Fast, Approximate Piecewise-Planar Modeling Based on Sparse Structure-from-Motion and Superpixels

András Bódis-Szomorú Hayko Riemenschneider Luc Van Gool
Computer Vision Lab, ETH Zurich, Switzerland
{bodis,hayko,vangool}@vision.ee.ethz.ch

Abstract

State-of-the-art Multi-View Stereo (MVS) algorithms deliver dense depth maps or complex meshes with very high detail, and redundancy over regular surfaces. In turn, our interest lies in an approximate, but light-weight method that is better to consider for large-scale applications, such as urban scene reconstruction from ground-based images. We present a novel approach for producing dense reconstructions from multiple images and from the underlying sparse Structure-from-Motion (SfM) data in an efficient way. To overcome the problem of SfM sparsity and textureless areas, we assume piecewise planarity of man-made scenes and exploit both sparse visibility and a fast over-segmentation of the images. Reconstruction is formulated as an energy-driven, multi-view plane assignment problem, which we solve jointly over superpixels from all views while avoiding expensive photoconsistency computations. The resulting planar primitives – defined by detailed superpixel boundaries – are computed in about 10 seconds per image.

1. Introduction

Automatic 3D reconstruction of urban scenes is a difficult and long-researched problem [19]. Our focus of interest is automatic reconstruction of man-made environments from street-level photographs. Based on developments in feature detection, description, and matching during the last decade, state-of-the-art Structure-from-Motion (SfM) pipelines are now capable to compute a sparse metric reconstruction of large-scale scenes [2, 9, 6, 20]. Relying on the camera models provided by SfM, various Multi-View Stereo (MVS) algorithms have been proposed that produce very dense surface meshes or point clouds [28, 12, 16] which are photoconsistent across multiple views.

Besides occlusions, non-diffuse surfaces, and repetitive patterns, a major challenge of MVS reconstruction is the lack of visual cues on textureless surfaces. This often causes holes or leads to noisy structures in these regions in the resulting depth maps or surface models. Such visual artifacts are particularly disturbing on regular surfaces, e.g. planar parts [22, 18, 13], which are predominant in man-made scenes. These effects are also observable on results produced in city scenarios, e.g. [20, 18]. MVS methods that can suppress these artifacts via strong regularization tend to oversmooth the surface. Moreover, dense MVS delivers a very redundant sampling over these regular parts, e.g. [16].

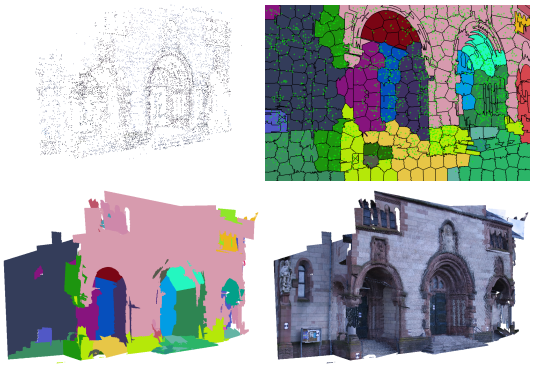

Figure 1. Our method joins sparse SfM with superpixels to obtain a light-weight, piecewise-planar, multi-view surface reconstruction.

MVS methods usually require intense computations, mainly photoconsistency calculations over many views and many depth hypotheses. Thus, they have poor scalability in runtime and storage space, which renders them less appealing for use in large-scale street-level urban reconstruction[1].

Motivated by these drawbacks, we propose a novel method that is capable of computing a piecewise planar reconstruction from street-level photographs and that relies only on sparse SfM data while avoiding intense computational steps. The final 3D primitives are delineated by projecting detailed superpixel boundaries from the images.

Our approach is not supposed to challenge the accuracy of dense MVS. To the contrary, it provides a good basis

[1]Companies like Apple, Google, Blom, Acute3D etc. have recently made large-scale city models automatically by dense MVS methods. These are typically based on aerial images, and deliver no street-level detail.

for applications that require less detail, but higher efficiency and better scalability. Our main contributions are:

- the combination of sparse SfM and superpixels in a multi-view MRF for obtaining a dense, approximate, light-weight, piecewise-planar surface reconstruction,
- an energy formulation that enables an efficient assignment of superpixels from all views to planes, based on a 3D fitting score, sparse visibility constraints, pairwise color similarity and image gradients,
- a plane quality criterion that measures the sensitivity of 3D polygons w.r.t. noise in the SfM points.

2. Related work

We categorize related work into four major groups: (1) segmentation-based stereo, (2) dense MVS, (3) segmentation of depth maps or dense point clouds, (4) methods relying on sparse SfM only and extracting planar primitives.

The first group lies in the field of dense two-view depth estimation. Several top-performing algorithms [30, 34] on the Middlebury challenge [21] exploit an image oversegmentation and enforce pairwise disparity consistency in a Markov-Random-Field (MRF). These methods effectively propagate depth information from textured to ambiguous textureless areas [21, 4]. They estimate initial disparities via standard window-based stereo and group them per image segment. By lifting the domain from the pixel to the superpixel level, both the computational complexity and the susceptibility to noise are reduced [35, 25].

The second group lies in the field of Multi-View Stereo (MVS). The classic approach extends pairwise stereo by linking and fusing stereo depth maps. As an example, [15] incorporates piecewise planar patch priors into the process.

