Piecewise Planar Segmentation for Automatic Scene Modeling

Adrien Bartoli, *Adrien.Bartoli@inria.fr*
INRIA Rhône-Alpes, 655, av. de l'Europe
38334 St. Ismier cedex, France.

Abstract

In this paper, we investigate the problem of the automatic creation of 3D models of man-made environments that we represent as collections of textured planes.

A typical approach is to automatically compute a sparse feature reconstruction and to manually give their plane-memberships as well as the delineation of the planes. Textures are then extracted from the images while optimizing the model, typically the disparity between marked and predicted edges.

We propose a means to automatically estimate the model of the scene, in terms of the number of planes and their parameters from a point feature reconstruction. The method is based on random sampling of reconstructed points to generate plane hypotheses. Each of these is then evaluated using a measure of approximate photoconsistency while recovering the corresponding plane delineation.

We then compute the maximum likelihood estimate of all scene parameters, i.e. the set of planes and reconstructed points as well as relative camera pose, with respect to actual images. The approach is validated on simulated data and real images.

1. Introduction

Automatic modeling of *3D* scenes from images is one of the most challenging research areas in computer vision. Among the variety of all possible scenes, we consider the widespread case of man-made environments. Those scenes are often composed of piecewise planar primitives and can be modeled by a collection of planes [2, 3, 5, 7] to a reasonable degree of approximation. Figure 1a shows an image of such a scene, overlaid by a piecewise planar segmentation, while figure 1b shows a view of the corresponding recovered model.

Existing systems are most of the time semi-automatic and can be seen as three-stage processes [5, 11, 16]. First, a sparse *3D* reconstruction of features (points, lines, etc.) as well as cameras is performed automatically [4]. From

there, two stages remain: choosing the scene model and estimating its parameters. The first stage is achieved by clustering reconstructed features into higher level geometric primitives such as cubes by e.g. marking edges in actual images. The second one consists in optimizing the quality of the model parameters, e.g. in minimizing the disparity between marked and predicted edges.

This approach has proven to give highly photorealistic results, but becomes effort-prone as the scene considered grows in complexity. In this paper, we devise a method that

(a)   (b)

Figure 1. Example of piecewise planar modeling: (a) **shows an image overlaid with the automatically recovered piecewise planarity and** (b) **the model rendered from a different point of view.**

is automatic and produces the most likely scene parameters with respect to actual images.

In [13], the scene surface is modeled as a set of triangles. The most likely triangulation with respect to actual images is computed using edge swaps from an initial solution obtained based on a Delaunay triangulation. The process is not guaranteed to converge to the global optimum and its complexity is high. The piecewise planarity is not taken into account, which reduces photorealism when rendering planar surfaces from oblique points of view.

To overcome these problems, we represent a scene as a collection of planes. This reduces the complexity of the model computation as well as its rendering and yields more photorealistic view synthesis of planar surfaces. The recovered structure and motion can be greatly enhanced by using the coplanarity information [3]. We follow the same three stages described above. Our modifications are two-

fold. First, we fill the gap of interactivity by automatically choosing a piecewise planar model of the scene and second, we compute the maximum likelihood estimate of the model parameters with respect to actual images.

Choosing the scene model can be done by fitting planes to reconstructed points. This is difficult because coplanar configurations that do not correspond to any world surface often arise in practice and have to be disambiguated using a physically meaningful criterion, i.e. photometric information. For that purpose, we propose to use a random sampling technique to hypothesize multiple plane equations. We then select the most likely planes with respect to actual images while checking for them to be distinct. Such an algorithm allows a point to lie on several planes.

We then refine the model parameters by non-linearly minimizing the squared greylevel difference between the actual and predicted images. This is the maximum likelihood estimate with respect to actual images. During this stage, we enforce the multi-coplanarity constraints. This is therefore a constrained photometric bundle adjustment.

This paper is organized as follow. In §§2 and 3, we review existing work and give our scene model and notations. In §4, we formally state the maximum likelihood estimation problem. We then propose an algorithm to choose the model of the scene in §5. Finally, we validate the method in §6, using both simulated data and real images.

2. Existing Work

Existing work on piecewise planar segmentation can be divided into two subsets, whether the photometric information, i.e. dense greylevel data, is taken into account or not. In particular, purely geometric criteria are sometimes used, see e.g. [1, 15]. Such criteria do not allow to eliminate coplanar configurations that do not correspond to physical planes.

