A Globally Optimal Algorithm for Robust TV-L^1 Range Image Integration

Christopher Zach
VRVis Research Center
zach@vrvis.at

Thomas Pock, Horst Bischof
Institute for Computer Graphics and Vision, TU Graz
{pock,bischof}@icg.tu-graz.ac.at

Abstract

Robust integration of range images is an important task for building high-quality 3D models. Since range images, and in particular range maps from stereo vision, may have a substantial amount of outliers, any integration approach aiming at high-quality models needs an increased level of robustness. Additionally, a certain level of regularization is required to obtain smooth surfaces. Computational efficiency and global convergence are further preferable properties. The contribution of this paper is a unified framework to solve all these issues. Our method is based on minimizing an energy functional consisting of a total variation (TV) regularization force and an L^1 data fidelity term. We present a novel and efficient numerical scheme, which combines the duality principle for the TV term with a point-wise optimization step. We demonstrate the superior performance of our algorithm on the well-known Middlebury multi-view database and additionally on real-world multi-view images.

1. Introduction

Volumetric range image integration methods have their origin in the fusion of range images acquired by active sensors [12, 14]. These approaches are specifically targeted to combine several 2.5D range images into one resulting 3D model. Using an intermediate volumetric representations allows the generation of models with arbitrary genus and avoids the numerical difficulties encountered with polygonal techniques (e.g. [30]).

Combining several range images can be performed using general surface-from-point-clouds reconstruction algorithms (e.g. [15, 3, 1, 6]). However, using these very generic methods for range image integration is somewhat futile because of the special structure of the input data and the presence of a substantial amount of outliers.

Early work on volumetric range image integration [12, 14, 32] employs an averaging scheme of 3D distance fields to combine several 2.5D range images. Hence, the obtained surface is basically the minimizer of an underlying energy function with quadratic fidelity terms. As pointed out in [17], simple averaging without further regularization causes inconsistent surfaces due to frequent sign changes of the mean distance field. Therefore, an additional regularization force is required to favor smooth geometry. A commonly used approach is to penalize the surface area of the resulting 3D model, which has been successfully applied in other domains in conjunction with graph-cut algorithms [31, 16, 5] and variational techniques (e.g. [34, 22, 24, 21]). Other recent work on volumetric surface reconstruction [19, 20] directly estimates the corresponding characteristic function from (oriented) point samples. The smoothness of the obtained surface is enforced only implicitly, mainly due to the structure of the underlying numerical problem. The performance of this method in the presence of gross outliers is not demonstrated.

In this work we focus on building high-quality 3D models from a set of range images respectively depth maps. Our main application is the modeling of objects and large outdoor scenery from multiple views. Especially in the latter case, active sensors are of limited use. Therefore, we employ depth maps obtained from small baseline stereo algorithms. There exist several approaches for model reconstruction directly from multiple views based on a photo-consistency criterion [27, 34, 31, 29, 28, 16]. Nevertheless, we utilize intermediate small-baseline stereo results for several reasons: First of all, there is a number of stereo algorithms available, including real-time methods and high-quality approaches. The appropriate computational stereo method can be selected as a black box depending on the particular application. Next, we bypass explicit visibility and delicate robust photo-consistency estimation for voxels required in direct approaches. Finally, the quality of the final model can be approximately evaluated right after the first depth maps are available, which adds an interactive component to the work-flow.

Since stereo is a highly ill-posed problem, one has to deal with several problems. Primarily, the integration procedure must be robust against gross outliers occurring in the range images. This does not only address isolated outliers e.g. at depth discontinuities or occlusions, but also includes

large, but incorrectly matched background regions as well. Furthermore, the resulting 3D mesh should be smooth and preferably watertight without loosing too many sharp features present in the range images. Finally, the corresponding numerical procedure for range image integration should be efficient and yield a globally optimal result.

This paper aims at solving all these issues in a well-founded mathematical framework. Our method is based on minimizing an energy functional incorporating a total variation (TV) regularization term and a L^1 data fidelity term. It is well known that TV minimization leads to minimal surfaces and thus regularizes the resulting 3D model in a proper way [9]. Moreover, we utilize the L^1 norm to measure data fidelity, which is known to be robust against outliers while still being convex. For minimization we develop a novel globally convergent numerical scheme by combining the dual formulation of the TV energy with a point-wise optimization scheme. Since this scheme is embedded in a multiscale approach, high quality 3D models can be computed in the order of a few minutes on a standard desktop computer. The final 3D geometry can be extracted by any implicit surface polygonization technique.