In turn, [14, 12] represent the surface by a set of oriented photoconsistent 3D patches, and reconstruct them directly by exploiting multiple views. They iterate between patch optimization, local 3D expansion and filtering. [26] also uses multi-view consistency filtering and presents efficient point-cloud reconstructions of large-scale city scenes. These approaches are relatively simple yet effective. The results are quasi-dense point clouds, which contain noise and have to be further processed to obtain a surface model.

Plane-sweep stereo sweeps planes along a few principal directions (obtained from SfM) to generate disparity hypotheses. The drawbacks are that surfaces are assumed to be orthogonal to the sweeping directions and the scene is usually restricted to Manhattan-world [20, 11]. Finding non-orthogonal dominant directions requires multi-structure fitting, or normal estimation and clustering, which tend to be unstable from sparse SfM data. [18] additionally exploits superpixels to better cope with textureless areas, but they rely on the Manhattan assumption, and compute photoconsistency per superpixel over all plane hypotheses to then merge the resulted depth maps. In turn, we allow for many plane orientations, and do not rely on photoconsistency.

The third group attacks the problem of redundancy in pixel-wise depth maps or dense point clouds by detecting planes [5] or spheres, cones, cylinders and tori [17]. These works perform robust multi-structure fitting directly to the dense 3D data. [13] additionally exploits the images for finding planar and non-planar regions in dense depth maps. They assign image pixels to a discrete set of pre-decided plane primitives. However, their MRF optimization operates independently in each image for efficiency.

Finally, the fourth group is the most related to ours. These works generate planar hypotheses from sparse SfM point clouds, either by direct plane fitting [23], or by first reconstructing line segments and vanishing directions, and then using the images to detect support regions or to fine-tune the plane primitives [31, 23]. [10] iterates between photoconsistent support region growing and updating the plane parameters. However, the method has difficulties with textureless regions. [22] copes with textureless areas by exploiting a global MRF with multi-view constraints, yet they assign individual pixels to the plane hypotheses, which is time-consuming and less robust. [33] joins SfM points with MRF optimization over superpixels but only for the purpose of semantic scene segmentation.

In contrast to existing methods, ours does not require dense point clouds or depth maps, exploits image oversegmentation to simplify robust multi-structure fitting, allows for any number of plane directions (no Manhattan assumption) and even detects minor planes. The depth estimation is truly multi-view as we use a global MRF optimization over superpixels in all views, treating all views equally.

3. The proposed method

Our method starts by estimating the underlying camera models, the sparse structure and its visibility by using an existing SfM tool, e.g. [32, 27, 23]. Then finding the primitives involves solving three joint problems:

- *Fitting problem*: compute the continous plane parameters of each primitive,
- *Segmentation*: find the inlier SfM points and image support regions for each primitive,
- *Visibility reasoning or occlusion problem*: determine which region of a primitive is visible in which image.

These problems are inter-related, e.g. fitting requires the support region of the particular primitive, support region segmentation in the images (or inlier-outlier separation in

3D) requires the model parameters and visibility, while visibility reasoning over a surface relies on known views and known surface.

3.1. Superpixels and Plane Hypotheses

In the first step, we assume that the scene is composed of a set of planar primitives, and we aim to generate plane hypotheses that explain the point cloud. Similar in spirit to slanted-plane stereo [30, 34], our initial planes are restricted to local neighborhoods defined by superpixels. This simplifies the three-fold problem of fitting, support region search and visibility reasoning. This approach can capture multiple planes, while not suffering from the difficulties of direct fitting of global planes [29], or of local plane growing either in the image [10] or in the sparse point cloud [5]. In our experience, local plane growing [5] gives decent results over dense point clouds but fails to capture the right planes in a sparse SfM point cloud.

The components of our hypothesis generation method are image over-segmentation, robust local plane fitting, quality filtering, and plane merging.

3.1.1 Local plane fitting

Assume each of the M images $\mathcal{I}^m$ of the scene is preliminarily partitioned into a number of superpixels, e.g. by any method in [1]. Each segment is a 4-connected set of pixels. Let S denote the number of superpixels over all views, and $\mathcal{S} = \{s_1, s_2, \ldots, s_S\}$ the set of superpixels s_i irrespective of their image $\mathcal{I}(s_i)$.

Using RANSAC [8], we fit a plane π_i robustly to the set of 3D points $\mathcal{P}_i$ in each segment s_i, with inlier threshold τ. To discriminate between random hypotheses having the same number of inliers, we replace the inlier count scoring of RANSAC by the sum of the weighted distances of *all* points $p_k \in \mathcal{P}_i$ from each plane hypothesis π:

$$\mathcal{C}(\mathcal{P}_i, \pi) = \sum_{p_k \in \mathcal{P}_i} \exp\left(-\frac{1}{2\tau^2} d^2(p_k, \pi)\right), \qquad (1)$$

where $d(\cdot, \cdot)$ is the point-to-plane distance. This relaxed score is a more robust measure than simple inlier counting.

Once the best plane hypothesis π_i is found for superpixel s_i, we do a final refitting to the inlier set $\mathcal{Q}_i \subseteq \mathcal{P}_i$. Although uniform weighting of the 3D points is used here, we note that incorporating triangulation uncertainties into the weights could further improve the quality of plane fitting.