Photometric information is taken into account in [2, 7]. The method described in [2] follows an approach based on line matching to segment planes from aerial images of urban scenes. In [7], the segmentation is performed using a collection of architectural primitives (pillars, doors, windows, etc.) with prior probabilities depending on the scene type.

Other possibilities for *3D* modeling are space-carving [10] and image-consistent triangulation [13], as described in the introduction.

3. Notations and Scene Model

We denote a scene as $\mathcal{S} = \mathcal{S}^m \cup \mathcal{S}^p$. As said before, it is essential to make a clear distinction between the scene model $\mathcal{S}^m$ and its parameters $\mathcal{S}^p$.

The scene model $\mathcal{S}^m = \{l, \theta_1^m, \ldots, \theta_l^m\}$ includes the number of planes l and the plane models while the scene parameters $\mathcal{S}^p = \{\theta_1^p, \ldots, \theta_l^p, \mathcal{Q}, \mathcal{P}\}$ comprise the plane parameters, the cloud of *3D* reconstructed points $\mathcal{Q}$ and the projection operator $\mathcal{P}$, equivalent to reconstructed cameras.

model θ^m	$\mathcal{V}$	visibility images
	Π	individual geometric support
	Π'	global photometric support
	∂	polygonal delineation
parameters θ^p	π	plane equation
	$\mathcal{T}$	texture map

Table 1. Modeling a plane π is achieved by a set $\theta = \theta^m \cup \theta^p$ where θ^m is the plane model and θ^p its parameters.

Similarly, each scene plane π is modeled by $\theta = \theta^m \cup \theta^p$, i.e. it has a model and an associated set of parameters, both described in table 1. In more detail, $\mathcal{V}$ designates the set of images where π is visible and Π, Π' and ∂ are defined respectively as two sets of points lying on the plane with respect to two different criteria described in §5 and the polygonal delineation of the plane, expressed as a list of points.

The set of n actual images is denoted as $\mathcal{I}$. The projection operator $\mathcal{P}$ allows, given a scene $\mathcal{S}$, to predict its images $\hat{\mathcal{I}}$.

Notations $\#\Pi$ and $\mathrm{conv}\,\Pi$ designate the number of elements and the convexe envelope of a set of points Π.

4. Maximum Likelihood Estimation

In this section, we describe how to estimate the most likely parameters $\mathcal{S}^p$ of the scene with respect to actual images, given the scene model $\mathcal{S}^m$.

If we assume our prior probability on the scene to be uniform, this is achieved by finding [7, 13]:

$$\underset{\mathcal{S}^p}{argmax}\,\Pr(\mathcal{I}|\mathcal{S}),$$

where $\Pr(\mathcal{I}|\mathcal{S})$ denotes the likelihood of the scene parameters $\mathcal{S}^p$ for the scene model $\mathcal{S}^m$ and the image set $\mathcal{I}$. We derive an expression for this likelihood, in terms of the actual and predicted images.

We assume each pixel luminance to be corrupted by an i.i.d. centered gaussian noise. Using the same reasoning as in [13], we obtain that the most likely parameters correspond to the minimum squared difference between the actual and predicted images.

The problem is then to find $argmin_{\mathcal{S}^p}\mathcal{C}(\mathcal{S})$, where the cost function is defined by the *RMS* (Root Mean of Squares) error:

$$\mathcal{C}(\mathcal{S}) = ||\mathcal{I} - \hat{\mathcal{I}}||^2, \tag{1}$$

where $\hat{\mathcal{I}}$ are the predicted images. This can be decomposed over the set of planes as:

$$\mathcal{C}(\mathcal{S}) = \sum_{\pi \in \mathcal{S}} \mathcal{C}(\mathcal{S}_\pi),$$

where $\mathcal{S}_\pi$ designates the scene restricted to plane π.

Given the scene model, the maximum likelihood estimate of its parameters can then be obtained by non-linearly minimizing (1) while enforcing the multi-coplanarity constraints between points and planes.

For that purpose, we minimally parameterize the estimation process using the technique given in [3] so that multi-coplanarity constraints are exactly enforced during the optimization. It is important to note that the cost function is not the same as that given in [3], which is purely geometric, but similar to that given in [7]. The optimization is conducted using the Levenberg-Marquardt algorithm [9].