2. Robust Range Image Integration

In this section we present the mathematical framework for robust and globally optimal integration of range images (respectively, depth maps) using a TV-L^1 functional. We presume, that a set of 2D range images, $\{r_i : D_i \to \mathbb{R}\}$ is provided, where $D_i \subseteq \mathbb{R}^2$ is the image domain (usually D_i is simply a rectangle). Moreover the associated alignment information (i.e. the projection matrices) for the images are given. The actual input for our method consists of truncated distance fields $f_i : \Omega \to [-1, 1]$ over a voxel space $\Omega \subseteq \mathbb{R}^3$, which are calculated from the provided range images. We use the convention, that $f_i(\vec{x})$ has positive sign for carved voxels, i.e. points lying in front of the hypothetical surface.

2.1. Generation of Distance Fields

Fig. 1 illustrates the generation of the distance fields f_i. The range images r_i are converted into truncated 3D signed distance fields by computing the directional signed distances along the line-of-sight (similar to [12]). The signed distance from the voxel center to the surface point is weighted by a factor $1/\delta$ and truncated to fit into the interval $[-1, 1]$. Thus, the parameter δ controls the width of the relevant near-surface region. The choice of δ reflects the expected uncertainty of the depth values in z-direction. Using truncated distance fields has the additional advantage, that the memory consumption can be substantially reduced (see Section 3).

We note that the assignment of f_i has varying confidence along the employed lines-of-sight. Voxels in front of the surface induced by the range image have a relatively high confidence, even for a (signed) distance $d(\vec{x}) \gg \delta$. On the contrary, values of f_i have almost no certainty, when $d(\vec{x}) \ll -\delta$, since the corresponding voxels are hidden in the respective view. Therefore, we provide an additional weight volume for each range image, $w_i : \Omega \to \mathbb{R}_0^+$, which serves several purposes. First of all, it supplies confidences for f_i by assigning low values, if $d \ll -\delta$. Moreover, missing values in the range images (e.g. due to view frustum culling or missing depth values) can be handled by assigning zero confidence to voxels along the corresponding line-of-sight.

We employ a simple binary weighting, $w_i(\vec{x}) \in \{0, 1\}$, using $w_i(\vec{x}) = \mathbf{1}_{\{d(\vec{x}) > -\eta\}}$ for $\eta > \delta$. η controls the width of the occluded region behind the surface, which is assumed to be solid, i.e. not carved. Hence, the parameter η indirectly controls the gap size of depth discontinuities, which are still closed (hole-filled). Finally, for all voxels $\vec{x}$ the set of relevant indices $\mathcal{I}(\vec{x}) = \{i : w_i(\vec{x}) > 0\}$ is defined.

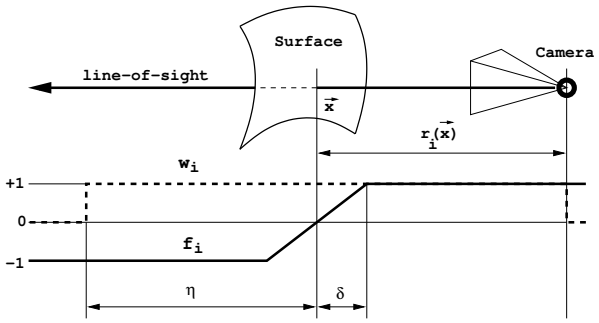


Figure 1. Generation of 3D distance fields from range images.

2.2. A TV-L^1 energy for range image integration

In this section we discuss range image integration using a TV-L^1 energy functional. The goal is to compute a regularized field $u : \Omega \to [-1, 1]$, which simultaneously approximates all input fields f_i. Basically, this is achieved by computing the minimizer of a suitable energy functional. The corresponding surface geometry is obtained as the zero level set of u.

For this purpose, we propose the following TV-L^1 energy functional:[1]

$$E = \int_\Omega \left\{ |\nabla u| + \lambda \sum_{i \in \mathcal{I}(\vec{x})} w_i(\vec{x}) |u - f_i| \right\} d\vec{x} \,, \qquad (1)$$

where λ is free parameter controlling the data fidelity weight. The first term is the total variation which was first proposed by Rudin, Osher and Fatemi (ROF) for nonlinear image denoising [25]. The main property of the TV

[1] We omit the explicit dependency of function variables on the position $\vec{x}$ from now on.

term is that it penalizes the perimeter of the level sets in u, which is in our case exactly the surface area. The second term is the data fidelity and measures the distances of u to all f_i by means of the robust L^1 norm. The primary reason for using a L^1 norm is the increased robustness. When ignoring the spatial regularization (TV term), we see that the minimizer is the point-wise weighted median of the set $\{f_i : i \in \mathcal{I}\}$ [18]. Combining the L^1 data fidelity with convex and edge-preserving regularization forces and the implications on outlier detection is treated in depth in [23]. Using the L^1 norm in conjunction with TV regularization has many other consequences, but we mention only one important qualitative result from [9]: the regularization induced by the TV-L^1 energy is pure geometric, i.e. the obtained minimizer solely depends on the level sets of the supplied input data. Lowering the data fidelity weight λ results in disappearance of isolated, small-scale features instead of increased smoothing of u. Hence, it is expected that isolated clutter is substantially reduced and preferable low-genus iso-surfaces are generated. Finally, we note that the TV-L^1 energy is not strictly convex, therefore there is no unique global minimizer (but all local minima are global minima as well).

