Piecewise Planar City 3D Modeling from Street View Panoramic Sequences*

Branislav Mičušík[1,2] Jana Košecká[1]
[1]George Mason University, Computer Science Department, Fairfax, USA
[2]Austrian Research Centers, Video and Security Technology Unit, Vienna, Austria

Abstract

City environments often lack textured areas, contain repetitive structures, strong lighting changes and therefore are very difficult for standard 3D modeling pipelines. We present a novel unified framework for creating 3D city models which overcomes these difficulties by exploiting image segmentation cues as well as presence of dominant scene orientations and piecewise planar structures. Given panoramic street view sequences, we first demonstrate how to robustly estimate camera poses without a need for bundle adjustment and propose a multi-view stereo method which operates directly on panoramas, while enforcing the piecewise planarity constraints in the sweeping stage. At last, we propose a new depth fusion method which exploits the constraints of urban environments and combines advantages of volumetric and viewpoint based fusion methods. Our technique avoids expensive voxelization of space, operates directly on 3D reconstructed points through effective kd-tree representation, and obtains a final surface by tessellation of backprojections of those points into the reference image.

1. Introduction

Demand for low cost acquisition of large scale city 3D models from a video stream taken from a moving vehicle has been increasing. Such models have been of use in many applications including navigation, driving direction pre-visualizations and augmented reality as demonstrated in Google Earth or Microsoft Virtual Earth. The city environments often lack textured areas, contain repetitive structures, many occlusions, strong lighting changes, and cast shadows. These properties make vision-based 3D modeling difficult in the sense of finding enough reliable point matches between overlapping images so important for following surface reconstruction. To overcome the ill-posed matching stage, as a remedy, laser scanners are often used [8]. The availability of laser scans in extended urban areas is still sparse and acquisition quite costly, compared to the vast amount of geo-registered panoramic imagery as provided by applications such as Google Streetview.

In this paper, we are interested in purely passive vision-based 3D modeling. Previously proposed attempts [1, 11, 4] each presents some variations and improvements of the standard 3D modeling pipelines. Each pipeline typically starts with image matching followed by pose estimation and dense stereo, and ends up with a fusion of partial depth maps into one consistent 3D model. The dense stereo part is the most crucial part as the existing dense stereo methods are in most cases pixel-based [16], working reliably on well textured surfaces. Because of repetitive or no texture, the state-of-the-art multi-view stereo methods, *e.g*. [9], when applied in urban settings, lack depth estimates in many areas of uniform intensity and recovered planar surfaces are slightly bumpy despite the smooth Poisson surface reconstruction. Examples of various comparisons can be found in [14]. Mapped textures therefore look locally deformed, squeezed or prolonged, which results in jaggy projections of rectilinear structures like windows, doors, see results in [1, 11]. To avoid disadvantages of pixel-based stereo apparent in urban settings, [4] assumes ruled surfaces models of urban scenes, which enable to compute photoconsistency similarity over line segments and hence reduce the stereo matching ambiguities. This however makes the method unsuitable for recovery of various vertical facade indentations.

To overcome the above mentioned drawbacks we therefore suggest to a priori exploit image segmentation cues as well as presence of dominant scene orientations and piecewise planar structures. We employ those assumptions already in the dense matching and reconstruction stage, in contrast to model based techniques which fit various primitives, *e.g*. windows, walls, afterwards to reconstructed cloud of points [5, 6]. The piecewise planarity allows any facade indentations, and moreover, to explicitly suppresse the jaggy effect coming from the incorrect texture mapping. We believe that locally line preserving texture mapping on a

*This research received funding from the US National Science Foundation Grant No. IIS-0347774 and the Wiener Wissenschafts-, Forschungs- und Technologiefonds - WWTF, Project No. ICT08-030.

