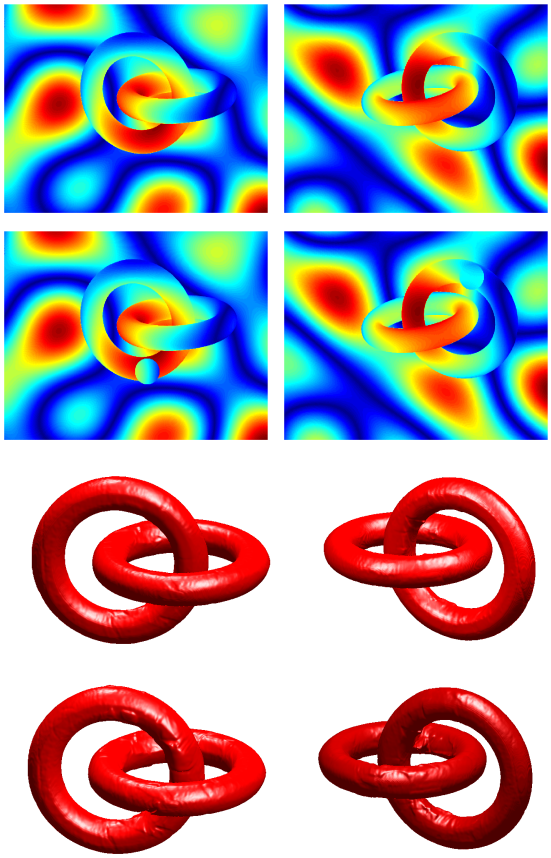

Figure 1: Reconstruction of two concatenated tori from synthetic images where the foreground and the background have the same texture (static clutter) and with or without time-varying clutter (circular regions that change position in each frame). Top row: two of the input images of the `two_tori` synthetic dataset with only static clutter. Second row: two of the input images of the `two_tori` synthetic dataset with time-varying clutter. Third row: two views of the reconstructed 3D model from the `two_tori` synthetic dataset with only static clutter. Bottom row:two views of the reconstructed 3D model from the `two_tori` synthetic dataset with time-varying clutter.

*Corresponding author

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
ICVGIP '10, December 12-15, 2010, Chennai, India
Copyright 2010 ACM 978-1-4503-0060-5/10/12 ...$10.00.

houettes; objects in static clutter (e.g., a background similar to the object of interest); objects with time-varying clutter (e.g., crowded scenes with moving people).

By building on recent work in the literature, we propose a method that can deal with all the three above scenarios in a globally optimal fashion. More specifically, our contribution is that we formulate MVS to simultaneously deal with the dependency of multiple views, to be independent of initialization, to easily incorporate surface regularization terms, not to have degenerate solutions (e.g., the empty set), not to use the visual hull or silhouettes explicitly or implicitly at any step of the algorithm (see Figure 1), to have a unique solution, and to be computationally and memory efficient. Our method is a continuous formulation of MVS (section 3) which combines a Bayesian formulation of the visibility of each camera in a convex cost functional (section 4). In section 5 we explicitly deal with the integration of 3D surface estimates from different views. The convex cost functional is then minimized by an efficient gradient-flow in section 6. Finally, in section 7 we demonstrate the method on data with clutter and on the (uncluttered) Middlebury data set, where we show that it still performs similarly to current state-of-the-art methods.

2. PRIOR WORK

There is a large body of prior work on multi-view stereo algorithms (see [14] for an excellent recent survey). Among the most successful methods, are those based on shape from silhouette, which obtain an estimation of the 3D surfaces from binary object/background segmentations of each view [16, 18]. These methods are known to be robust and computationally efficient, but cannot reconstruct all concavities. Other popular approaches are those based on space carving, where voxels that do not correspond to pixels that are photoconsistent are removed [11]. These methods have the limitation that regularization is not enforced and reconstructions are often noisy. Solutions that incorporate regularization have also been proposed. In [21, 15, 5] a deformable model is updated in a variational minimization scheme until a certain consistency criterion is satisfied. This approach allows to combine a data fidelity term on the unknown surface, which measures how well the solution explains the data, with a regularization term, which constrains the solution to be smooth. Although these methods achieve a higher robustness to image noise, they inherently define the empty set as a global optimum and typically depend on the initialization. To compensate for such limitations, methods that incorporate ballooning terms have been proposed [19]. Other very effective methods merge depth maps obtained from small groups of neighboring views [7, 4]. Our method relates to several of the above methods, and, in particular, to work by Kolev et al. [10] and Nikolova et al. [13], as we also pose the multi-view stereo problem as a globally optimal minimization problem, and Hernandez et al. [7] as we also formulate the problem as a probabilistic 3D segmentation and rely on the computations of the visibility via the depth maps. Our formulation however differs in the specific choices of the noise, the visibility models, and, more importantly, how depth maps are merged, as we outline here below.

3. A CONTINUOUS AND CONVEX FORMULATION OF MULTI-VIEW STEREO

As in most recent MVS work [14], we pose the problem of estimating the surface of Lambertian objects in the scene from multiple calibrated views as the problem of determining whether a point in space (a voxel) lies inside or outside any of the objects. The estimated surface is then implicitly defined as the interface separating the two groups of voxels.