By assuming $w_i(\vec{x}) \in \{0, 1\}$, we have $w_i(\vec{x}) = 1$ iff $i \in \mathcal{I}(\vec{x})$ and Eq. 1 reads as:

$$E = \int_\Omega \left\{ |\nabla u| + \lambda \sum_{i \in \mathcal{I}(\vec{x})} |u - f_i| \right\} d\vec{x}. \qquad (2)$$

Although the energy functional in Eq. 2 seems to be simple, it offers some computational difficulties. The main reason is that both the regularization term and the data term are not continuously differentiable. To overcome this, we introduce an auxiliary function v (similar to [2, 4]) and propose to solve the following convex approximation of Eq. 2:

$$E_\theta = \int_\Omega \left\{ |\nabla u| + \frac{1}{2\theta} (u - v)^2 + \sum_{i \in \mathcal{I}(\vec{x})} |v - f_i| \right\} d\vec{x}\,, \qquad (3)$$

where θ is a small constant, such that v is a close approximation of u. E_θ is convex in u and v, therefore an alternating descent approach as described next is universally convergent and returns a global minimizer:

1. For v being fixed, minimize the first two terms in the energy above and solve for u:

$$\min_u \int_\Omega \left\{ |\nabla u| + \frac{1}{2\,\theta} (u - v)^2 \right\} d\vec{x}. \qquad (4)$$

   This is exactly the ROF energy [25, 8].

2. For u being fixed, optimize the last two terms in the energy and solve for v:

$$\min_v \int_\Omega \left\{ \frac{1}{2\,\theta} (u - v)^2 + \lambda \sum_{i \in \mathcal{I}(\vec{x})} |v - f_i| \right\} d\vec{x}. \qquad (5)$$

   This minimization problem can be solved point-wise, since it does not depend on spatial contexts of v.

A solution of the first step was proposed in [8], which uses a dual formulation of Eq. 4 to derive an efficient and globally convergent scheme. Since this algorithm is an essential part of our method, we briefly reproduce the main results from [8]:

Proposition 1 *The solution of Eq. (4) is given by*

$$u = v - \theta\, \mathbf{div}\, \vec{p}\,, \qquad (6)$$

where $\vec{p} = (p^1, p^2, p^3)$ is a vector-valued field and fulfills $\nabla(\theta\, \mathbf{div}\, \vec{p} - v) = |\nabla(\theta\, \mathbf{div}\, \vec{p} - v)|\, \vec{p}$, which can be solved by the following iterative fixed-point scheme:

$$\vec{p}^{\,k+1} = \frac{\vec{p}^{\,k} + \tau \nabla(\mathbf{div}\, \vec{p}^{\,k} - v/\theta)}{1 + \tau |\nabla(\mathbf{div}\, \vec{p}^{\,k} - v/\theta)|}\,, \qquad (7)$$

where $\vec{p}^{\,0} = \vec{0}$ and the time step $\tau \leq 1/6$.

The minimization task in Eq. 5 can be solved point-wise, i.e. on a voxel basis. The minimizer can be obtained by a generalization of the thresholding scheme presented in [2]. Before stating the respective proposition, we start with a few preliminaries: Without loss of generality, we can assume that $f_i \leq f_{i+1}$, since the data fidelity term in Eq. 2 is invariant with respect to permutations of $\mathcal{I}(\vec{x})$. Hence we can postulate a sorted sequence $\{f_i : i \in \mathcal{I}\}$. In order to avoid consideration of special cases, we add $f_0 = -\infty$ and $f_{|\mathcal{I}|+1} = \infty$ to this sequence. Finally, the median of $\{f_i : i \in \mathcal{I}\}$ is denoted by m (which is not affected by the addition of f_0 and $f_{|\mathcal{I}|+1}$).

Proposition 2 *The minimizer of Eq. 5 lies in the interval between u and m and can be obtained by the following procedure: If $v_1 := u - \lambda\theta\,(2k - |\mathcal{I}|) \in (f_k, f_{k+1})$ for some $k \in \{0, \ldots, |\mathcal{I}|\}$, then $v = v_1$. Otherwise,*

$$v = \arg \min_{v_2 \in \{f_i\}} \left((u - v_2)^2 + 2\,\lambda\,\theta \sum_i |v_2 - f_i| \right). \qquad (8)$$

Proof: Without loss of generality assume $u \leq m$. If $v < u$ or $v > m$, then smaller energies can be obtained for $v = u$ and $v = m$, respectively (since the distances to v and m are reduced simultaneously). Hence, $v \in [u, m]$.