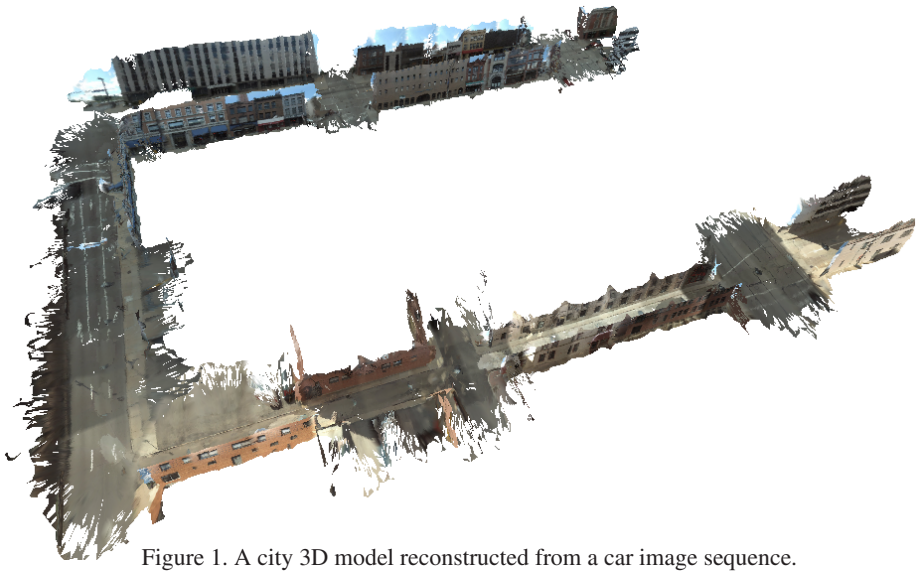


Figure 1. A city 3D model reconstructed from a car image sequence.

coarse planar 3D model often provides better visual experience than deformed textures on not completely or incorrectly reconstructed details.

The contribution of this paper is a unified and complete pipeline for piecewise planar city 3D modeling from street view panoramic sequences. Namely, *i)* we modify a method for pose estimation to exploit beneficial properties of the panoramic camera with one virtual optical center, *ii)* we utilize dominant scene orientations and adopt superpixels in a modified Markov Random Field (MRF) based dense stereo reconstruction method, and *iii)* we introduce a new depth map fusion algorithm combining advantages taken from volumetric- and viewpoint-based fusion methods. Our technique avoids expensive voxelization of space, operates directly on 3D reconstructed points through an effective kd-tree representation. As the result, a textured triangulated surface mesh of an observed environment is obtained, see an example in Fig. 1.

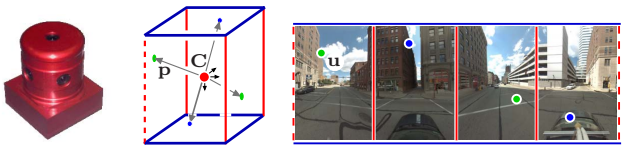
The structure of the paper is the following. The camera model, matching, and pose estimation is explained in Sec. 2. The camera poses are utilized in the superpixel dense stereo method, outlined in Sec. 3. Partial depth maps are fused by the algorithm described in Sec. 4. Examples of reconstructed 3D models are discussed in Sec. 5.

2. Pose Estimation

Using standard cameras in urban scenes makes SfM very difficult or impossible as the images often contain just one plane (road or building facade), or most of the image is occluded by moving cars and pedestrians.

Let us assume that we have images acquired by standard perspective cameras aligned in a circle, see Fig. 2 (a). We create one panoramic image by warping the radially undistorted perspective images onto the sphere assuming one virtual optical center. One virtual optical center is reasonable assumption considering that the structure around the sensor is very far compared to the discrepancy between optical centers of all the cameras. The sphere is backprojected into a quadrangular prism to get a piecewise perspective panoramic image, see Fig. 2. Our panorama is composed then of four perspective images covering in total 360 deg horizontally and 127 deg vertically. We do not use the top camera as there is not much information. To represent the panorama by using the piecewise perspective, rather than often used cylindrical, projection contributes to better performance of image point matching algorithms. The reason is that their assumption of locally affine transformation between matched image regions is more feasible for perspective images than for the cylindrical panoramas.