Let us represent such solution with a function $\phi : V \subset \mathbb{R}^3 \mapsto [-1, 1]$, with V the bounded volume in 3D space where reconstruction is performed. In our approach the function ϕ defines when a voxel $\mathbf{X} \in V$ is inside or outside an object. We call this function *visibility* of a voxel. The surface of the objects is defined implicitly as the set of points $\{\mathbf{X} : \phi(\mathbf{X}) = 0\}$. The next step is to define an energy such that its minimum is at the surface of the objects in the scene. To do so we introduce the following energy minimization

$$\begin{aligned}\hat{\phi} &= \arg\min_{\phi} E[\phi] \\ &\doteq \int \Phi(\tilde{\phi}(\mathbf{X})|\phi(\mathbf{X}))d\mathbf{X} + \alpha \int \Psi(\tilde{\phi}(\mathbf{X}))|\nabla\phi(\mathbf{X})|d\mathbf{X} \\ &\quad +\beta \int \theta(\phi(\mathbf{X}))d\mathbf{X}. \end{aligned} \tag{1}$$

The energy is composed of three terms: $\Phi(\epsilon|\gamma)$, which measures the discrepancy between ϵ and γ, $\Psi(\epsilon) \doteq \exp[-\epsilon^2/\mu]$ with positive constants α and μ, which penalizes large variations of ϕ at the surface of the object, and $\theta(\epsilon) \doteq \max\{0, |\epsilon| - 1\}$ with positive constant β, which is a convex penalty term that prevents ϕ from leaving the range $[-1, 1]$. In our notation the function $\tilde{\phi}$ is an approximate estimate of the visibility that we obtain, for instance, by combining the depth maps from several vantage points.

In our implementation we tested two choices for Φ. One choice is the discrepancy

$$\Phi(\tilde{\phi}(\mathbf{X})|\phi(\mathbf{X})) = |\tilde{\phi}(\mathbf{X}) - \phi(\mathbf{X})| \tag{2}$$

and a second choice is

$$\Phi(\tilde{\phi}(\mathbf{X})|\phi(\mathbf{X})) = (1+\tilde{\phi}(\mathbf{X}))(1-\phi(\mathbf{X}))+(1-\tilde{\phi}(\mathbf{X}))(1+\phi(\mathbf{X})) \tag{3}$$

If $\tilde{\phi}$ is either -1 or $+1$ for most of the voxels and has a quick transition through 0 at the surface of the objects, then we have found no noticeable difference between the solution obtained with eq. (2) and the solution obtained with eq. (3) in our experiments.

The interpretation of the term $\tilde{\phi}$ in eq. (2) is much more apparent than in eq. (3): It behaves as a *proxy*, i.e., as an initial estimate of the function ϕ obtained from the data. Then, by minimizing eq. (1) we approximate the proxy with a smooth function. An immediate consequence of this formulation is that the accuracy of the solution depends highly on the accuracy of the proxy. In this paper we will study how to calculate $\tilde{\phi}(\mathbf{X})$ so that we can tolerate discrepancies in the model due to sensor noise, changes in the brightness and contrast of the camera, departure from the Lambertian assumption, or to occlusions caused by clutter.

3.1 Relation to Kolev et al. [10]

In work by Kolev et al. [10], the energy term relative to the measurements and the model is defined as

$$E_{Kolev}(u) \quad = \quad \int (\rho_{bck}(\mathbf{X}) - \rho_{obj}(\mathbf{X}))u(\mathbf{X})d\mathbf{X} \tag{4}$$

where $\rho_{bck}(\mathbf{X}) + \rho_{obj}(\mathbf{X}) = 1$ and $\rho_{bck}(\mathbf{X})$ and $\rho_{obj}(\mathbf{X})$ depend on depth maps estimates (obtained from different points of view). For instance, they can be defined as the negative log-likelihood of $\mathbf{X}$ belonging to the object or the background. One of the nice features of this energy formulation is that the two terms $\rho_{obj}(\mathbf{X})$ and $\rho_{bck}(\mathbf{X})$ "compete" to define whether the voxel $\mathbf{X}$ lies inside or outside any of the objects. By minimizing this energy they obtain a solution u that takes $+\infty$ on voxels inside objects ($\rho_{obj}(\mathbf{X}) > \rho_{bck}(\mathbf{X})$), and $-\infty$ outside ($\rho_{obj}(\mathbf{X}) < \rho_{bck}(\mathbf{X})$). By adding a convex energy term that penalizes values of u out of the range $[0, 1]$, u will instead become the indicator function of the inside of the objects.

Now, define the following identities

$$\phi(\mathbf{X}) \doteq 2u(\mathbf{X}) - 1 \tag{5}$$

$$\tilde{\phi}(\mathbf{X}) \doteq \rho_{obj}(\mathbf{X}) - \rho_{bck}(\mathbf{X}) = 2\rho_{obj}(\mathbf{X}) - 1 \tag{6}$$

in the above energy term. The constraint on $u \in [0, 1]$ becomes $\phi \in [-1, +1]$. Since $\rho_{obj} \in [0, 1]$ by definition, we also have that $\tilde{\phi} \in [-1, +1]$. The resulting energy is identical, up to a constant scale factor, to eq. (3). Hence, one can immediately conclude that the term $\rho_{obj} - \rho_{bck}$ also defines an initial estimate of the surface of the objects and the accuracy with which it is obtained determines the overall performance of the reconstruction task.

4. BAYESIAN PHOTOCONSISTENCY

To determine the proxy $\tilde{\phi}$ we obtain depth maps from small groups of nearby views, so that outliers due to occlusions are minimized, and then merge them into a single 3D surface [12, 22]. In this paper we follow a similar merging strategy, but try to delay as much as possible hard decisions so as to maximize the amount of information used to take them. In broad terms, the key idea is to obtain a visibility map from each depth map and then to integrate all the visibility maps together via a robust interpolating function. To illustrate the steps needed, here we will show the computation of a single visibility map. The integration of all the visibility maps will be discussed in the next section.