Next, note that the energy $(u - v)^2 + 2\,\lambda\,\theta \sum_i |v - f_i|$ is convex and differentiable with respect to v in the interior of

intervals (f_k, f_{k+1}). Consequently, if we assume, that the stationary point v_1 given by

$$v_1 - u + \lambda\theta \sum_i \mathrm{sgn}(v_1 - f_i) = 0, \tag{9}$$

lies in the interval (f_k, f_{k+1}), then

$$\begin{aligned} \sum_i \mathrm{sgn}(v_1 - f_i) &= \#\{i : v_1 > f_i\} - \#\{i : v_1 < f_i\} \\ &= k - (|\mathcal{I}| - k) \\ &= 2\,k - |\mathcal{I}|. \end{aligned} \tag{10}$$

Hence, if the proposed stationary point v_1 stays inside the interval (f_k, f_{k+1}) for some k, then we have found the minimizer of Eq. 5. Otherwise, the minimizer resides on the boundary of one of those intervals, i.e. it can be found among the f_is. $\square$

Finding the optimum for Eq. 5 is much more costly than the simple three-way thresholding scheme sufficient for image denoising ([2], Proposition 4). We provide details for an efficient implementation: first, it is reasonable to distinguish between the two cases $u \leq m$ and $u > m$. In both cases, the search for suitable intervals and v_1 can be restricted, e.g. $k \in \{0, \ldots, |\mathcal{I}|/2\}$, if $u \leq m$. If no k gives a suitable stationary point v_1, then all f_j between u and m are candidates for the minimizer. Evaluating the energies for candidates f_j is costly because of computing the sum $\sum_i |f_j - f_i|$. If $u \leq m$. then we can restrict the candidates to $f_j \leq m$ and we get:

$$\begin{aligned} \sum_i |f_{j+1} - f_i| &= \sum_{i \leq j} (|f_j - f_i| + |f_{j+1} - f_j|) \\ &\quad + \sum_{i \geq j+1} (|f_j - f_i| - |f_{j+1} - f_j|) \\ &= \sum_i |f_j - f_i| - (2j - |\mathcal{I}|)\,|f_{j+1} - f_j|. \end{aligned} \tag{11}$$

An analogous identity can be obtained in the case $u > m$. Hence, $\sum_i |f_j - f_i|$ and the resulting energy in Eq. 5 can be computed efficiently for all candidates f_j, if the f_js are evaluated in ascending (respectively descending) order.

Finally, for voxels $\vec{x}$ with vanishing data fidelity (i.e. empty $\mathcal{I}(\vec{x})$), Eq. 5 reduces to $\min_{v(\vec{x})} \frac{1}{2\theta}(u(\vec{x}) - v(\vec{x}))^2$. Consequently, the update for $v(\vec{x})$ is simply $v(\vec{x}) = u(\vec{x})$.

3. Implementation

3.1. Depth Images From Multi-View Stereo

We utilize multi-view stereo methods to obtain the initial depth maps used for range image integration. We do not employ high-quality and sophisticated dense depth estimation methods for two reasons: first of all, even simple and purely local range image integration methods work well on high-quality and clean depth maps, hence such input data cannot demonstrate the capabilities of our proposed approach. Secondly, high-quality methods for dense depth estimation are typically time and memory consuming and do not fit well into our targeted high-performance 3D modeling pipeline. Hence, the depth maps are generated by simple and efficient methods. We use GPU-accelerated space-sweep and dynamic programming methods (see [33, 11, 35, 36]) to obtain the depth maps used as input for our range image integration approach.

3.2. Compressed Distance Volumes

The range images are converted to truncated distance volumes $f_i \in [-1, 1]$ and $w_i \in \{0, 1\}$ as described in Section 2.1. This step can be accelerated by graphics hardware. Since $w_i \in \{0, 1\}$, it is sufficient to store f_i and to indicate $w_i(\vec{x}) = 0$ by a dedicated value of f_i. Using the direct volumetric representation for f_i is not feasible, since even a moderately sized dataset (e.g. the "Temple" dataset, see Section 4) with 47 range images requires $\approx$ 1.7 GB at $200 \times 300 \times 160$ voxel resolution (using float components). This data can be easily compressed using run-length coding for the data volume f_i associated with every voxel. It is sufficient to compress each voxel individually, which allows a simple and fast decompression scheme to be applied at sufficient compression rates, e.g. the range data for the above-mentioned dataset reduces to 117 MB. Using better compression schemes is currently of limited use, since the memory footprint is dominated by u, v and $\vec{p}$.

3.3. Multi-Scale Approach

The method converges very rapidly near voxels with definite data, but it is a slow process in regions with missing data, where the final value of u is assigned by a diffusion procedure. In order to speed up the convergence in those regions, we employ a multi-scale (coarse-to-fine) approach using a volumetric pyramid. The solution u^{2h} found after a fixed number of iterations is upsampled to u^h at the next level, and the procedure is resumed. The initial v^h at the next level is set to u^h, and $\vec{p}^{\,h}$ can be initialized with $\vec{0}$, since the fixed-point scheme in Eq. 7 is very fast. Note, that the multi-scale approach is only used to accelerate the convergence, but not to avoid local minima, since the proposed method is globally convergent.