3.1.2 Stability-based plane filtering

To filter out poor plane hypotheses, we propose a powerful stability measure using Monte-Carlo (MC) perturbation analysis. While simple residual or point scatter analysis (by PCA) is solely based on the arrangement, our method also takes point uncertainties and viewpoints into consideration. It can discover both plane hypotheses π_i with unstable point support and planes π_i that are seen in very sharp angles across their supporting superpixel s_i. First, each superpixel s_i is represented by the convex hull of its boundary pixels in the image. Second, a coarse 3D reconstruction of the superpixel s_i is obtained by projecting the 2D vertices of the hull to the plane π_i. The aim is to quantify the 3D sensitivity of this superpixel reconstruction to the uncertainty in the plane fit π_i. We perturb all inlier points in $\mathcal{Q}_i$ to a distance τ in a random 3D direction. Note that it may be beneficial to replace this with a perturbation according to individual point uncertainties. The plane is re-fitted to the perturbed inliers and the superpixel hull is reconstructed on the new plane. We repeat this procedure N_{MC} times to obtain a perturbed set of 3D superpixel hulls (Figure 2). The sensitivity of the 3D hull is measured by the mean 3D vertex displacement, denoted by h_i. We measure the quality of a plane fit π_i by

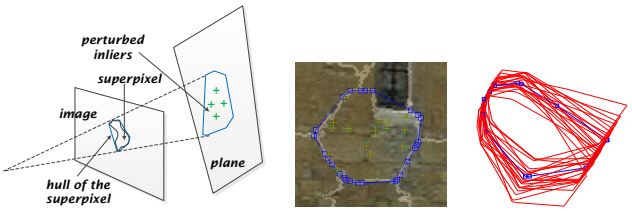

Figure 2. Convex hull of a superpixel and inlier points of the local plane fit. 3D reconstructions of the hull (right) obtained by perturbing the inlier point set in 20 Monte-Carlo experiments.

$$q_i = \exp(-h_i/\tau) \in [0, 1], \qquad (2)$$

where τ is the inlier threshold. Planes π_i with any vertex of the hull projected behind the camera in any of the MC experiments are considered degenerate by enforcing $q_i = 0$. We than remove planes with qualities below a threshold q_{th}.

In practice, even with a low number of MC experiments, the method tends to capture unstable fits, including most view-dependent degeneracies that are missed by PCA.

3.1.3 Plane merging

We significantly compress the set of remaining planes from all views by a simple global merging procedure: a plane is only accepted as hypothesis, if its inlier set is not fully explained by any already accepted plane. The result is a set of $L \ll S$ plane hypotheses $\Pi = \{\pi_1, \pi_2, \ldots, \pi_L\}$.

3.2. Energy-Driven Multi-View Segmentation

Given the initial planes Π, which form an incomplete and redundant, but relatively accurate approximation of the true surface, the problem simplifies to a multi-label segmentation problem, where one associates a plane π_l, represented by a label $l \in \{1, 2, \ldots, L\}$, to each superpixel $s_i \in \mathcal{S}$.

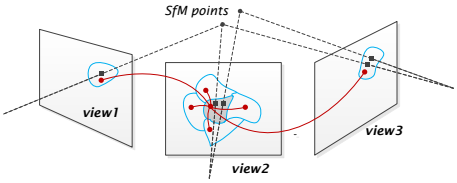

Figure 3. Graph of the multi-view plane segmentation problem. The red lines depict all adjacencies of the grey superpixel.

To solve this, consider the graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with superpixels from all views as vertices ($\mathcal{V} = \mathcal{S}$) and with the set of edges $\mathcal{E} = \mathcal{E}^w \cup \mathcal{E}^b$, where $\mathcal{E}^w$ is the subset of edges *within* the same view and $\mathcal{E}^b$ *between* views ($\mathcal{E}^w \cap \mathcal{E}^b = \emptyset$). We connect two vertices either if the two corresponding superpixels s_i, s_j are 4-connected neighbors in the same image, or if they lie in different images, but they contain *at least one* point match corresponding to a single SfM point, i.e. if $\mathcal{P}_i \cap \mathcal{P}_j \neq \emptyset$. This construction is illustrated in Figure 3. We formulate the cost of a given labelling $\mathcal{L} = (l_1, l_2, \ldots, l_S)$ – l_i being the plane label assigned to superpixel s_i – as

$$E(\mathcal{L}) = \sum_{i=1}^{S} D_i(l_i) + \sum_{(i,j)\in\mathcal{E}^w} V_{ij}^w(l_i, l_j) + \sum_{(i,j)\in\mathcal{E}^b} V_{ij}^b(l_i, l_j), \quad (3)$$

where $D_i(l_i)$ is the unary cost of assigning label l_i to s_i, and V_{ij}^w and V_{ij}^b are the pairwise costs of assigning labels l_i and l_j to segments s_i and s_j, provided they are in the same view, or in different views, respectively.

3.2.1 Unary terms

Each unary term can be written as

$$D_i(l_i) = D_i^{fit}(l_i) + D_i^{rays}(l_i) + D_i^{angle}(l_i). \quad (4)$$

The first subterm D_i^{fit} is a robust measure of how well each plane π_{l_i} with label l_i explains the 3D points detected in superpixel s_i. It is formulated as

$$D_i^{fit}(l_i) = \exp\left\{-\mathcal{C}(\mathcal{P}_i, \pi_{l_i})/\sigma_{fit}\right\}. \quad (5)$$

$\mathcal{C}(\mathcal{P}_i, \pi_{l_i})$ is the weighted inlier count in Eq. (1). It weights each 3D point $p_k \in \mathcal{P}_i$ in superpixel s_i by its distance from the plane hypothesis. We set $\sigma_{fit} = 3$ in all experiments, i.e. with 3 good inliers, the full penalty is attenuated to 37%.