In order to compute a depth map, we need to define how images are generated from the scene. Let $\{I_i : \Omega \subset \mathbb{R}^2 \mapsto \mathbb{R}^3_+\}_{i=1,\dots,N}$ be a collection of N calibrated color images, $\{\pi_i : V \mapsto \Omega\}_{i=1,\dots,N}$ be perspective projections of a voxel to pixel coordinates in the i-th view, and $\{\mathbf{C}_i \in V\}_{i=1,\dots,N}$ be the camera centers. Under the Lambertian assumption the intensity measured on the i-th camera sensor can be written as

$$\begin{aligned} I_i(\pi_i[\mathbf{X}]) &= r((\mathbf{X} - \mathbf{C}_i)\lambda_i^* + \mathbf{C}_i) \qquad \text{where} \\ \lambda_i^* &= \arg \min_{\lambda \in [0,\infty)} \{\lambda | \phi(\lambda \mathbf{X} + (1-\lambda)\mathbf{C}_i) = 0\} \end{aligned} \tag{7}$$

and $r : V \mapsto \mathbb{R}^3_+$ is the color intensity reflected at a point in space. The above definition formalizes two well-known notions: 1) If two images capture light from the same point in space, the same intensity is observed (photoconsistency); 2) The intensity captured by an image at a pixel $\pi_i[\mathbf{X}]$ depends on the closest point on the surface along the ray connecting the camera center $\mathbf{C}_i$ to the point in space $\mathbf{X}$.

Given the i-th view I_i we are interested in computing an estimate of the visibility of a point from this camera. In this case we have that if a point $\mathbf{X}$ on the surface is visible from both the i-th and the j-th camera then $I_j(\pi_j[\mathbf{X}]) = I_i(\pi_i[\mathbf{X}])) + \omega$, where ω is sensor noise, which we model with a Laplace distribution. Then we can write

$$\rho_{i,j}(\mathbf{X}) = \frac{\sigma}{2I_i(\pi_i[\mathbf{X}])} e^{-\sigma \left| \frac{I_j(\pi_j[\mathbf{X}])}{I_i(\pi_i[\mathbf{X}])} - 1 \right|} \tag{8}$$

where $\frac{1}{\sigma} I_i(\pi_i[\mathbf{X}])$ is the scale parameter. It is reasonable to assume that sensor noise in each view is independent from the other views. Thus the photoconsistency of M views can be computed as the product of individual pairwise photoconsistency terms. The quantity $\rho_{i,j}(\mathbf{X})$ is the probability of photoconsistency and is maximal at the surface when all points are visible and distortions are well modeled by Laplacian noise. Notice that the long tails of the Laplacian distribution allow to compensate for occlusions and other distortions. Furthermore, the degree of tolerance to outliers can be varied by changing the scale parameter. By combining the different views, we obtain

$$\rho_i(\mathbf{X}) = \prod_{j=j_1}^{j_M} \rho_{i,j}(\mathbf{X}). \tag{9}$$

REMARK 1. *The above model rejects outliers similarly to other robust functions that have been suggested in the literature (see, for instance [20, 2]). In practice, the overall behavior is that the photoconsistency term should be as sensitive as possible to small intensity deviations between the views, which are more likely to have been generated by a genuine point on the surface, rather than large intensity deviations, which might have been generated by extremely different phenomena (e.g., occlusions, clutter, and quantization).*

The computation of the photoconsistency term eq. (9) can be done in a reasonably efficient manner by parsing each point in the volume V. We simply compute the photoconsistency probability independently at each point $\mathbf{X}$ in space. We would like to point out that for the sake of simplicity we do not integrate the visibility within windows or slanted planes or compute normalized cross-correlations, although such options are all possible in our framework.

Once eq. (9) has been evaluated, we map the photoconsistency probability ρ_i to the visibility of a point $\mathbf{X}$ from the i-th view. Notice that the visibility $\tilde{\phi}_i$ must be a non-decreasing function as we evaluate voxels along a ray from the camera center C_i. We enforce such constraint by considering the integral of $\rho_i(\mathbf{X})$ along the projection ray passing through $\mathbf{C}_i$. We then shift and truncate such function so that the visibility $\tilde{\phi}_i$ is 0 at the depth map, and between -1 and 1 everywhere else. First, for each ray, we compute the location of the depth map

$$\lambda^* \doteq \arg \max_{\lambda \in [0,1]} \rho_i(\mathbf{C}_i + \lambda(\mathbf{X}_{max} - \mathbf{C}_i)) \tag{10}$$

where $\mathbf{X}_{max}$ is the furthest point from the camera $\mathbf{C}_i$ along the chosen ray in the volume V. Then, we define the visibility from the i-th view along the ray via the cumulative distribution function of ρ_i, i.e.,

$$\begin{aligned} &\tilde{\phi}_i(\mathbf{C}_i + \mu(\mathbf{X}_{max} - \mathbf{C}_i)) = \\ &\quad \max\left\{-1, \min\left\{1, \int_{\lambda^*}^{\mu} \rho_i(\mathbf{C}_i + \lambda(\mathbf{X}_{max} - \mathbf{C}_i))d\lambda\right\}\right\}, \\ &\qquad \forall \mu \in [0, 1]. \end{aligned} \tag{11}$$

Once we have obtained estimates of the visibility $\tilde{\phi}_i$ from each view, we need to integrate them together in a single

methods	0%accu	0%com	1%accu	1%com	2%accu	2%com	3%accu	3%com	accu(clutter)	com(clutter)
robust	9.01e-5	100%	1.03e-4	100%	1.58e-4	99.99%	2.75e-4	99.73%	0.0014	89.04%
geometric	4.08e-4	99.42%	5.06e-4	98.29%	6,51e-4	96.53%	7.79e-4	94.87%	0.0033	67.50%
local	1.58e-4	99.88%	1.81e-4	99.70%	5.06e-4	95.75%	0.001	89.83%	0.0048	48.81%
ideal	1.65e-4	99.62%	3.28e-4	98.37%	8.93e-4	90.27%	0.0015	78.54%	0.0047	46.92%
min	3.03e-4	98.83%	7.72e-4	93.27%	0.0017	75.59%	0.0025	56.20%	0.0046	46.99%

Table 1: Performance on the `bigball` synthetic dataset with 5 different interpolating functions: `robust`, `geometric`, `local`, `ideal` and `min` (see the text for their definition). The percentages in the first row indicate the noise levels. `accu` is a shorthand notation for accuracy and `com` for completeness. The unit of the accuracy is meter. The last column shows the performance when time-varying clutter (a randomly placed disk) is present.

visibility function $\tilde{\phi}$. This will be presented in the next section.