The process can be speeded up by estimating the error around the interest points only, as described in [2].

5. Piecewise Planar Segmentation

In this section, we present a method to compute a piecewise planar model of the scene, i.e. to choose $\mathcal{S}^m$.

5.1. Overview of the Algorithm

The algorithm inputs are the actual images as well as a cloud of reconstructed *3D* points and camera matrices or equivalently the projection operator, see e.g. [4].

The piecewise planar segmentation is done by iteratively selecting the most likely plane using a random sampling technique. We use a multiple hypotheses version of *RANSAC* [8] modified in two ways. First, as described in [18], we maximize the likelihood of the plane instead of its support (the support of a plane is the set of points lying on it, up to a certain tolerance). Secondly, we devise a segmentation scheme inspired from [1] that allows for overlapping data segmentation, which is important in the piecewise planar segmentation of real scenes since a great number of planes are defined by points lying on several other planes.

5.2. Overlapping Data Segmentation

In this section, we present the robust estimator *RANSAC* and its application to disjoint data segmentation. We then propose a scheme that handles overlapping clusters.

RANSAC. This robust hypothesize and verify algorithm proceeds by repeatedly generating solutions estimated from minimal sets of points gathered from the data. This is the random sampling phase. It then tests each solution for support from the complete set of data. The parameters that maximize the support, i.e. the set of inliers, is then selected.

In the case of plane estimation, three points are randomly selected and a plane equation π is computed while testing for degeneracy. The set of points Π that geometrically lie on π, up to a predefined tolerance, is then computed image-based [1, 15] and the operation repeated. The dominant plane is the one that maximizes $\#\Pi$. The number of iterations is computed as indicated in e.g. [8].

As proposed in [18] in the case of image points, it is possible to use another cost function instead of $\#\Pi$ such as the likelihood $\Pr(\mathcal{S}_\pi|\mathcal{I})$ of the plane hypothesis. Computing this likelihood from the plane equation is described in §5.3.

Once the dominant plane has been estimated, points can be classified whether they lie, or not on the plane. In the former case, they belong to Π and in the latter, they are outliers.

Disjoint Data Segmentation. In the case of data segmentation, outliers have to be interpreted as points that do not lie on the dominant plane. Consequently, it is possible to recursively apply *RANSAC* on the set of outliers to perform data segmentation. This way of segmenting has proven to be efficient in the case of motion segmentation [6, 17] or particular planar grouping [14]. Indeed, in these cases the clusters are disjoint, e.g. a point will not, in the general case, satisfy two estimated different rigid motions.

In the case of piecewise planar segmentation, we can not make such an assumption of disjoint clusters since a point can lie on several planes. In the following paragraph, we present another scheme for data segmentation that allows overlapping clusters.

Overlapping Data Segmentation. In this case, we can not recursively apply *RANSAC* on the set of outliers for the reason mentioned above. Instead, we modify it to formulate multiple hypotheses, as described in [1], so that a point can be selected as an inlier for more than one plane. The problem is that the same plane may be hypothesized more than once. This can be solved by constraining the random sampling phase to select a plane that has not been detected so far. Therefore, we have to define a measure between two planes π_i and π_j given their support, Π_i and Π_j. This measure must be physically meaningful, e.g. comparing the equations of the two planes will not give satisfactory results since it is a purely algebraic measure. We compare the sets of inliers Π_i and Π_j of the two planes. The two planes are identical if $\mathcal{D}(i,j) < \gamma$ where $\mathcal{D}(i,j) = \frac{\#(\Pi_i \cap \Pi_j)}{\#\Pi_i + \#\Pi_j}$, i.e. the ratio of common inliers, and γ is a predefined rate, chosen as $0.4 < \gamma < 0.7$ in our experiments, see §6. Another solution would be to assess each support with the other plane equation.

We have shown how to formulate plane equation hypotheses with respect to possibly overlapping clusters. In the following section, we examine how to estimate the likelihood of an hypothesis.

5.3. Retrieving Plane Parameters

So far, we have an hypothesis of a plane equation π as well as its geometric support Π. In order to evaluate the likelihood of this hypothesis, we have to determine its model θ^m and parameters θ^p (see table 1).