4. Results

We evaluate our approach on several datasets, which are briefly characterized in Table 1. This table specifies the dataset names, image and depth resolutions, the resolution of the voxel space and the run-time of the different steps to create the dense geometric model from the calibrated and

Dataset	Images	Image res.	Voxels	Depth estimation	Conv.	Integration	Total
Temple	47	$640\times480\times400$	$200\times300\times160$	1m30s (WTA-SAD-3x3)	1m4s	3m37s	6m11s
Dino	48	$640\times480\times400$	$200\times240\times200$	1m26s (WTA-SAD-3x3)	1m12s	3m50s	6m28s
Statue	40	$512\times768\times400$	$180\times300\times180$	3m16s (WTA-NCC-5x5)	1m11s	4m20s	8m47s
Facades	40	$424\times568\times800$	$420\times200\times360$	5m40s (SO-NCC-5x5)	1m58s	6m25s	14m3s

Table 1. The illustrated datasets and their characteristic information.

(a) Depth image (47 total) (b) Level 2, 15532 triangles (c) Level 1, 67442 triangles (d) Level 0, 394502 triangles, front view (e) Level 0, back view

Figure 2. Selected depth image and final meshes at different pyramid levels for the "Temple" dataset.

registered input images. The 5th column, depth estimation, states the time for (GPU-accelerated) dense depth estimation and the utilized method. WTA is short for winner-takes-all depth extraction and SO indicates scanline optimization. The correlation score is either a sum of absolute differences (SAD) or a normalized correlation coefficient (NCC) using the specified aggregation window. The 6th column specifies the time required to convert the depth maps into compressed distance volumes. The 7th column indicates the time to optimize the proposed energy functional (Eq. 3). All timings were obtained on a desktop PC equipped with a 3.4 GHz Pentium4 processor, 2 GB RAM and a NVidia GeForce 7800 GS graphics card. In all experiments, the number of pyramid levels is three and 100 iterations are performed on each level. λ is set to 0.1 for the first two datasets and assigned to 0.3 for the other ones. θ is fixed to 0.02. The width δ to determine the relevant near-surface region is set to 1% of the diameter of the reconstructed volume, and $\eta = 3\,\delta$. The resolution of the voxel space is chosen, such that the voxels are approximately cubes in order to avoid anisotropic axes.

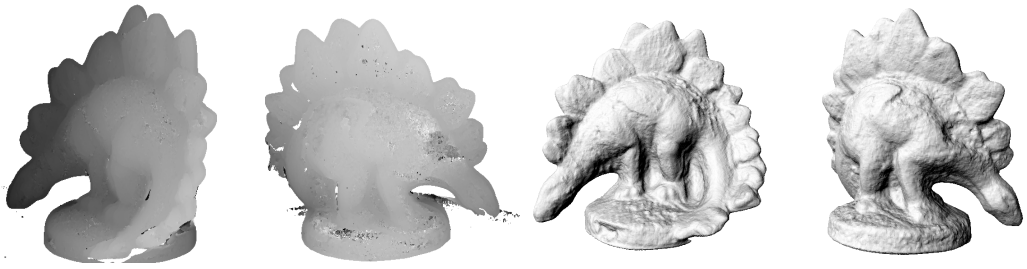
The first two datasets, "Temple" and "Dino", are the medium sized datasets provided for benchmarking multi-view reconstruction methods [26] (see also `vision.middlebury.edu/mview/`). These images are acquired in a controlled indoor environment, hence a simple and efficient SAD matching cost is sufficient. The dark background pixels of the captured scene are removed by thresholding the intensity values. Pixel with a brightness value of at least 10 are considered to be foreground/object pixels. Figure 2(a) displays one resulting depth map for the "Temple" dataset. Intermediate results of our multi-scale approach are depicted in Figure 2(b) and (c), and the front and back view of the final model can be seen in Figure 2(d) and (e).

The results for the "Dino" dataset are illustrated in Figure 3. The depth image in Figure 3(a) appears quite clean, whereas incorrect values in the "neck" region are clearly visible in the depth map displayed in Figure 3(b). The concavity visible in the second view on the mesh (Figure 3(d)) indicates a challenging region, where many multi-view stereo methods have difficulties (e.g. [13, 29, 28]).

The quantitative evaluation comparing our results with the laser-scanned ground truth confirms the convincing visual impression: according to the main evaluation table[2], the accuracy values for the "Temple" and "Dino" meshes are 0.58mm and 0.67mm, respectively. More remarkably, the completeness measures are 99.0% for the "Temple" result and 98.0% for the "Dino" mesh. These numbers and the observed timing results place our proposed approach among the most efficient and high quality methods. Of course, these figures will vary if a different dense depth estimation method is employed.