5. INTEGRATING MULTIPLE VIEWS IN THE PRESENCE OF CLUTTER

The most common method to integrate depth maps, or visibilities, is to determine which cameras share overlapping views. This can be achieved by using an initial estimate of the surface from the visual hull (i.e., via silhouettes) or from the depth maps themselves. The normals to the surface are then extracted and used to determine which cameras are potentially imaging a given voxel. This assumption results in a simple MVS formulation as one only needs to average preselected sets of overlapping depth maps. This, however, comes at a cost, as the normals from an approximate surface estimate could be incorrect and a hard decision on which cameras overlap could lead to averaging depth maps incorrectly.

The procedure just described can be written as a certain interpolating function $f : \mathbb{R}^N \mapsto [-1, 1]$, so that our visibility estimate is given by the simultaneous combination of all the visibilities

$$\tilde{\phi}(\mathbf{X}) = f(\tilde{\phi}_1(\mathbf{X}), ..., \tilde{\phi}_N(\mathbf{X})). \tag{12}$$

In this general formulation however, it is easier to create a consistent integration of the visibilities and to take into account clutter, occlusions and noise. The function f can be seen as a *voting* heuristic, where the vote cast by each visibility results in a decision for each voxel in space. Rather than defining f for each combination of votes, one can define it at some key locations and then let multi-linear interpolation fill the gaps. Furthermore, in defining f it is important to make sure that some desirable properties are satisfied. For instance, f should be invariant to permutations of the input parameters. Notice that f could be "learned" from training data so that one could determine the most robust integration method for a given camera configuration. In this paper, however, we do not investigate this direction.

In absence of a training procedure, we analyze several choices and report their performance (see Fig. 2 for 2 examples of f with only 2 visibilities):

min: This is one of the simplest and most computationally and memory efficient functions as its evaluation can be done recursively as visibilities become available

$$f(\xi_1, \cdots, \xi_N) = \min_i \xi_i. \tag{13}$$

Unfortunately it is also one of the worst performing ones

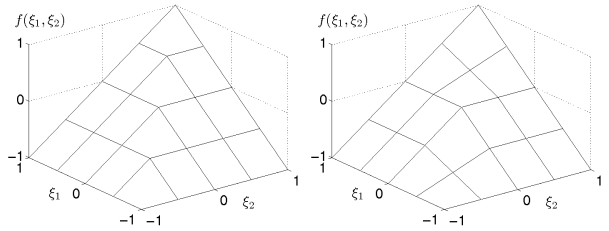

Figure 2: Interpolating functions with 2 views ξ_1 and ξ_2. Left: the `min` interpolating function. Right: the `ideal` interpolating function. Notice that if we threshold $f(\xi_1, \xi_2)$ at 0 both choices result in the same decision. However, a small perturbation of the inputs reveals that the `ideal` choice is biased towards the "no object" decision.

(see Table 1). Notice that this mapping is non linear in the arguments.

ideal: This interpolating function is suitable only when each visibility is perfect. If any of the input parameters is -1 then also f maps to -1. This corresponds to any camera agreeing that a voxel is outside any object. If all parameters are 1 then also f is 1. This corresponds to all cameras agreeing that a voxel is inside an object. Also, if all cameras find a voxel on a surface, i.e., the visibility is 0, then also f must be 0. More explicitly, we define f at the finite set of locations in the N-dimensional grid $\{-1, 0, 1\} \times \cdots \times \{-1, 0, 1\}$ as follows (an example of 2-dimensional grid $\{-1, 0, 1\} \times \{-1, 0, 1\}$ is used in Fig. 2):

$$f(\xi_1, \cdots, \xi_N) = \begin{cases} -1, & \text{if } \exists\, i : \ \xi_i = -1 \\ 1, & \text{if } \xi_i = 1 \quad \forall\, i \\ 0, & \text{at all other locations} \end{cases} \tag{14}$$

and then use N-linear interpolation away from the N-dimensional grid.

local: This interpolating function mimics the choice made by methods that use depth information to decide which depth maps to average. The idea is to average the visibilities at voxels close to the surface and to take the minimum

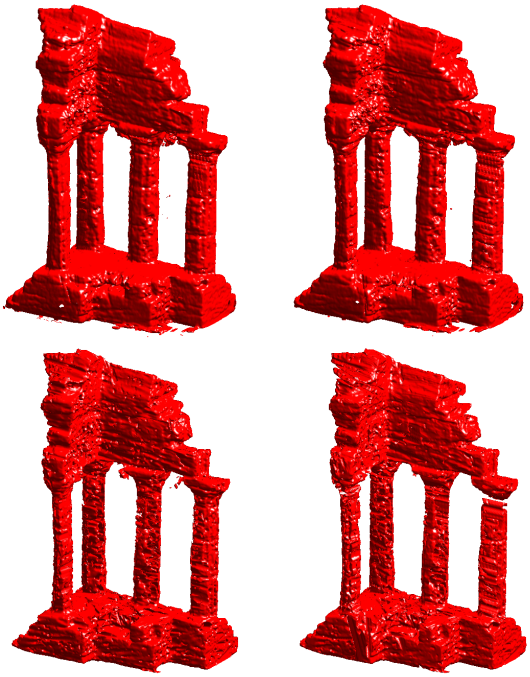

Figure 3: Comparison of the reconstructed visibilities $\tilde{\phi}$ (without smoothing) of four interpolating function choices with the `temple` dataset. In clockwise order from the top-left, they are the results of: `robust`, `geometric`, `ideal`, and `min` interpolating functions. The `robust` interpolating function is more successful than the other functions at preserving the correct surface of the object.