The second unary subterm $D_i^{rays}(l_i)$ penalizes the assignment of plane π_{l_i} to superpixel s_i if there are free-space violations on the plane in the region observed by the superpixel (see Figure 4). Visibility rays are shot from each viewpoint to all 3D points detected in the view, and each ray that intersects the plane π_i in the region visible in superpixel s_i contributes to the penalty cost, provided that the target 3D point of the ray is not inlier to the plane (farther from the plane than τ). If $\mathcal{C}_{rays}(s_i, \pi_{l_i})$ is the number of such rays,

$$D_i^{rays}(l_i) = 1 - \exp\left\{-\mathcal{C}_{rays}(s_i, \pi_{l_i})/\sigma_{rays}\right\}. \quad (6)$$

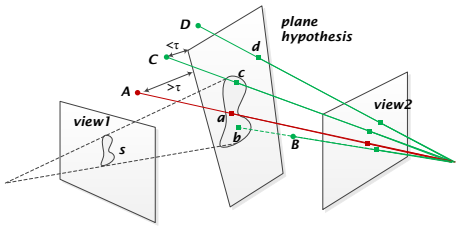

Figure 4. Free-space violations. Only the red ray contributes to the penalty of assigning the superpixel s to the plane, since it goes to an outlier point and intersects the superpixel's plane projection .

Since the number of visibility rays is usually much higher than the number of SfM points, the rays yield a dense distribution of plane intersections. Thus, we relax the penalty by setting $\sigma_{rays} = 5$ in all experiments to make our score robust to incorrect rays shot to outlier SfM points.

The third unary subterm $D_i^{angle}(l_i)$ discourages the assignment of a plane to a superpixel if the plane is seen in a sharp angle α_i through the superpixel. It is defined as

$$D_i^{angle}(l_i) = \begin{cases} \frac{1}{2} + \frac{1}{2}\cos\left\{\frac{\pi}{\Delta}\left(\alpha_i - \frac{\pi}{2}\right)\right\}, & \alpha_i \geq \frac{\pi}{2} - \Delta \\ 0, & \text{otherwise} \end{cases}$$

where $\alpha_i \in [0, \frac{\pi}{2}]$ is the largest angle between the plane normal and the incident viewing rays from the 3D vertices of the superpixel's reconstructed convex hull (Fig. 2). Only angles above $\frac{\pi}{2} - \Delta$ are penalized, where Δ is set to $5°$.

3.2.2 Within-views pairwise terms

The terms V_{ij}^w in Eq. (3) penalize any two neighboring superpixels s_i and s_j within the same image being assigned to different planes. These pairs are connected by an edge in $\mathcal{E}^w$ in the graph representation $\mathcal{G}$. We use the following site-dependent Potts-model.

$$V_{ij}^w = (\alpha C_{ij} + \beta G_{ij}) \cdot \omega_{ij}^w \cdot \mathbb{I}[l_i \neq l_j], \quad (7)$$

where $\mathbb{I}[\cdot]$ is 1 if its argument is true, and is 0 otherwise. C_{ij} is a color term, G_{ij} a gradient term, ω_{ij}^w is an additional site-dependent weight characterizing the adjacency strength of the pair s_i, s_j, while α and β represent the balance with respect to the unary terms $D_i(l_i)$ in Eq. (3). We set $\beta = \alpha$.

The color term C_{ij} increases the penalty for two neighboring superpixels of similar color having different labels.

$$C_{ij} = \exp(-c_{ij}/\sigma_c), \quad (8)$$

where σ_c is a shaping parameter we fix to 0.05, and c_{ij} is the difference between the mean color of neighboring superpixels s_i and s_j, measured as the mean of the absolute differences between individual RGB color components (ranging between 0 and 1). The color term C_{ij} enforces that two superpixels similar in color should observe the same scene

plane, which is desirable when a homogeneous surface is artificially split into multiple superpixels. However, it does not take care of a potential boundary edge observed between them, e.g. as an indication of a crease edge or discontinuity between walls of similar color.

The gradient term enforces the same plane to neighboring superpixels if their shared boundary section observed in the image is weak. It is formulated as

$$G_{ij} = \exp(-g_{ij}^{w}/\sigma_g), \tag{9}$$

where σ_g is a shaping parameter also fixed to 0.05, and g_{ij} is the magnitude of the image gradient along boundary pixels between s_i and s_j. The lower the gradient, the more the superpixels are enforced to observe the same plane. This assumes that edges between superpixels are observed where there is a crease or discontinuity between scene planes. However, it does not distinguish these from texture edges.

We decrease the effect of neighboring superpixels on each other, if they share a shorter boundary. Therefore, ω_{ij}^{w} is an additional weighting in function of the relative length of the shared boundary between s_i and s_j, namely

$$\omega_{ij}^{w} = 1 - \exp(-b_{ij}/\sigma_b), \tag{10}$$

where σ_b is a shaping parameter (0.1 in all experiments) and b_{ij} is the shared boundary length divided by the shorter superpixel circumference. Using ratio instead of pixel count makes the formulation indifferent to the size of superpixels.