We employ the SURF-based matching algorithm by [2] between each consecutive image pair along the sequence. The spherical, respectively prismatic, representation of the omnidirectional image allows us to construct corresponding 3D rays $\mathbf{p}, \mathbf{p}'$ for established tentative point matches $\mathbf{u} \leftrightarrow \mathbf{u}'$. The tentative matches are validated through RANSAC-based epipolar geometry estimation formulated on their 3D rays, *i.e.* $\mathbf{p}'^{\top}\,\mathtt{E}\,\mathbf{p} = 0$, yielding thus the essential matrix $\mathtt{E}$ [10]. To treat the images as being captured by a central omnidirectional camera is very beneficial in many aspects. As the 3D rays are spatially well distributed and cover large part of a space, it results in very stable estimate of the essential matrix, studied in [3]. Moreover, improved convergence of RANSAC can be achieved if rays are sampled uniformly from each of four subparts of the panorama. The large field of view (FOV) especially contributes towards better distinguishing of a rotation and translation obtained from the essential matrix. The four-fold ambiguity remain-



(a) (b) (c)

Figure 2. Acquisition device. (a) Point Grey Ladybug® camera. (b) Quadrangular prismatic camera model with a virtual optical center $\mathbf{C}$. (c) A panoramic piecewise perspective image as an outer surface of the prism. An image point $\mathbf{u}$ is represented by a calibrated 3D vector $\mathbf{p}$.

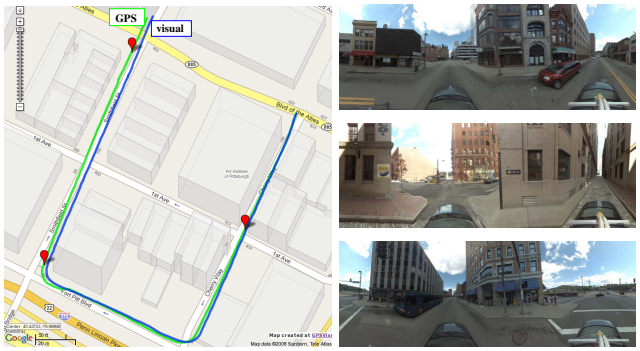
ing after extracting the rotation and translation from the essential matrix is solved by assuming that camera moves forward.

For solving scales of translations between consecutive pairs of images along the sequence we employed the following strategy. We set the norm of the translation for the first pair to be 1. For each new image, the pose is estimated from epipolar geometry between the new and previous image. Scale of the translation is estimated by a linear closed-form 1-point algorithm on corresponding 3D points triangulated by DLT [10] from the previous image pair and the actual one. This is possible because of the aforementioned fact that the omnidirectional cameras, thanks to their large FOV, give very stable pose estimate even from the epipolar geometry. The estimate in this way offers poses accurate enough even without bundle adjustment unless the baseline is too small. The same observations were done by [18], where they use the rotation coming from epipolar geometry and search not only for the scale but also for translation proposing a 2-point algorithm. It differs from a well known and often used technique [15] where full pose of the third camera, *i.e*. the translation, its scale, and the rotation, is estimated through the 3-point algorithm. However, in case of omnidirectional cameras, [18] reports superior performance of using the epipolar geometry and reduced 2-point than the full 3-point algorithm.

Fig. 3 shows an example of the pose estimation with comparison to GPS data. Notice that the GPS position can get biased or can contain small jumps, especially when satellites in one half of the hemisphere are occluded by nearby buildings. Moreover, the GPS itself does not provide rotation information unless it is equipped with a compass. The visual odometry offers very good complementary information to the GPS, and optimally, they should be combined.

3. Superpixel stereo

Given the estimated poses we want to compute depths for all pixels in each image along the sequence. Standard multiview dense stereo methods, reviewed in [16], are well conditioned in restricted scenarios. They lack robustness or



(a) (b)

Figure 3. Pose estimation. (a) Trajectory estimated by our method from images (blue) and by GPS (green) visualized in the Google Maps by the GPSVisualizer. Our trajectory is put into the world coordinate system by aligning first two estimated poses in the sequence with the GPS. (b) Three images captured along the trajectory at places marked in the map to the left.

produce inaccurate and incomplete models for many specific scenes like the urban ones. Ambiguities in the standard dense stereo pipeline are magnified when the scenes lack textured areas, contain repetitive textures, and when lighting varies dramatically across the views. We propose to tackle those problems by utilizing unique properties of the urban scenes, such as piecewise planarity and dominant orientations to condition better the problem of the 3D structure estimation. It is shown that utilization of the scene priors yields, compared to the standard dense stereo pipelines [19, 9], more accurate and visually plausible results in many urban scenes. We have adapted the multiview superpixel stereo method proposed in [14] to work directly on prismatic panoramas and exploit additional priors related to the knowledge of the camera and horizon line. The spatial constraints between neighboring regions in panoramas, enable us most effectively exploit the entire 360 deg horizontal FOV. Let us briefly outline the main steps.