Visibility images $\mathcal{V}$. Let us denote as Π' the set of points determined to lie on the plane after the delination process explained below has been conducted. Obviously, we have

$\Pi' \subset \Pi$. Consequently, we use for $\mathcal{V}$ the images where all points of Π are visible.

Delineation ∂. The process of determining the delineation is illustrated in figure 2. It is straightforward to show
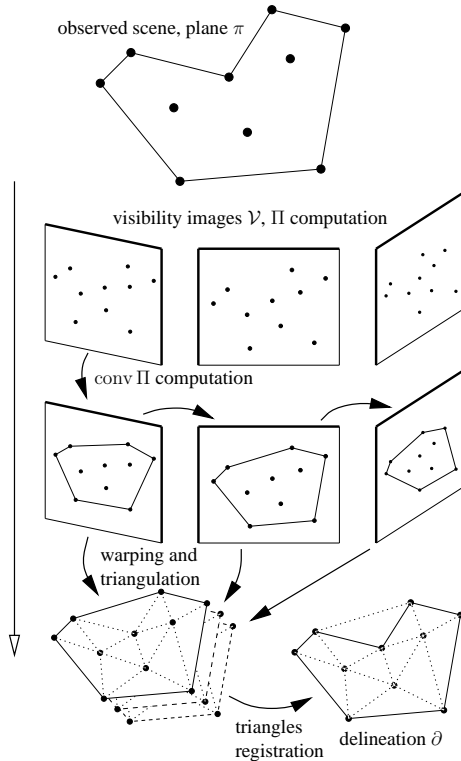

observed scene, plane π

visibility images $\mathcal{V}$, Π computation

conv Π computation

warping and triangulation

PSfrag replacements

triangles registration   delineation ∂

Figure 2. Evaluating the delineation ∂ of a plane.

that $(\Pi' \subset \Pi) \Rightarrow (\partial \subset \text{conv}\,\Pi)$. therefore, we compute conv Π and iteratively cut it to reach ∂.

For that purpose, we first warp each image of the plane with delineation conv Π, so that image points are aligned, and perform a Delaunay triangulation. The warping is done using inter-image plane homographies and bicubic interpolation. If we assume that the triangulation does not cut the edges of ∂, removing triangles from conv Π leads to ∂. Such a condition can be ensured by using an image-consistent triangulation [13]. Each triangle is then registered using an r-consistency check, as described in [10]. The *dispersion radius* r, expressed in pixels, controls the maximum image error of a reprojected point, allowing for non-perfectly planar surfaces and noisy images to be handled. The r-consistency check determines whether aligned triangles arising from different images represent the projection of the same planar surface, up to a tolerance modeled by r. The check fails if the greylevels of all pixels of all triangles are not shared by the other triangles in a disk of radius r around the pixel considered.

Such an algorithm allows for non convex and non-simply

connected plane delineation recovery, provided an appropriate data structure in the latter case.

The set of points that lie inside ∂ is the global photometric support Π' of the plane.

Texture $\mathcal{T}$. The texture is evaluated by warping each image of the plane onto a virtual fronto-parallel plane [12] and by taking the average of these values.

6. Experimental Results

In this section, we present the results we obtained by applying the algorithm described in this paper on both simulated data and real images.

6.1. Simulated Data

We compare our method to existing ones, notably to those consisting in using a disjoint data segmentation [7] or a purely geometric criterion [1, 15].

The test bench consists of a cube of one meter side length observed by a set of cameras. Points are generated on the cube, possibly offset from their planes in order to simulate non-perfect coplanarity, and projected onto the images. Each face of the cube is texture mapped using random values. The texture is also projected onto the images and normalized to lie between 0 and 1. Up to 30, 5 and 1 points are generated on respectively each face, edge and vertex of the cube. Two cameras with a focal length of 1000 (expressed in number of pixels) and a 1 meter baseline are situated at a distance of 10 meters from the cube, such that 3 of its faces are observed. The intrinsic parameters are not supposed to be known which yields projective reconstructions.