3.2.3 Between-views pairwise terms

The terms V_{ij}^{b} in Eq. (3) may penalize any two superpixels s_i and s_j being assigned to different planes, if they share at least one feature match arising from the same SfM point. Such pairs are connected by an edge in $\mathcal{E}^b$ in the graph $\mathcal{G}$. We formulate the between-view penalty as

$$V_{ij}^{b} = \gamma \omega_{ij}^{b} C_{ij} \cdot \mathbb{I}[l_i \neq l_j], \tag{11}$$

where γ is the balance w.r.t. the unary terms $D_i(l_i)$ in Eq. (3). We set γ to α multiplied by the ratio between the number of average intra-view and the average inter-view correspondences per superpixel, to balance out the effect of the two types of adjacencies. C_{ij} is the color term of Eq. (8), and ω_{ij}^{b} is a weight encoding the neighborhood strength for superpixels between views, replacing ω_{ij}^{w} of Eq. (10).

$$\omega_{ij}^{w} = 1 - \exp(-n_{ij}/\sigma_n), \tag{12}$$

where σ_n encodes how strongly two superpixels should be tied in color in function of the number of shared 3D points n_{ij}. The higher the number of shared points the more we take into consideration differences in color in the Potts penalty. We use $\sigma_n = 2$ in all experiments.

3.3. Optimization of the support regions

Finding the global minimum of the energy function in Eq. (3) is NP-hard, but there exist efficient graph-cuts algorithms that have guarantees for the local minimum computed. Since our pairwise terms are regular, we use the α-expansion algorithm [3] for minimization.

As a result, we have one of the plane hypotheses π_{l_i}, $l_i \in \{1, 2, \ldots, L\}$ assigned to each superpixel $s_i \in \mathcal{S}$.

Common plane labels organize superpixels into groups that span across multiple views. This extends the support regions of the initial plane hypotheses to larger sets of SfM points, and provides the superpixel boundaries as natural borders for the support regions. To delineate the support regions per plane, we extract the pixelwise polygonal boundaries of each connected component from the label map per view, project them to the plane to obtain 3D polygons, apply standard polygon simplification, and consider the union of the polygons as the support region. The views complement each other, i.e. different areas of each support region on a plane may come from different views, and scene parts observed in multiple views may not be captured by perfectly matching regions, due to the sparseness of the SfM cloud.

Since labelling enforces a plane label to each superpixel, components with no data support may occur. After a connected component analysis on the labelled graph $\mathcal{G}$, we remove all planes with no point support, as well as the ones with weak support, based on an area ratio criterion.

In our formulation, we do not penalize for depth discontinuities along superpixel boundaries, as this would not allow to efficiently pre-compute all costs. However, our penalties arising at similar neighboring superpixels separated by weak boundaries enforce planarity in such regions.

Feeding all the initially fit planes along with their qualities as label costs to discourage the assignment of poor quality planes to superpixels is possible, but is inefficient and unstable. The immense number of planes before merging, and very similar plane hypotheses result in an ineffective α-expansion. In turn, our plane filtering and merging steps of Section 3.1 effectively reduce the number of hypotheses.

In summary, our graph-based optimization has several advantages. First, it propagates planes from data rich planar regions to weakly supported or empty regions, where initial planes are not available. Second, it reduces the number of planes used to model the scene. Third, it makes the support regions as consistent as possible within and between views and allows to produce dense piecewise planar outputs by projecting the superpixels to their assigned planes.

4. Experiments

We demonstrate our method on several outdoor sequences consisting of around 0.8 MPixel images of landmark and street-side scenes. Herz-Jesu-P8 [24], Merton

College I, III [31] are smaller public datasets, while Pozzoveggiani [7], and our Mirbel dataset are larger datasets with considerable clutter. We use VisualSFM [32] to obtain a sparse SfM reconstruction, and extract around 400-600 SLIC superpixels [1] per image to assure that they are large enough to contain enough points for plane fitting. Fig. 5 gives an overview of our pipeline, Fig. 6 shows results for more datasets, and Table 4 summarizes numerical results.

The only varying parameter between datasets is the threshold τ. It depends on the scale of the reconstruction and on the level-of-detail requirements. It could be easily fixed over datasets if their scales were known, e.g. geolocated. We set the factor between the unary and pairwise terms to $\alpha = 0.1$ in all experiments, and use a plane quality threshold $q_{th} = 0.1$, which is a good trade-off between eliminating bad planes and keeping slanted structure planes, e.g. roofs. Other parameters are fixed as described earlier.

Our plane stability criterion effectively filters out bad plane hypotheses, and the merging step greatly reduces the number of hypotheses by eliminating redundant ones (see Table 4). This allows the Graph-Cuts Optimization (GCO) to work efficiently on the reduced set. Note how the optimization completes the structures by extending support regions from data-rich to weakly supported areas based on image content. The boundaries of the support regions align with superpixel borders. The result is a dense 3D arrangement of polygons approximating the geometry of the scene.

Failure cases are planes extended into sky regions that are strongly contaminated by SfM outliers, drift in plane extrapolation that causes fragmentation of planar regions into multiple segments, and artifacts at plane boundaries due to weakly observable crease edges or discontinuities crossed by superpixels. These artifacts could be removed by skyline detection, by E-M style iterations between plane fitting and segmentation, by superpixel splitting, and by using an additional volumetric optimization, such as in [5, 16].