We provide additional results for two own datasets. The first "Statue" dataset is a sequence of images capturing an indoor statue. One selected source view is shown in Figure 4(a). Since the brightness varies between the images, we employ the normalized correlation coefficient as matching score to compensate for brightness changes. No background segmentation is performed, hence the depth images (see Figure 4(b)) contain substantial noise in background

[2] see [26] for the exact evaluation methodology.



(a) Depth image #1 (b) Depth image #2 (c) Mesh view #1 (d) Mesh view #2

Figure 3. Selected depth images and the final mesh (379958 triangles) for the "Dino" dataset.

regions. Since this noise is largely inconsistent in multiple views, the final integrated model (Figures 4(c) and (d)) is very clean. Parts of the pedestal are missing due to depth outliers induced by specular reflections.

(a) One source view (b) Depth image (c) Front view (d) Back view

Figure 4. The "Statue" dataset (consisting of 40 source views). The final mesh has 230460 triangles.

The finally presented dataset comprises a sequence of facade images used for terrestrial city modeling (Figures 5(a) and (c)). We employ a fast dynamic programming approach for depth estimation to obtain better results in textureless facade regions (Figures 5(b) and (d)), which still have incorrect matches e.g. at mirroring display windows. Figure 5(e) displays the mesh generated by our proposed integration method.

5. Discussion

This section briefly discusses the relationship of our approach with pure binary image and shape denoising, and suggests the integration of additional knowledge into our framework for range image fusion using weighted total variation.

5.1. Distance Field/Shape Denoising

It is tempting to ask, whether (robust) averaging of distance fields near the hypothetical surface is strictly necessary, or if pure binary input fields $f_i \in \{-1, 1\}$ are sufficient, where $f_i(\vec{x}) = 1$ indicates carved voxels according to the range image r_i. Such an approach coincides with selecting the width $\delta \to 0$. The TV-L^1 energy in Eq. 2 simplifies to

$$E = \int_\Omega \Big\{ |\nabla u| \; + \; \lambda N^+(\vec{x}) |u(\vec{x}) - 1| \; + \; \lambda N^-(\vec{x}) |u(\vec{x}) + 1| \Big\} \, d\vec{x}, \tag{12}$$

where $N^+(\vec{x})$ is the number of range images voting for a carved voxel, i.e. $N^+(\vec{x}) = |\{i : d_i(\vec{x}) \geq 0\}|$. $N^-(\vec{x})$ is the number of range images confidently voting for an occluded voxel, namely $N^-(\vec{x}) = |\{i : d_i(\vec{x}) \in (0, -\eta)\}|$.

The minimizer for the energy in Eq. 12 can be again found by an alternating optimization procedure as described in the previous section. It is easy to see, that the solution to the intermediate point-wise minimization step

$$\min_v \left\{ \frac{1}{2\theta}(u - v)^2 + \lambda \left(N^+ |v - 1| + N^- |v + 1| \right) \right\} \tag{13}$$

is now given by

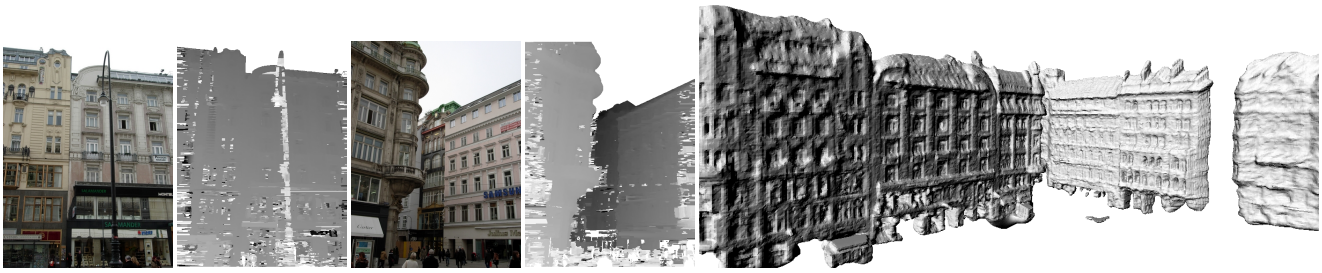
$$v = \max(-1, \min(1, u + \lambda\theta(N^+ - N^-))). \tag{14}$$

Of course, this scheme is more efficient than the procedure outlined in Proposition 2, and the overall computing time is reduced to about 60% in our implementation. However, this approach is very vulnerable to aliasing artefacts in practice, which are clearly visible in Figure 6(a). An analysis for the case of pure binary input fields f_i in the spirit of [10] still needs to be done.