In the sequel, we vary independently each of these parameters to compare the different approaches under various conditions. We measure the quality of segmentations using the absolute difference between the number of recovered planes and the number of simulated planes, i.e. 3. We use the median value over 100 trials. Matched image points are supposed to be known up to a gaussian centered noise of 1 pixel. The cloud of *3D* points as well as camera matrices are obtained by minimizing reprojection errors in a bundle adjustment manner, using the ground truth as a starting point. The estimators compared are:

- *DDS-geom*: uses disjoint data segmentation and a purely geometric criterion, i.e. #Π, see §5.2;

- *DDS-photo*: idem but uses the photometric likelihood $\Pr(\mathcal{S}_\pi | \mathcal{I})$ criterion given in this paper in §5.3;

- *ODS-geom*: uses overlapping data segmentation described in §5.2 and a purely geometric criterion;

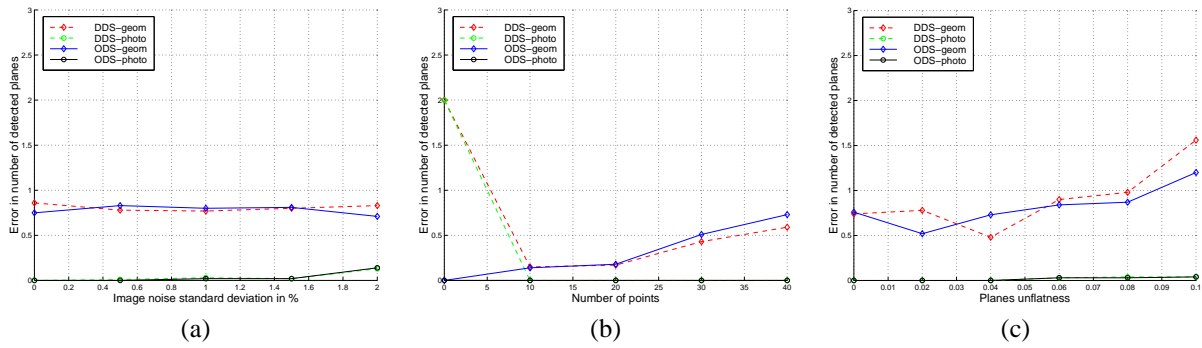- *ODS-photo*: uses both the overlapping data segmentation and the photometric likelihood criterion.

4

Figure 3. Comparison of the error in the number of detected planes for different approaches versus different scene parameters.

Let us describe the different experimental situations when varying a scene parameter and the simulation results we have obtained:

- *Added image noise*, figure 3a: a gaussian centered noise with standard deviation between 0.0 and 0.02 (i.e. 2% of the maximum) is added to the greylevel of each pixel;

- *Number of points*, figure 3b: the number of points is varied from 0 to 40 for each face of the cube. We also project each vertex but no points are simulated on the edges;

- *Plane unflatness*, figure 3c: *3D* points are offset from the plane they lie on by distances drawn from a normal distribution with standard deviation between 0.0 and 0.1 meters.

We observe that in the general case, methods *-photo*, based on photometric information perform better than methods *-geom*, based on a purely geometric criterion. In particular, the method *ODS-photo* performs better than the others in all cases.

In the case of added image noise, we observe that methods *-geom* perform worse than methods *-photo*. This can be explained by the fact that purely geometric criteria can not differentiate real planes from coplanar configurations that do not correspond to any simulated plane and as the number of simulated points is relatively low, such configurations might occur.

When varying the number of points, we observe that for low numbers (only the vertices of the cubes are used at the begining of the graph), methods *DDS-* perform worse. This is due to the fact that planes can be detected only when taking into account multi-coplanarity. When the number of points increases, methods *-geom* perform worse for the same reason as that given in the case of image noise.

In the case of increasing plane unflatness, methods *-geom* perform worse since the probability of coplanar configurations that do not correspond to any simulated plane

increases. The performance of methods *-photo* is not dramatically affected.

6.2. Real Images

In this section, we present the modeling results we obtained using the images shown in figure 4. Similar results have been obtained with other images.

We insist on the difficulty to build a piecewise planar model from these images. Indeed, windows undergo a significant parallax form their facade and can not be completely modeled using the given points which lie only on the facades. Moreover, their appearance changes across viewpoints because of specular effects. Several planes are also very thin.

We describe the different steps followed to perform a complete reconstruction, from the images to the *3D* textured model. The camera has been off-line calibrated using a calibration pattern. However, it is clear that this information is only exploited for visualizing the results and does not affect the behaviour of the scene modeling.