The overall runtime of our Matlab implementation on a 3.4 GHz i7 CPU is around 10-17 seconds per image, i.e. 7 minutes for Mirbel, 11 for Pozzo, whereas related work reports 40 minutes and more than 2 hours for similar-size datasets [22]. Moreover, we have not explicitely used any parallelization, and 60-80% of the runtime is spent in our fairly slow implementation of superpixel extraction. We expect a large benefit from processing superpixels in parallel (including over-segmentation, plane fitting, plane filtering), as well as calculating individual energy terms in parallel. It is mainly the final GCO that requires joint data. GCO runtime is more affected by the plane hypotheses than the number of superpixels/images (Table 4). Without plane merging, the number of labels grows linearly with the number of images. However, the same scene parts are observed by many views (as also required by SfM), and in urban scenes, real planes (wall, roof) often extend over many views and many buildings. Hence, the merging step is effective and makes the growth sublinear. Further speed-up could be obtained by restricting the planes to building or city blocks, and by only processing a pre-selected set of views (view clustering and view selection).

5. Conclusion

State of the art for Multi-View Stereo (MVS) produces detailed and accurate geometry, but suffers from high runtime, redundant output, and often lacks higher-level knowledge of the geometry. For applications requiring less detail but better scalability, e.g. city reconstruction from ground-level images, we proposed a method that computes a piecewise planar approximation of the scene from sparse data, while exploiting dense multi-view image support.

Our approach combines sparse SfM data with superpixels and solves a dense multi-view segmentation problem in an efficient way by avoiding expensive photoconsistency computations. Our solution extracts both dominant and small planar structures without using the Manhattan assumption, or without the need for clustering normals, which are difficult to precisely compute in sparse point clouds. Our plane quality criterion and merging step eliminates low-quality planes and significantly shrinks the hypothesis set prior to optimization. Our method does not require dense depth maps [13], or dense point clouds [5] as input, efficiently deals with textureless areas, unlike e.g. [10, 11], produces denser results than PMVS [11], and is much faster than, e.g. [22], which is also based on sparse SfM.

Future work will investigate real-time capabilities and the combination of planar and free-form elements into a watertight volumetric reconstruction, as well as experimentation with large-scale city scenes.

Acknowledgement. We thank Julien Weissenberg for his valuable comments. This work was supported by the European Research Council (ERC) under the project VarCity (#273940).

References

[1] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Suesstrunk. Slic superpixels compared to state-of-the-art superpixel methods. *PAMI*, 34(11):2274–228, 2012.

[2] S. Agarwal, N. Snavely, I. Simon, S. Seitz, and R. Szeliski. Building rome in a day. In *ICCV*, 2009.

[3] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *PAMI*, 23(11):1222–1239, 2001.

[4] M. Brown, D. Burschka, and G. Hager. Advances in computational stereo. *PAMI*, 25(8):993–1008, 2003.

[5] A. Chauve, P. Labatut, and J. Pons. Robust piecewise-planar 3d reconstruction and completion from large-scale unstructured point data. In *CVPR*, 2010.

	Input data			Superpixels / MRF					Planes						Timing		
Dataset	imgs	pts	rays	sp	sp(data)	pts/sp	$\bar{\mathcal{E}}^w$	$\bar{\mathcal{E}}^b$	ini	filt	merge	gco	cc	fincc	sp	3D	gco
HJ-P8	8	8.3k	25.4k	3.0k	76.6%	11.1	5.4	6.6	1883	1193	64	29	72	53	53.6	28.6	0.30
Merton I	3	2.9k	6.7k	1.6k	60.0%	7.2	5.3	2.1	623	409	69	19	57	41	26.9	9.5	0.14
Merton III	3	2.2k	5.0k	1.4k	47.9%	7.4	5.3	1.7	474	317	55	13	27	12	25.2	7.4	0.10
Mirbel	26	19.5k	66.0k	16.9k	57.0%	6.9	5.2	4.8	6068	1426	292	185	1134	576	262.1	172.5	10.6
Pozzo	53	38.6k	135k	21.4k	50.5%	12.5	5.4	6.7	8152	5481	80	58	260	76	418.8	249.3	4.5

Table 1. Numerical results. sp stands for superpixels, sp(data) for the percentage of superpixels with observed SfM points, pts/sp is the average number of points per non-empty superpixel, $\bar{\mathcal{E}}^w$ and $\bar{\mathcal{E}}^b$ are the average number of within-view and between-view graph adjacencies per superpixel. We report the number of initial (ini), filtered (filt) and merged plane hypotheses, and the number of planes after optimization (gco). The number of connected components is reported both after optimization (cc) and after the final filtering (fincc). We provide timings for superpixel extraction (sp), piecewise planar reconstruction (3D; includes optimization) and optimization (gco) in seconds.