The depth map is estimated for each image in the sequence by taking consecutively each image as a reference one while considering two previous and two next images. The multiview stereo is thus solved on 5 consecutive images where the middle one is the reference image. First, the image is pre-segmented into superpixels, see Fig. 4. The superpixels have been used in the past extensively as intermediate primitives in various formulations of image parsing and object recognition tasks. This is due to the fact that they often naturally correspond to semantically meaningful object (scene) primitives/parts. In connection to dense stereo the superpixels have been utilized by [17, 22] to disambiguate textureless areas, *i.e*. by enforcing pixels in detected color-consistent regions to lie at the similar depth. They use small superpixels and use them in the smoothness term to regularize the estimate. While they still estimate depth for each image pixel we assign one depth per entire

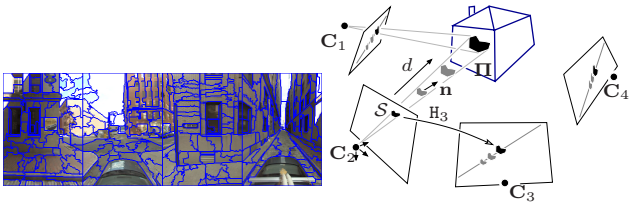


Figure 4. Superpixel dense stereo. Left: A reference image segmented into superpixels by [7]. Right: Concept of the sweeping sketched up, for clarity, for a set of perspective cameras. In case of omnidirectional images, just the quadrangular prisms would be replaced for image planes. A superpixel $\mathcal{S}$ from the reference image is swept along a corresponding projective ray by d with the normal $\mathbf{n}$ and projected into other views in order to measure the photoconsistency. It is analogous to applying a plane-induced homography $\mathtt{H}_k(\mathbf{n}, d)$ on pixels inside the superpixel.

superpixel.

The goal is to find for each superpixel its depth and normal giving minimal photoconsistency error when projecting it into other views while considering smooth changes of the depths and normals of the neighboring superpixels. We search for a Maximum Posterior Probability (MAP) assignment of a MRF, whose graph structure is induced by neighborhood relationships between the superpixels. Formally, we seek such $\mathcal{P}^*$ that

$$\mathcal{P}^* = \underset{\mathcal{P}}{\operatorname{argmin}} \left[\sum_s E_{photo} + \lambda_1 \sum_s E_{geom} + \lambda_2 \sum_{\{s,s'\}} E_{norm} + \lambda_3 \sum_{\{s,s'\}} E_{depth} \right], \quad (1)$$

where E's stand for energies, discussed later, λ's are their weights, $\{s, s'\}$ are neighboring superpixels, and $\mathcal{P}$ is a set of all possible planes for all S superpixels, *i.e.* $\mathcal{P} = \{\mathbf{\Pi}_s : s = 1 \dots S\}$ and $\mathbf{\Pi}_s = [\mathbf{n}_s^\top \; d_s]^\top$ consists of a superpixel normal and depth. In the MRF graph structure the superpixels s stand for graph vertices and pairwise connections $\{s, s'\}$ are established between all neighboring superpixels.

To make the NP-hard problem in Eq. (1) tractable we formulate it as a discrete *labeling* problem on a graph with fixed small number of L labels per superpixel. This is advantageous compared to discretizing the space of disparities as typically done in MRF formulations of dense stereo methods. Our labels correspond to planar hypotheses obtained in the sweeping stage. We assume the Manhattan world, *i.e.* we restrict number of plane normals to three orthogonal ones, captured by vanishing points. The vanishing points can be detected automatically [12], however, it is not always possible to find them reliably in all images. We therefore propagate them and project them through known camera poses to images where the detection is difficult.