Structure and motion initialization. Image points are obtained by tracking by hand. We perform a partial recon-

Figure 4. 3 out of the 6 actual images of an architectural scene overlaid with features. Note the significant parallax of windows relatively to the wall.

struction from two images and incrementally add the others to obtain the complete structure and motion. We then run a bundle adjustment to minimize the reprojection error.

Piecewise planarity estimation. We use the algorithm described in this paper to recover the piecewise planarity of the scene. We have tried with a large dispersion radius [10], see §5.3, of 2% of the image size, i.e. about 10 pixels This tolerates too high errors and results in merging planes
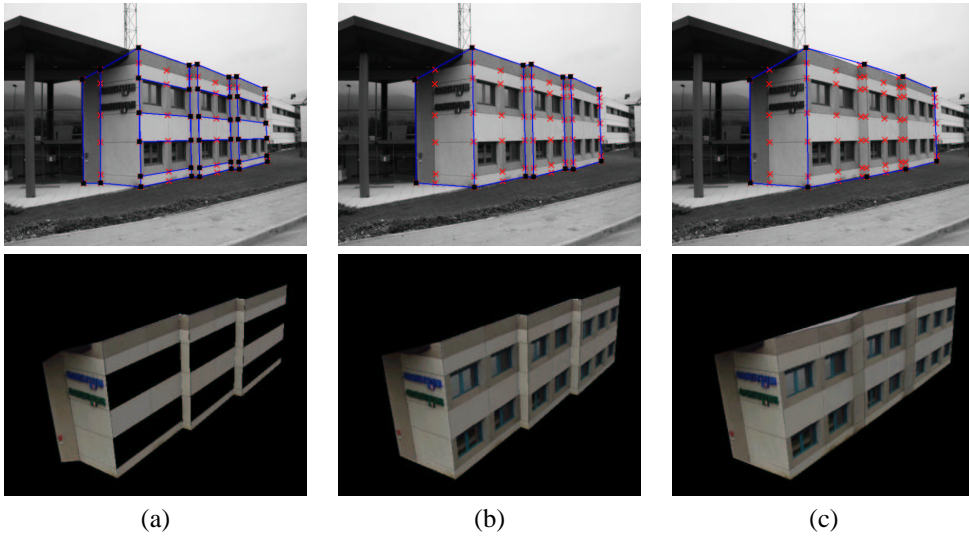
Figure 5. Recovered piecewise planarity and rendering of the corresponding model using successively 0.2, 1 and 2% of the image size (roughly 1, 5 and 10 pixels) for the dispersion radius r. In the first case, (a), planes are accurately modeled, see e.g. the two left planes. The holes correspond to windows that can not be entirely modeled and are not approximated by a plane when using such a low tolerance. For the second case, (b), the tolerance is sufficiently high, so that windows can be approximated by a plane. Note that the two left planes of (a) are now merged. In the last case, (c), two planes suffice to completely approximate the scene surface, which yields several artefacts on the rendering.

that could have been modeled separately, figure 5c. In the case of a low dispersion radius, 0.2% of the image size, which roughly corresponds to 1 pixel, the recovered piecewise planarity is incomplete, due to the parallax of windows relatively to the wall, figure 5a. An in-between dispersion radius of 1% of the image size, i.e. about 5 pixels, gives a complete and photorealistic-looking piecewise planar model, figure 5b.

The results we obtained are therefore satisfactory. The method tolerates approximately planar surfaces and is able to detect and model small planes.

Maximum likelihood estimate. We use the technique described in [3] to minimally parameterize the structure while enforcing previously recovered piecewise planarity. The optimization process is conducted as described in §4 using the scene model $\mathcal{S}^m$ obtained with r=5 pixels (figure 5b). The sparse feature reconstruction is exactly piecewise planar after constrained optimization. Figure 6 shows texture-mapped renderings of the recovered model from various points of view different from the original ones. The texture maps for windows have been extracted from one image each, to avoid the blur effect caused by the parallax. Another possibility to render such approximate planes is to use the View-Dependent Texture-Mapping described in [5].