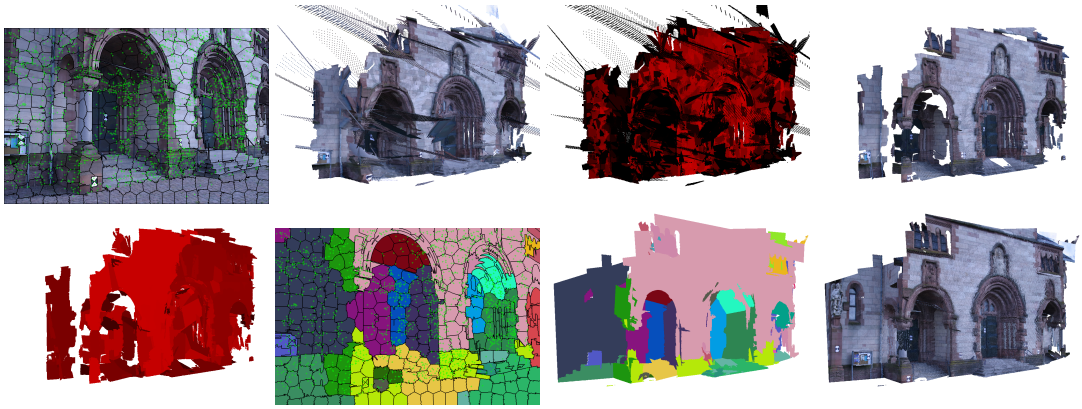

Figure 5. Detailed results for Herz-Jesu-P8. Top: input image with overlaid SfM points and superpixels; textured reconstruction of initial noisy plane estimates; color-coded plane stability measure (dark represents unstable); reconstruction with stable planes. Bottom: stability of remaining planes; segmentation result (colors represent planes); color-coded plane reconstruction; final textured reconstruction.

[6] D. Crandall, A. Owens, N. Snavely, and D. P. Huttenlocher. Discrete-continuous optimization for large-scale structure from motion. In *CVPR*, 2011.

[7] M. Farenzena, A. Fusiello, and R. Gherardi. Structure-and-motion pipeline on a hierarchical cluster tree. In *Proc. IEEE International Workshop on 3-D Digital Imaging and Modeling*, pages 1489–1496, 2009.

[8] M. Fischler and R. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of ACM*, 24(6):381–395, 1981.

[9] J.-M. Frahm, P. Georgel, D. Gallup, T. Johnson, R. Raguram, C. Wu, Y.-H. Jen, E. Dunn, B. Clipp, S. Lazebnik, and M. Pollefeys. Building rome on a cloudless day. In *ECCV*, 2010.

[10] F. Fraundorfer, K. Schindler, and H. Bischof. Piecewise planar scene reconstruction from sparse correspondences. *IVCJ*, 24(4):395–406, 2006.

[11] Y. Furukawa, B. Curless, S. Seitz, and R. Szeliski. Manhattan-world stereo. In *CVPR*, 2009.

[12] Y. Furukawa and J. Ponce. Accurate, Dense, and Robust Multi-View Stereopsis. *PAMI*, 32(8):1362–1376, 2010.

[13] D. Gallup, J. Frahm, and M. Pollefeys. Piecewise planar and non-planar stereo for urban scene reconstruction. In *CVPR*, 2010.

[14] M. Habbecke and L. Kobbelt. A surface-growing approach to multi-view stereo reconstruction. In *CVPR*, 2007.

[15] C. Haene, C. Zach, B. Zeisl, and M. Pollefeys. A Patch Prior for Dense 3D Reconstruction in Man-Made Environments. In *3DPVT*, 2012.

[16] V. Hiep, P. Labatut, J. Pons, and R. Keriven. High Accuracy and Visibility-Consistent Dense Multi-view Stereo. *PAMI*, 34(5):889–901, 2012.

[17] F. Lafarge, R. Keriven, and M. Bredif. Insertion of 3D-primitives in mesh-based representations: Towards compact models preserving the details. *IEEE Trans. on Image Processing*, 19(7):1683–1694, 2010.

[18] B. Micusik and J. Kosecka. Multi-view superpixel stereo in urban environments. *IJCV*, 2010.

[19] P. Musialski, P. Wonka, D. G. Aliaga, M. Wimmer, L. Van Gool, and W. Purgathofer. A survey of urban reconstruction. In *EUROGRAPHICS*, 2012.

[20] M. Pollefeys, D. Nister, J. Frahm, A. Akbarzadeh, P. Mordohai, B. Clipp, C. Engels, D. Gallup, S. Kim, P. Merrell, C. Salmi, S. Sinha, B. Talton, L. Wang, Q. Yang, H. Stewenius, R. Yang, G. Welch, and H. Towles. Detailed Real-Time Urban 3D Reconstruction From Video. *IJCV*, 78(2):142–167, 2008.

[21] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *IJCV*, 47(1/2/3):7–42, 2002.

[22] S. Sinha, D. Steedly., and R. Szeliski. Piecewise Planar Stereo for Image-based Rendering. In *ICCV*, 2009.