visibility value at voxels far from the surface:

$$f(\xi_1, \cdots, \xi_N) = \begin{cases} \min\limits_i \xi_i, & \text{if } \forall i: \ |\xi_i| > \tau \\ \dfrac{1}{N'} \displaystyle\sum_{i=1}^{N'} \xi_i, & \text{if } \exists i_1, i_2, \cdots, i_{N'} : |\xi_{i_k}| \leq \tau \end{cases} \tag{15}$$

for a small positive constant τ.

geometric: This interpolating function integrates the visibilities from each view by computing a geometric mean. The idea is to reduce the effect of occlusion on the visibility:

$$f(\xi_1, \cdots, \xi_N) = \left(\prod_{i=1}^{N} (1 + \xi_i) \right)^{1/N} - 1. \tag{16}$$

robust: The above interpolating functions are suitable for small errors or inconsistencies between the visibilities $\tilde{\phi}_i(\mathbf{X})$. When images contain static or time-varying clutter, the combination of the visibilities needs to tolerate conflicting terms. For instance, an incorrect depth estimate where one visibility is zero at the exact location would spoil the whole estimate of the visibility in all above interpolating functions (see Figure 3). This effect introduces a bias towards carving and it becomes particularly evident when clutter is present. We propose to use a robust interpolating function that can tolerate a (small) percentage M of incorrect visibilities. As in the case of the ideal interpolating function, we define

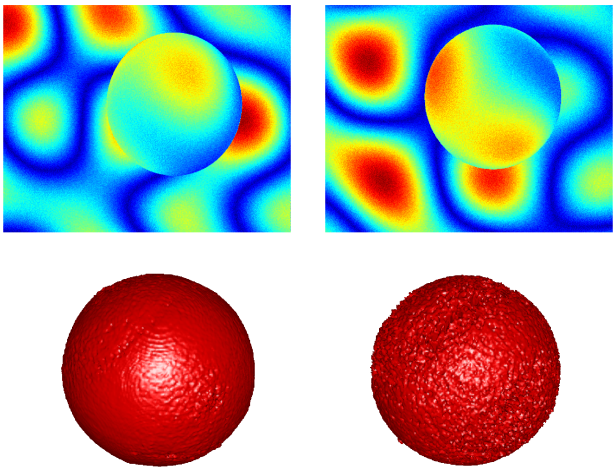

Figure 4: Comparison between the reconstructed results of the `bigball` synthetic dataset without time-varying clutter at 3% noise level. Top row: Two of the 60 input images. Bottom row: The reconstructed 3D model with robust interpolating function (left) and min interpolating function (right).

f at the finite set of locations in the N-dimensional grid $\{-1, 0, 1\} \times \cdots \times \{-1, 0, 1\}$ as follows:

$$f(\xi_1, \cdots, \xi_N) = \begin{cases} 0, & \text{if } \exists\ i_1, \cdots, i_M : \xi_{i_k} = 0 \\ 1, & \text{if } \exists\ i_1 \cdots i_{N-M+1} : \xi_{i_k} = 1 \\ -1, & \text{at all other locations} \end{cases} \tag{17}$$

and then use multi-linear interpolation away from the N-dimensional grid. The meaning of the above definition is that we set a voxel to be inside the object if at least $N-M+1$ visibilities agree that it is inside the object. Similarly, we set a voxel to be on the surface of the object if at least M visibilities think that it is on a surface. All other combinations in the finite grid are set to outside the object. This is particularly effective in the presence of static or time-varying clutter, where several visibilities may be incorrect at some locations. In our experiments we use $M = 3$.

In Figure 4 we show some experiments on synthetic data that demonstrate the effectiveness of the approach in dealing with clutter while still performing well in uncluttered data. We run several experiments where we consider images with different levels of noise and static clutter (a background with texture similar to the object of interest) and time-varying clutter (a disk with texture similar to the object of interest and placed randomly across the images). The results are summarized in Table 1 where one can appreciate the robustness of the proposed interpolating function. We also show a direct comparison of 4 interpolation choices on the Middlebury dataset in Figure 3. In clockwise order from the top-left we show the reconstructed visibility (without smoothing) at the surface voxels for: **robust**, **geometric**, **ideal**, and **local** interpolating functions. As one can see, the **robust** interpolating function is more successful than the other functions in preserving more of the surface and at the same time in avoiding artifacts due to incorrect visibility estimates.

One shortcoming of the proposed approach is that the in-

terpolating function grows in complexity with the number of views that it integrates. While this is perfectly tolerable for medium-size reconstructions, it is unmanageable for large-size reconstructions. Addressing this challenge is beyond the investigation in this paper.