A label l_s for a superpixel s corresponds to a possible candidate for depth with a particular normal obtained in the sweeping stage, conceptually shown in Fig. 4, as a local minimum of the photoconsistency measure

$$C_s(\mathbf{n}, d) \quad = \quad \frac{1}{|\mathcal{K}|} \sum_{k \in \mathcal{K}} \left(\chi^2_{sk} + \alpha \|\mathbf{c}_s - \mathbf{c}_k\|^2 \right), \quad (2)$$

over the depth d. We typically consider 3 best minima as the candidates at each out of three normal. The sum is evaluated over all valid projections $\mathcal{K}$ of the superpixel s in other views. At each homography induced projection $\mathtt{H}_k(\mathbf{n}, d)$ a chi-squared color histogram and α weighted chromacity difference is evaluated. Interior pixels of the superpixel s and its kth projection are first used to compute chromacity vectors $\mathbf{c}_s$ and $\mathbf{c}_k$, second, to photometrically normalize them (enforcing zero mean and unit variance per color channel). Integrating appearance over larger spatially meaningful area of superpixels contributes to higher robustness of the photoconsistency measure than widely used NCC, SSD or SAD measures applied on small squared windows in pixel-based stereo techniques. That allows to handle much better the inaccuracies in pose estimates and lighting changes across the views.

The photoconsistency term $E_{photo}(s, l_s)$ is then equal to $C_s(\mathbf{n}, d)$ at the depth represented by the label l_s. The geometric term $E_{geom}(s, l_s)$ captures consistency of the superpixel boundary to be parallel to a plane with the normal represented by the l_s. It is measured via deviation of the gradient orientation of the pixels along the superpixel boundary to two vanishing points. The smoothness term $E_{depth}(s, s', l_s, l_{s'})$, resp. $E_{norm}(s, s', l_s, l_{s'})$, enforces a smooth depth, resp. normal, transition between neighboring superpixels $\{s, s'\}$. Due to space limitation, detailed description of the energy terms can be found in [14].

The MAP of the MRF enforcing neighborhood relation is solved by [20] twice in two iterations. First iteration gives the rough estimate of the 3D layout of the scene and thus offers a prior on scene depths. The prior is then combined with the photoconsistency measure in Eq. (2) to obtain new more reliable depth candidates used to rebuild the weights in the MRF whose MAP gives the final solution. Compared to [14] we utilize two additional constraints. First, we employ point matches established in the pose estimation stage, described in Sec. 2. If the superpixel contains at least 2 matches reconstructed at similar depth the search for depth candidates is restricted to a small band including those depths. Second, the depth of a road can be reliably detected after the first iteration as the most dominant depth in the direction of the vertical normal. The depth of the road directly restricts the depths of other superpixels as we enforce everything to lie above the road. This significantly helps to eliminate many incorrect depth candidates found in the sweeping stage and yields a much more accurate result.

Fig. 5 shows one of the dense stereo result on the middle

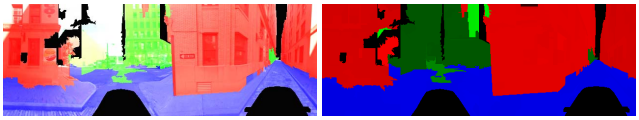


Figure 5. A depth map. Left: Estimated normals superimposed into the input reference image. Color encodes the index of the assigned normal where black color stands for undecided pixels. Right: Depth estimates encoded by color saturation. Further the superpixel is estimated along the assigned normal in darker color it is depicted. Notice that most of the planes in the scene got superpixels assigned to the correct normal and depth (same depth is indicated by the same color saturation). Best viewed in color.

image from Fig. 3 (b). One can see that most of the superpixels are assigned the correct normal and depth. Some of them got undecided because of too high cost in any decision. Nevertheless, the incorrectly or not estimated places are treated and filtered out by the fusion step described in the following section.

4. Fusion of Depth Maps

Given the depth maps for all images along the sequence, like the one shown in Fig. 5, we want to fuse them and build one consistent 3D model represented by a triangulated surface mesh. In many instances we expect that raw stereo depth maps are mutually redundant and contain many errors and do not completely agree with one another.