5.2. Weighted Total Variation

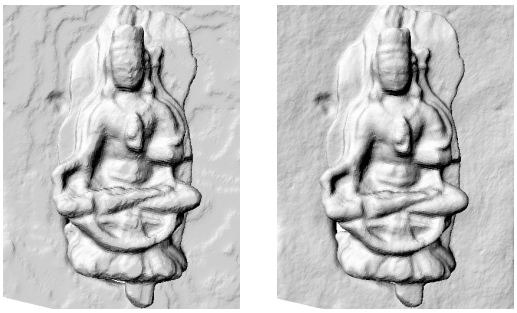
The homogeneous total variation regularization can be replaced by a *weighted* TV-regularization [4], which enables an efficient solution procedure for the geodesic active contour model [7]. The isotropic TV-L^1 energy functional in Eq. 1 can be extended to incorporate a weighted TV-norm:

$$E_g = \int_\Omega \left\{ g(\vec{x}) \, |\nabla u| + \lambda \sum_{i \in \mathcal{I}(\vec{x})} w_i(\vec{x}) |u - f_i| \right\} d\vec{x}, \tag{15}$$



(a) View #1 (b) Depth #1 (c) View #2 (d) Depth #2 (e) Mesh view (688280 triangles)

Figure 5. The "Facades" dataset.



(a) $f_i \in \{-1, 1\}$, $\delta = 0$ (b) $f_i \in [-1, 1]$, $\delta > 0$

Figure 6. Aliasing artefacts occurring in case of purely binary input fields, $f_i \in \{-1, 1\}$ (left image). No artefacts occur, if the width δ of the near-surface region is chosen appropriately (right image).

where $g : \Omega \rightarrow [0, 1]$ is a weighting function. The replacement of the homogeneous TV-norm by a weighted one requires only minimal modifications of the solution procedure, i.e. the fixed point approach to minimize the ROF-energy (recall Eq. 7) is now [4]:

$$\vec{p}^{\,k+1} = \frac{\vec{p}^{\,k} + \tau \nabla(\mathbf{div}\, \vec{p}^{\,k} - v/\theta)}{1 + \frac{\tau}{g(\vec{x})} |\nabla(\mathbf{div}\, \vec{p}^{\,k} - v/\theta)|} \,, \qquad (16)$$

In the range image integration setting, the geodesic model can be used to incorporate sparse geometric data, e.g. 3D points or lines, in addition to the range images. In this case, $g(\vec{x})$ is the (scaled and truncated) unsigned distance to the provided geometric features. As a result, the final iso-surface is more likely to pass through or resides close to the provided features. Note that using *merely* 3D features without input range images has a degenerated solution ($u \equiv 0$). Restricting the solution to stay inside some region of interest (e.g. crust voxels in [17]) offers a solution in this case.

Instead of using geometric features, $g(\vec{x})$ can be based on the photo-consistency of a voxel $\vec{x}$, thereby adopting voxel-coloring principles (e.g. [27]) into the range image integration.

6. Conclusion

We presented a novel and efficient method for robust volumetric integration of 2.5D range images based on a suitable TV-L^1 energy. Our proposed method is globally convergent and returns a (not necessarily unique) global optimum. Visual assessment and the quantitative evaluation applied on the Middlebury datasets indicate the excellent performance of our approach, even when very noisy depth maps are provided as input.

Future work needs in particular to address the scalability of the proposed method, since our current implementation uses 6 additional volumetric data structures (u, v, $\vec{p}$ and the median), hence the maximum resolution of the voxel space is limited. A smaller memory footprint can be achieved e.g. by restricting the computation to a "narrow band" close to the hypothetical surface.

Acknowledgments

Part of this work has been done in the VRVis research center (www.vrvis.at), which is partly funded by the Austrian government research program Kplus. Partial support by the VM-GPU Project No. 813396, financed by the Austrian Research Promotion Agency (http://www.ffg.at), is acknowledged.

References

[1] N. Amenta, S. Choi, and R. Kolluri. The power crust. In *Proceedings of 6th ACM Symposium on Solid Modeling*, pages 249–260, 2001.

[2] J.-F. Aujol, G. Gilboa, T. Chan, and S. Osher. Structure-texture image decomposition–modeling, algorithms, and parameter selection. *Int. Journal of Computer Vision*, 67(1):111–136, 2006.

[3] J.-D. Boissonnat and F. Cazals. Smooth surface reconstruction via natural neighbour interpolation of distance functions. In *Symposium on Computational Geometry*, pages 223–232, 2000.

[4] X. Bresson, S. Esedoglu, P. Vandergheynst, J. Thiran, and S. Osher. Fast Global Minimization of the Active Con-

tour/Snake Model. *Journal of Mathematical Imaging and Vision*, 2007.

[5] N. Campbell, G. Vogiatzis, C. Hernandez, and R. Cipolla. Automatic 3D object segmentation in multiple views using volumetric graph-cuts. In *British Machine Vision Conference*, 2007.

[6] J. C. Carr, R. K. Beatson, J. B. Cherrie, T. J. Mitchell, W. R. Fright, B. C. McCallum, and T. R. Evans. Reconstruction and representation of 3D objects with radial basis functions. In *Proceedings of SIGGRAPH 2001*, pages 67–76, 2001.