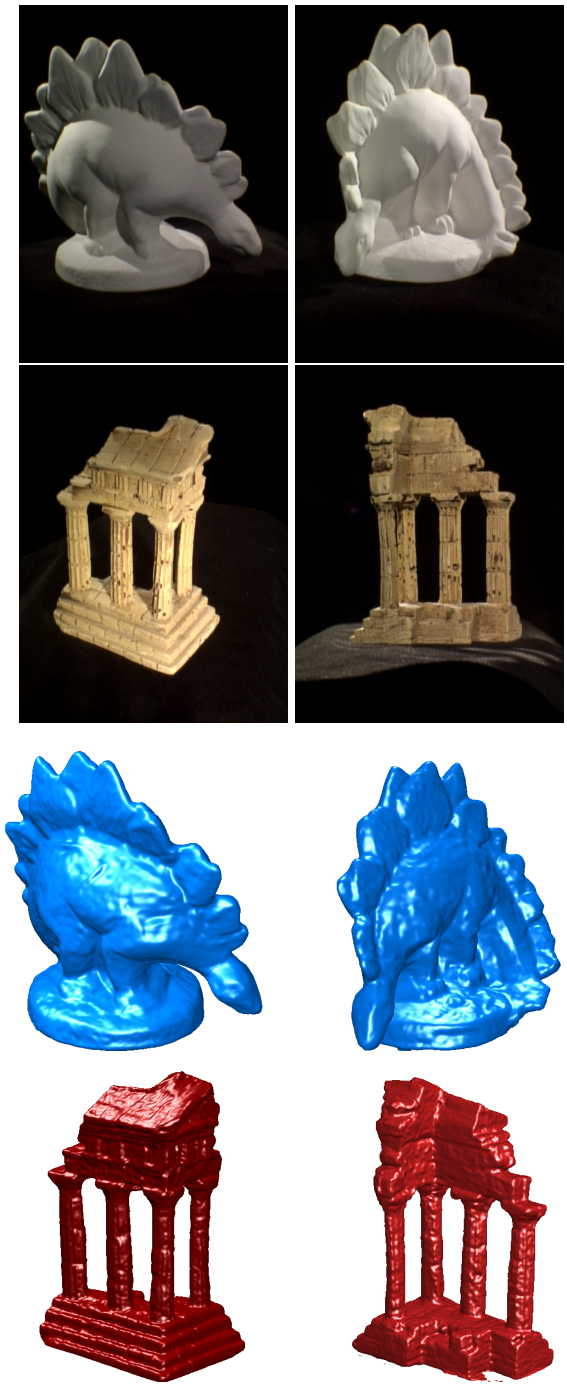

Figure 5: The reconstruction of `dinosaur` dataset and `dinosaur` dataset without static or time-varying clutter. The first two rows show two input images of each dataset. The last two rows show two views of the reconstructed 3D model of each dataset obtained with the robust interpolating function.

6. NUMERICAL IMPLEMENTATION

Now that we have defined all the functions and parameters for the minimization problem (1), we can solve it by first computing the Euler-Lagrange equations and then using a numerical scheme to solve them. In the case of the discrepancy term (2) we have

$$\nabla E[\phi(\mathbf{X})] = \frac{\phi(\mathbf{X}) - \tilde{\phi}(\mathbf{X})}{|\phi(\mathbf{X}) - \tilde{\phi}(\mathbf{X})|} - \alpha\nabla \cdot \left(\Psi(\tilde{\phi}(\mathbf{X}))\frac{\nabla\phi(\mathbf{X})}{|\nabla\phi(\mathbf{X})|}\right) + \beta\theta'(\phi(\mathbf{X})) = 0 \qquad \forall \mathbf{X} \in V \tag{18}$$

and in the case of the discrepancy term (3) we have

$$\nabla E[\phi(\mathbf{X})] = -2\tilde{\phi}(\mathbf{X}) - \alpha\nabla \cdot \left(\Psi(\tilde{\phi}(\mathbf{X}))\frac{\nabla\phi(\mathbf{X})}{|\nabla\phi(\mathbf{X})|}\right) + \beta\theta'(\phi(\mathbf{X})) = 0 \qquad \forall \mathbf{X} \in V. \tag{19}$$

These equations could be solved via linearization and successive over-relaxation (or other iterative solvers for linear systems). However, we find that a gradient descent works quite efficiently on these functionals and most of the iteration time is actually spent in the pre-computation of the estimate $\tilde{\phi}$. Notice that we prevent any division by zero by introducing a small positive constant. The solution of the Euler-Lagrange equations via a gradient descent is given by

$$\phi(\mathbf{X}, t+1) = \phi(\mathbf{X}, t) - \varepsilon\nabla E[\phi(\mathbf{X}, t)] \tag{20}$$

for a small step $\varepsilon > 0$. Starting from any initial condition, this iterative scheme will converge to the global minimum of functional (1). Notice that the smoothness term is formulated as total variation and therefore it tends to yield 3D surfaces that are piecewise smooth.

7. EXPERIMENTS

To demonstrate the effectiveness of the proposed method we use the two multi-view stereo data sets publicly available at the Middlebury website [1]: `dinosaur` and `temple` datasets. We test our algorithms by working on the full collection of images. The performance results that we have obtained in the case of uncluttered scenes with the **robust** and **geometric** interpolating functions are shown in Table 2. Two input images of each dateset and two views of the reconstructed models from **robust** interpolating function are shown in Figure 5. The reconstructions obtained with the two discrepancies eq. (3) and eq. (2) are virtually identical; so, we only display the results obtained for eq. (3). In Figure 6 and 7 we show the input images and corresponding reconstructed results of `dinosaur` and `temple` datasets with synthetic static and time-varying clutter. In the top two rows we show two central views of each dataset with the occlusions highlighted and magnified. The third row shows two views of the reconstructed model from **robust** interpolating function and bottom row shows two views of the reconstructions obtained by **geometric** interpolating function. It can be seen that in the presence of time-varying occlusions, the reconstruction results from the **robust** interpolating function are still comparable to the ones obtained with uncluttered images, but the results obtained from other interpolating functions are much worse. Here we just display the results from the **geometric** interpolating function because of the space limitations. We used a Mac Pro 8-core

Data set	completeness	accuracy	#views
dinosaur(robust)	99.3%	0.60 mm	60
temple(robust)	98.6%	0.76 mm	56
dinosaur(geometric)	95.7%	1.94 mm	60
temple(geometric)	89.6%	1.45 mm	56

Table 2: Performance on the `dinosaur` and `temple` datasets with two different interpolating functions `robust` and `geometric`. Notice that despite the simplicity of the proposed approach, the robust interpolation performs at the same level as the state-of-the-art in MVS.