To fuse the multiple depth maps there are volumetric or the viewpoint-based approaches. The volumetric approaches, *e.g*. [21], voxelize 3D space and in probabilistic framework, each depth votes for the voxels along the corresponding projective ray. The final surface is estimated as an isosurface using *e.g*. the Marching Cubes or using graph cuts. There are techniques sampling the 3D space more efficiently by using Octrees but still, those methods are very memory and computationally expensive. We want to represent large area where voxel number would be enormous and their processing too time demanding. We instead pursue a viewpoint-based approach. The most similar approach to ours is [13] which renders multiple depth maps into the reference view as well as renders the reference depth map into the other views in order to detect occlusions and free-space violations and to find a closest stable depth.

Since we have three dominant normals used in the depth map estimation step, it allows us to split the fusion into three steps and process together only the points on the same normal without mutual interaction. As oppose to the rendering of the depth maps as in [13], we propose to perform the fusion on the reconstructed 3D points to better handle inaccuracies in the depth estimates. This strategy is motivated by successful voxel-based approaches working in 3D space while here avoiding the space voxelization. We utilize the kd-tree data structure which can be favorably utilized to effectively traverse point neighbors. The fusion strategy is as follows. Set the counter $k := 1$.

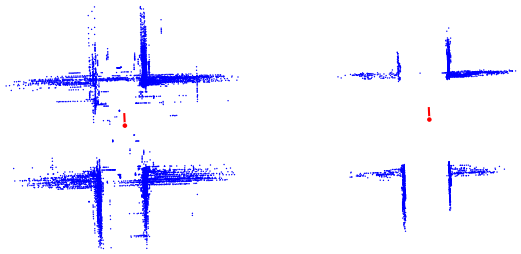


Figure 6. Filtering stage. The images show a top view of the reconstructed cloud of 3D points from a set of 10 consecutive depth maps (the set starts from the middle image in Fig. 3 (b), the car is in the middle of a road crossing). Only points assigned to two normals which are perpendicular to vertical facades are shown. The red dot in the middle with a short line represents the position of the reference image and moving direction of a car. Left: An initial reconstruction of points containing many spurious outliers. Right: Filtered points. Only stable 3D points having supported by multiple depth maps survived and were justified.

1. Consider M consecutive images with their depth maps starting at the index k and choose the middle image as being the reference one.

2. Reconstruct 3D points from all points in the provided depth maps and transform them by known camera poses into the coordinate system of the reference image. Construct three kd-trees, each from the 3D points belonging to one out of three normals.

3. For each 3D point, find (utilizing fast kd-tree operation) points with the same normals inside a sphere with a pre-defined radius centered at that point. Project the points onto a normal vector and compute a mean of the projections. Replace the point by the new mean value only if there are at least 3 other points inside the sphere coming from different depth maps. Otherwise, remove the point as being an outlier. See Fig. 6.

4. Project the filtered 3D points back into the reference view. Merge those projections falling into the same image pixels, triangulate them, and assign the means over 3D points which correspond to the merged projections to 3D coordinates of triangle vertices. Store color of each vertex as the color of its projection in the reference image. Special attention need the points under the car. They are used for triangulation to get consistent surface but the corresponding color is taken from an image 3 steps backwards. Finally, the triangles having in 3D at least one side longer than some threshold are removed. Those triangles correspond usually to places not reliably reconstructed, at occlusions or depth changes.

Figure 7. Detailed views on the 3D model from Fig. 1.

5. Store triangulated 3D vertices with their colors and set $k := k + M/2$. If it is the first considered M-tuple, continue with the step 1.

6. Fuse the actual 3D mesh with the previous one by the following strategy. Take out the 3D points in a predefined band around a plane perpendicular to the line connecting optical centers of the actual and the previous reference image. Merge those points from actual and previous mesh which are mutually close enough, average their positions and color, and update both meshes. Again, the kd-tree is used to significantly speed up the search for the nearest neighbors. Finally, discard the 3D points and corresponding triangles outside the band which are closer to the previous reference view.

The use of one reference image and projecting the filtered 3D points into it is very advantageous. It allows to triangulate the projected points in the image plane and thus avoids the need of expensive and problematic voxel based surface extraction methods. We use $M = 10$ in our experiments.