[7] V. Caselles, R. Kimmel, and G. Sapiro. Geodesic active contours. *Int. Journal of Computer Vision*, 22(1):61–79, 1997.

[8] A. Chambolle. An algorithm for total variation minimization and applications. *Journal of Mathematical Imaging and Vision*, 20(1–2):89–97, 2004.

[9] T. F. Chan and S. Esedoglu. Aspects of total variation regularized L^1 function approximation. *SIAM Journal on Applied Mathematics*, 65(5):1817–1837, 2004.

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

[11] N. Cornelis and L. Van Gool. Real-time connectivity constrained depth map computation using programmable graphics hardware. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1099–1104, 2005.

[12] B. Curless and M. Levoy. A volumetric method for building complex models from range images. In *Proceedings of SIGGRAPH '96*, pages 303–312, 1996.

[13] M. Goesele, B. Curless, and S. Seitz. Multi-view stereo revisited. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2402–2409, 2006.

[14] A. Hilton, A. J. Stoddart, J. Illingworth, and T. Windeatt. Reliable surface reconstruction from multiple range images. In *European Conference on Computer Vision (ECCV)*, pages 117–126, 1996.

[15] H. Hoppe, T. DeRose, T. Duchamp, and W. S. J. McDonald. Surface reconstruction from unorganized points. In *Proceedings of SIGGRAPH '92*, pages 71–78, 1992.

[16] A. Hornung and L. Kobbelt. Hierarchical volumetric multi-view stereo reconstruction of manifold surfaces based on dual graph embedding. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 503–510, 2006.

[17] A. Hornung and L. Kobbelt. Robust reconstruction of watertight 3D models from non-uniformly sampled point clouds without normal information. In *Eurographics Symposium on Geometry Processing*, pages 41–50, 2006.

[18] P. J. Huber. *Robust Statistics*. Wiley, New York, 1981.

[19] M. Kazhdan. Reconstruction of solid models from oriented point sets. In *Symposium on Geometry Processing*, pages 73–82, 2005.

[20] M. Kazhdan, M. Bolitho, and H. Hoppe. Symposium on geometry processing. In *Symposium on Geometry Processing*, pages 61–70, 2006.

[21] K. Kolev, M. Klodt, T. Brox, S. Esedoglu, and D. Cremers. Continuous global optimization in multiview 3d reconstruction. In *International Conference on Energy Minimization Methods in Computer Vision and Pattern Recognition*, 2007.

[22] M. Lhuillier and L. Quan. Surface reconstruction by integrating 3d and 2d data of multiple views. In *IEEE International Conference on Computer Vision (ICCV)*, pages 1313–1320, 2003.

[23] M. Nikolova. A variational approach to remove outliers and impulse noise. *Journal of Mathematical Imaging and Vision*, 20(1–2):99–120, 2004.

[24] J.-P. Pons, R. Keriven, and O. Faugeras. Modelling dynamic scenes by registering multi-view image sequences. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 822–827, 2005.

[25] L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60:259–268, 1992.

[26] S. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 519–526, 2006.

[27] S. Seitz and C. Dyer. Photorealistic scene reconstruction by voxel coloring. *Int. Journal of Computer Vision*, 35(2):151–173, 1999.

[28] C. Strecha, R. Fransens, and L. V. Gool. Combined depth and outlier estimation in multi-view stereo. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2394–2401, 2006.

[29] S. Tran and L. Davis. 3d surface reconstruction using graph cuts with surface constraints. In *European Conference on Computer Vision (ECCV)*, pages 219–231, 2006.

[30] G. Turk and M. Levoy. Zippered polygon meshes from range images. In *Proceedings of SIGGRAPH '94*, pages 311–318, 1994.

[31] G. Vogiatzis, P. Torr, and R. Cipolla. Multi-view stereo via volumetric graph-cuts. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 391–398, 2005.

[32] M. Wheeler, Y. Sato, and K. Ikeuchi. Consensus surfaces for modeling 3d objects from multiple range images. In *IEEE International Conference on Computer Vision (ICCV)*, pages 917 – 924, 1998.

[33] R. Yang and M. Pollefeys. Multi-resolution real-time stereo on commodity graphics hardware. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 211–217, 2003.

[34] A. Yezzi and S. Soatto. Stereoscopic segmentation. *Int. Journal of Computer Vision*, 53(1):31–43, 2003.

[35] C. Zach, M. Sormann, and K. Karner. High-performance multi-view reconstruction. In *International Symposium on 3D Data Processing, Visualization and Transmission (3DPVT)*, 2006.

[36] C. Zach, M. Sormann, and K. Karner. Scanline optimization for stereo on graphics hardware. In *International Symposium on 3D Data Processing, Visualization and Transmission (3DPVT)*, 2006.