3.2GHz and with non-optimized Matlab code. The reconstructions are defined on a volume of $359 \times 301 \times 307$ voxels for the `dinosaur` dataset and $465 \times 301 \times 219$ voxels for the `temple` dataset and are produced in less than 30 minutes after the computation of the visibility $\tilde{\phi}$. The time taken for computing the visibility function depends on how many reference views have been used. If 12 reference views are used in `dinosaur` dataset, it takes about 245 minutes for **robust** and **ideal** interpolating functions and about 72 minutes for **geometric**, **local** and **min** interpolating functions.

8. CONCLUSION

We have presented a novel framework for multi-view stereo in the presence of time-varying clutter. We cast the problem as that of recovering the smooth surface separating voxels in the scene that are outside objects from voxels that are inside objects via a robust integration of depth map estimates from different vantage points. This framework has been designed to avoid relying on some estimate of the reconstructed object either via the visual hull or silhouettes. The proposed approach can also be easily modified to take into account novel image formation models or to incorporate general regularization schemes in a globally optimal and computationally efficient numerical implementation. We have illustrated how to robustly perform the integration of all visibilities simultaneously so as to tolerate both static and time-varying clutter. Experimental results on synthetic and publicly available real data demonstrate the effectiveness of the proposed method.

9. ACKNOWLEDGEMENTS

This work has been partly supported by EPSRC grant EP/F023073/1(P) and ONR grant N00014-09-1-1067.

10. REFERENCES

[1] http://vision.middlebury.edu/.

[2] B. Appleton and H. Talbot. Globally minimal surface by continuous maximal flows. *IEE Transactions on Pattern Analysis and Machine Intelligence*, 28:106–118, 2006.

[3] T. Bonfort and P. Sturm. Voxel carving for specular surface. *ICCV*, pages pages 691–696, 2003.

[4] D. Bradley, T. Boubekeur, and W. Heidrich. Accurate multiview reconstruction using robust binocular stereo and surface meshing. *CVPR*, pages pages 1–8, 2008.

[5] O. Faugeras and R. Keriven. Complete dense stereovision using level set methods. *ECCV*, pages 379–393, 1998.

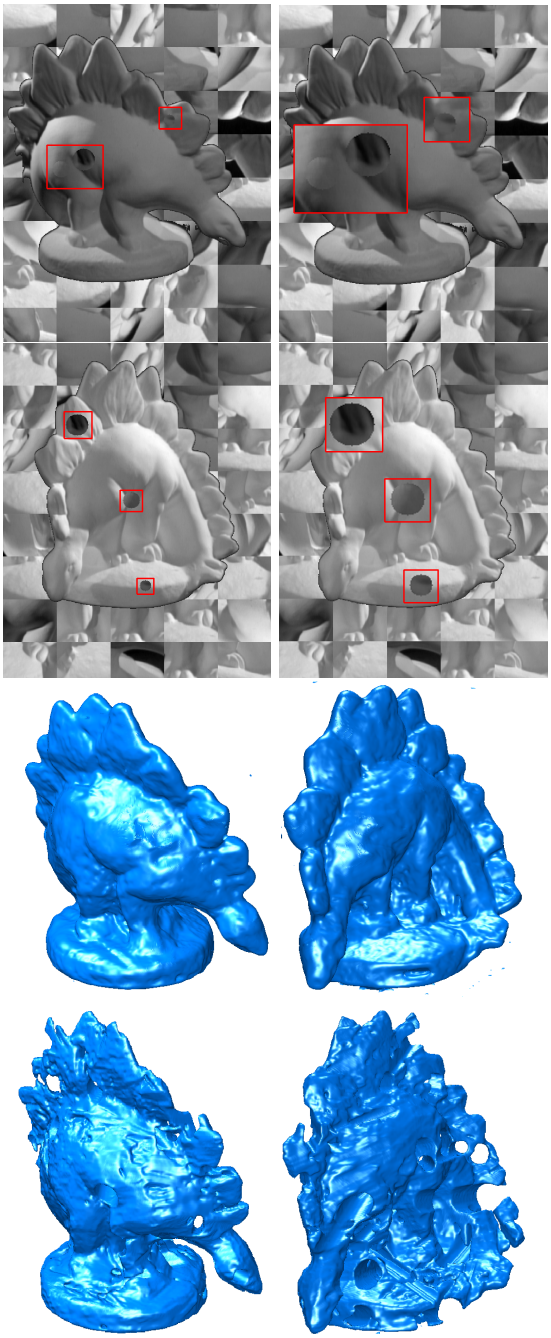

Figure 6: Input images and reconstructed results from the `dinosaur` dataset in the presence of static and time-varying occlusions. The left images in the first two rows are two central views with the occlusions included in the red boxes. The right images in the first two rows are the same views with occlusions magnified. The third row shows two views of the reconstructed model from the `robust` interpolating function and the bottom row shows two views of the reconstructed model from the `geometric` interpolating function.