5. Experiments

We present results on two street-view sequences[1], each consisting of 200 panoramic images taken about 2 m apart. For the pose estimation we use full resolution of the images 2560×904 pixels, but for the dense stereo we downsampled the panoramas by factor 4. The first sequence is captured along a U-shaped trajectory. Three of its images and pose estimation are shown in Fig. 3 and the final 3D model in Fig. 1. Most of the seen facades and road surface are nicely reconstructed and together with mapped texture the method provides visually appealing model. The 3D models are visualized in the free MeshLab tool.

The second, L-shaped, sequence is more challenging as it contains many trees, moving cars, and pedestrians. See Fig. 8 for the entire 3D model. Fig. 9 depicts a part

[1]Provided and copyrighted by Google.

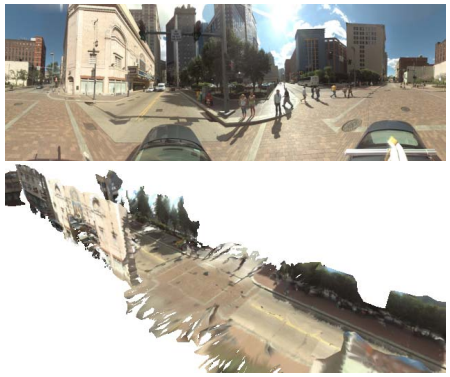


Figure 9. An image from the second sequence and the detail on the 3D model at this part. The are many trees to the right of the car, reconstructed as planar surfaces.

of the scene with pedestrians and trees. The trees are typically reconstructed as planar surfaces and pedestrians are automatically removed in the fusion stage unless they are static. The moving cars occlude substantial part of the road and may cause holes in the final model, depending on their speed.

The obtained 3D model, Fig. 7, with locally line preserving texture mapping is often sufficient for 3D visual experience of the environment. Moreover, notice in both models nicely reconstructed road surface throughout the entire sequences despite the road being weakly textured. Although at the moment we do not handle explicitly non-planar structures, such as cars and trees, they can be properly handled by integrating a detection / recognition module as done for the cars in [4].

The reconstruction of the facades perpendicular to the camera motion is poorly conditioned compared to the facades parallel to the motion. This is due to the fact that the forward motion of the vehicle places epipoles at the center of front and rear views of the panorama and for the pixels close to epipoles large changes in depth cause small displacement in the image, making many depth hypotheses equally likely. Our suggested superpixel representation partially overcomes these problems by explicitly merging crucial pixels with those further away from the epipoles and hence reducing the likelihood of many spurious hypotheses.

6. Conclusion

We have presented a 3D modeling pipeline of urban environments from a sequence of images taken by a set of cameras mounted on a moving vehicle. We have modeled the acquisition setup as an omnidirectional camera with one virtual optical center and achieved robust visual odometry estimate without a need for bundle adjustment. In order to handle difficult, usually textureless, urban scenes, we have utilized image segmentation cues and dominant scene orienta-

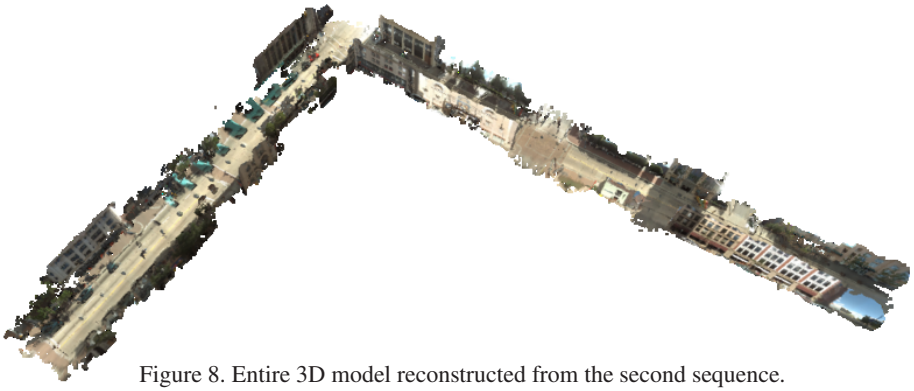


Figure 8. Entire 3D model reconstructed from the second sequence.

tions for a dense stereo reconstruction. A partial piecewise-planar models of the scene are fused by a proposed novel method into one textured triangle surface mesh.