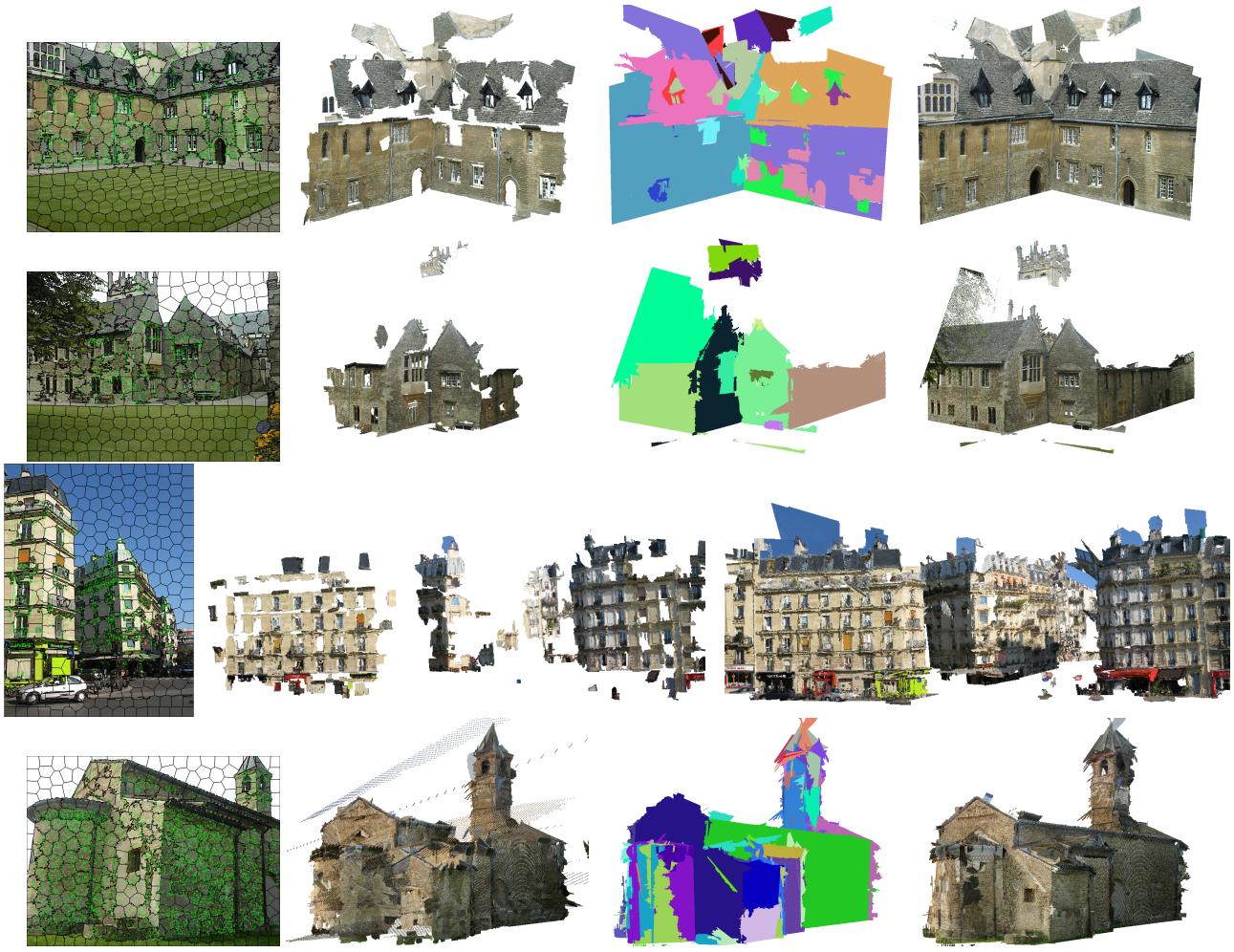

Figure 6. Results for Merton I, Merton III, Mirbel and Pozzo datasets, respectively (best viewed in color). Each row contains: (1) an input image overlayed with SfM data and superpixels, (2) initial textured plane hypotheses after filtering and merging, (3) optimized piecewise planar reconstruction where each color represents a plane (not provided for Mirbel in the 3rd row), (4) textured final reconstruction.

[23] N. Snavely, S. Seitz, and R. Szeliski. Photo tourism: Exploring image collections in 3d. In *SIGGRAPH*, 2006.

[24] C. Strecha, W. von Hansen, L. Van Gool, P. Fua, and U. Thoennessen. On benchmarking camera calibration and multi-view stereo for high resolution imagery. In *CVPR*, 2008.

[25] H. Tao, H. Sawhney, and R. Kumar. A global matching framework for stereo computation. In *ICCV*, 2001.

[26] E. Tola, C. Strecha, and P. Fua. Efficient large scale multi-view stereo for ultra high resolution image sets. *Machine Vision and Applications*, 23(5):903–920, 2012.

[27] M. Vergauwen and L. Van Gool. Web-based 3D reconstruction service. *Machine Vision Applications*, 17(6):411–426, 2006.

[28] G. Vogiatzis, C. Hernandez, P. Torr, and R. Cipolla. Multi-view Stereo via Volumetric Graph-cuts and Occlusion Robust Photo-Consistency. *PAMI*, 29(12):2241–2246, 2007.

[29] H. Wang, T.-J. Chin, and D. Suter. Simultaneously fitting and segmenting multiple-structure data with outliers. *PAMI*, 34(6):1177–1192, 2012.

[30] Z.-F. Wang and Z.-G. Zheng. A region based stereo matching algorithm using cooperative optimization. In *CVPR*, pages 1–8, 2008.

[31] T. Werner and A. Zisserman. New techniques for automated architecture reconstruction from photographs. In *ECCV*, 2002.

[32] C. Wu, S. Agarwal, B. Curless, and S. Seitz. Multicore bundle adjustment. In *CVPR*, 2011.

[33] J. Xiao, T. Fang, P. Zhao, M. Lhuillier, and L. Quan. Image-based street-side city modeling. *ACM Graphics*, 28(5), 2009.

[34] Q. Yang, L. Wang, R. Yang, H. Stewenius, and D. Nister. Stereo matching with color-weighted correlation, hierarchical belief propagation, and occlusion handling. *PAMI*, 31(3):492–504, 2009.

[35] C. Zitnick and S. Kang. Stereo for image-based rendering using image over-segmentation. *IJCV*, 75(1):49–65, 2007.