[6] Y. Furukawa and J. Ponce. Accurate, dense and robust

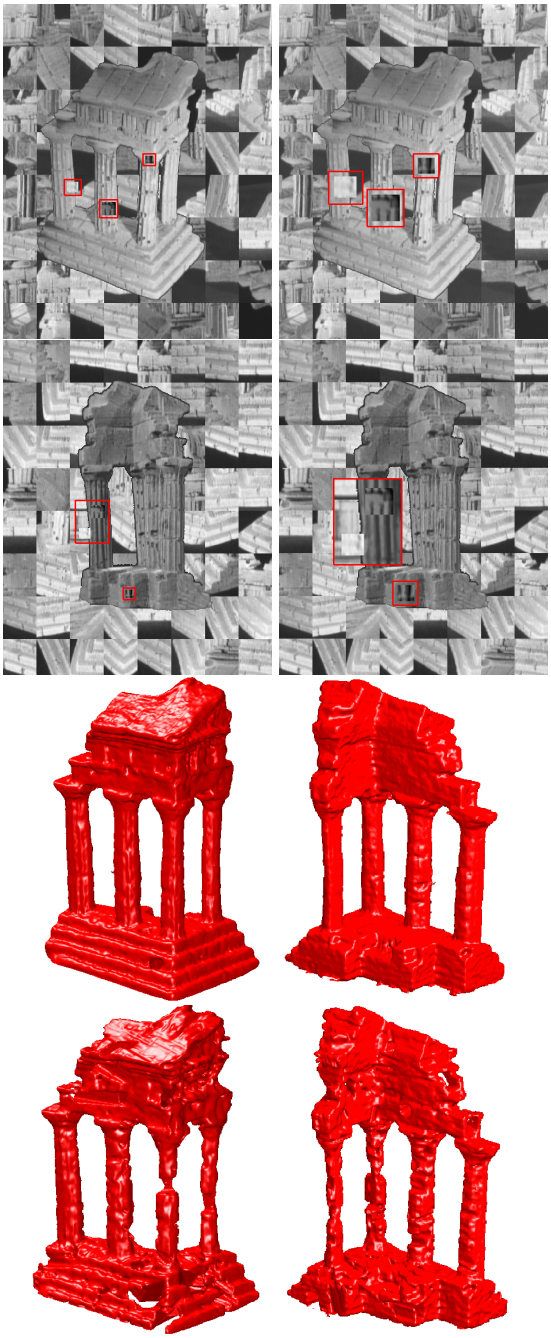

Figure 7: Input images and reconstructed results from the `temple` dataset in the presence of static and time-varying occlusions. The left images in the first two rows are two central views with the occlusions included in the red boxes. The right images in the first two rows are the same views as the left ones with occlusions magnified. The third row shows two views of the reconstructed model from the `robust` interpolating function and the bottom row shows two views of the reconstructed model from the `geometric` interpolating function.

multi-view stereopsis. *IEEE transactions on Pattern Analysis and Machine Intelligence*, Vol. 1, 2008.

[7] C. Hernandez, G. Vogiatzis, and R. Cipolla. Probabilistic visibility for multi-view stereo. *CVPR*, page pages, 2007.

[8] V. H. Hiep, R. Keriven, P. Labatut, and J.-P. Pons. Towards high-resolution large-scale multi-view stereo. *CVPR*, 2009.

[9] H. Jin, D. Cremers, A. Yezzi, and S. Soatto. Shedding light on stereoscopic segmentation. *CVPR*, Vol. 1:36–42, 2004.

[10] K. Kolev, M. Klodt, T. Brox, and D. Cremers. Continuous global optimization in multiview 3d reconstruction. *International Journal of Compuer Vision*, 84:80–96, 2009.

[11] K. N. Kutulakos and S. M. Seitz. A theory of shape by space carving. *Int. Journal of Computer Vision*, Vol. 38(3):199–218, 2000.

[12] P. Merrell, A. Akbarzadeh, L. wang, P. Mordohai, J.-M.Frahm, R.Yang, D. Nister, and M. Pollefeys. Real-time visibility-based fusion of depth maps. *ICCV*, 2007.

[13] M. Nikolova, S. Esedoglu, and T. F. Chan. Algorithms for finding global minimizers of image segmentation and denoising models. *SIAM Journal on Applied Mathematics*, 66(5):1632–1648, 2006.

[14] S. M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. *CVPR*, 1(2):519–528, 2006.

[15] A. Starti and S. Tubaro. Image-based multiresolution implicit object modeling. *EURASIP*, Vol. 1:1053–1066, 2002.

[16] S. Sullivan and J. Ponce. Automatic model construction, pose estimation and object recognition from photographs using triangular splines. *ICCV*, pages pages 510–516, 1998.

[17] A. Treuille, A. Hertzmann, and S.Seitz. Example-based stereo with general brdfs. *ECCV*, pages pages 457–469, 2004.

[18] R. Vaillant and O. D. Faugeras. Using extremal boundaries for 3d object modelling. *IEEE Trans. Pattern Analysis and Machina Intelligence*, 14:157–173, 1992.

[19] G. Vogiatzis, C. Hernandez, P. H. S. Torr, and R. Cipolla. Multiview stereo via volumetric graph-cuts and occlusion robust photo-consistency. *IEEE Trans. on PAMI*, 29:2241–2246, 2007.

[20] G. Vogiatzis, P. Torr, and R. Cipolla. Multi-view stereo via volumetric graph-cuts. *CVPR*, Vol. 1:391–398, 2005.

[21] A. Yezzi, G. Slabaugh, R. Cipolla, and R. Schafer. A surface evolution approach of probabilistic space carving. *First International Symposium on 3D Data Processing Visualization and Transmission*, pages pages 618–621, 2002.

[22] C. Zach, T. Pock, and H. Bishop. A globally optimal algorithm for robust tv-l1 range image integration. *ICCV*, 2007.