Our suggested pipeline offers a powerful alternative towards handling of the difficult urban scenes. The suggested approximation of observed scenes by piecewise planar models captured by superpixels usually results in more visually pleasing texture mapping than the standard approaches. However, the use of superpixels may be limited at places no well described by local planar patches. In those cases, the superpixels can be further down-segmented, in limit, approaching pixel-based dense stereo approaches.

References

[1] A. Akbarzadeh, J. Frahm, P. Mordohai, B. Clipp, C. Engels, D. Gallup, P. Merrell, M. Phelps, S. Sinha, B. Talton, L. Wang, Q. Yang, H. Stewenius, R. Yang, G. Welch, H. Towles, D. Nister, and M. Pollefeys. Towards urban 3D reconstruction from video. In *Proc. of Int. Symp. on 3D Data, Processing, Visualiz. and Transmission (3DPVT)*, 2006.

[2] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. Speeded-up robust features (SURF). *Computer Vision and Image Understanding (CVIU)*, 110(3):346–359, 2008.

[3] T. Brodsky, C. Fermüller, and Y. Aloimonos. Directions of motion fields are hardly ever ambiguous. *IJCV*, 1(26):5–24, 1998.

[4] N. Cornelis, B. Leibe, K. Cornelis, and L. Van Gool. 3D urban scene modeling integrating recognition and reconstruction. *IJCV*, 78(2-3):121–141, 2008.

[5] P. E. Debevec, C. J. Taylor, and J. Malik. Modeling and rendering architecture from photographs: A hybrid geometry- and image-based approach. In *SIGGRAPH*, pages 11–20, 1996.

[6] A. R. Dick, P. H. Torr, and R. Cipolla. Modelling and interpretation of architecture from several images. *IJCV*, 60(2):111–134, 2004.

[7] P. Felzenszwalb and D. Huttenlocher. Efficient graph-based image segmentation. *IJCV*, 59(2):167–181, 2004.

[8] C. Frueh, S. Jain, and A. Zakhor. Data processing algorithms for generating textured 3D building facade meshes from laser scans and camera images. *IJCV*, 61(2), 2005.

[9] Y. Furukawa and J. Ponce. Accurate, dense, and robust multi-view stereopsis. In *Proc. of CVPR*, 2007.

[10] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, second edition, 2004.

[11] A. Irschara, C. Zach, and H. Bischof. Towards wiki-based dense city modeling. In *ICCV Workshop on Virtual Representations and Modeling of Large-scale environments (VRML)*, 2007.

[12] K. Kanatani and Y. Sugaya. Statistical optimization for 3-D reconstruction from a single view. *IEICE Trans. on Information and Systems*, E88-D(10):2260–2268, 2005.

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

[14] B. Mičušík and J. Košecká. Multi-view superpixel stereo in man-made environments. Technical Report GMU-CS-TR-2008-1, George Mason University, USA, 2008.

[15] D. Nistér. Preemptive RANSAC for live structure and motion estimation. *Machine Vision Application (MVA)*, 16(5):321–329, 2005.

[16] S. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *Proc. of CVPR*, pages 519–528, 2006.

[17] J. Sun, Y. Li, S. B. Kang, and H. Y. Shum. Symmetric stereo matching for occlusion handling. In *Proc. of CVPR*, pages II: 399–406, 2005.

[18] J.-P. Tardif, Y. Pavlidis, and K. Daniilidis. Monocular visual odometry in urban environments using an omnidirectional camera. In *Proc. of IEEE Int. Conf. on Intelligent Robots and Systems (IROS)*, 2008.

[19] M. Vergauwen and L. Van Gool. Web-based 3D reconstruction service. *Machine Vision Application (MVA)*, 17(6):411–426, 2006. `http://www.arc3d.be`.

[20] T. Werner. A linear programming approach to Max-sum problem: A review. *PAMI*, 29(7):1165–1179, 2007.

[21] C. Zach, T. Pock, and H. Bischof. A globally optimal algorithm for robust TV-L1 range image integration. In *Proc. of ICCV*, 2007.

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