Quality assessment. We have performed several measures on the structure obtained before and after the constrained photometric bundle adjustment for the piecewise planarity corresponding to a 1 pixel dispersion radius (fig-

	σ_1	σ_2	μ
point-based	0.0138	0.0419	0.0350
plane-based	0.0087	0.0302	0.0209

Table 2. Metric measures on the initial euclidean reconstruction (point-based) and on that obtained after the constrained photometric bundle adjustment (plane-based). The lower σ_1, σ_2 and μ (see text) are, the better the reconstruction is.

ure 5a). Two kinds of quantity are significant: length ratios and angles. Table 2 shows measures of such quantities. In this table, σ_1 and σ_2 are the variances of the length of respectively the 6 vertical edges and the 6 horizontal edges of equal length, whereas μ is the mean of $1 - 2\alpha_i/\pi$ where α_i are the measures of right angles.

The values given in table 2 show that the metric reconstruction obtained with the method given in this paper is of superior quality than the one obtained with a traditional method based only on points. This is important since several planes of the scene have a uniform appearance and therefore do not greatly contribute to the photometric bundle adjustment.

Similar measures performed on the other reconstructions show that the lower the dispersion radius is, the better the reconstruction is. This is due to the fact that high dispersion radii tend to merge close planes (see e.g. the two left planes

6

Figure 6. Texture-mapped model obtained.

of figure 5a, merged in figures 5b and 5c). For that reason, the individual position of each plane is less accurate and the reconstruction quality lower. Consequently, there is a trade-off between the reconstruction quality and the surface approximation.

7. Conclusions, Perspectives

We have investigated the automatic modeling of scenes using planes. The method is based on segmenting a cloud of reconstructed *3D* points into multi-coplanar groups. We use random sampling to generate multiple plane hypotheses and select the most likely planes with respect to actual images. The use of a photometric measure allows to distinguish coplanar point configurations that do not correspond to any world plane from real planes. The algorithm allows for overlapping data segmentation which permits to detect the majority of planes.

The maximum likelihood estimate of the model with respect to actual images is then computed using an appropriate structure parameterization that enforces multi-coplanarity constraints.

The use of simulated data shows that our method performs better, in terms of the number of detected planes, compared to those based on a purely geometric criterion or using a disjoint data segmentation scheme. We have also validated the method using real images. This shows a good rendering quality and demonstrates how the difficulty of approximate planarity of real planes can be overcome using approximate photoconsistency, i.e. r-consistency.

However, there are still limitations and further research to pursue. Indeed, the method depends on a judicious selection of feature points on edges and corners. This difficulty can be tackled using a matching process relying on piecewise planarity. We also plan to incorporate a model of light variation in the photoconsistency measure to specifically handle specular surfaces such as windows that often arise in practice.

References

[1] J. Alon and S. Sclaroff. An integrated approach for segmentation and estimation of planar structure. Technical report 2000-23, Boston University, December 2000.

[2] C. Baillard and A. Zisserman. Automatic reconstruction of piecewise planar models from multiple views. In *Proceedings of the Conference on Computer Vision and Pattern Recognition, Fort Collins, Colorado, USA*, pages 559–565. IEEE Computer Society Press, June 1999.

[3] A. Bartoli and P. Sturm. Constrained structure and motion from N views of a piecewise planar scene. In *Proceedings of the First International Symposium on Virtual and Augmented Architecture, VAA'01, Dublin, Ireland*, pages 195–206, June 2001.

[4] P. Beardsley, P. Torr, and A. Zisserman. 3D model acquisition from extended image sequences. In B. Buxton and R. Cipolla, editors, *Proceedings of the 4th European Conference on Computer Vision, Cambridge, England*, volume 1065 of *Lecture Notes in Computer Science*, pages 683–695. Springer-Verlag, April 1996.

[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 '96*, New Orleans*, August 1996.

[6] D. Demirdjian and R. Horaud. Motion-egomotion discrimination and motion segmentation from image-pair streams. *Computer Vision and Image Understanding*, 78(1):53–68, April 2000.

[7] A.R. Dick, P.H.S. Torr, S.F. Ruffle, and R. Cipolla. Combining single view recognition and multiple view stereo for architectural scenes. In *Proceedings of the 8th International Conference on Computer Vision, Vancouver, Canada*, 2001.

[8] M.A. Fischler and R.C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Graphics and Image Processing*, 24(6):381 – 395, June 1981.

[9] P.E. Gill, W. Murray, and M.H. Wright. *Practical Optimization*. Academic Press, 1981.

[10] K.N. Kutulakos. Approximate N-view stereo. In *Proceedings of the 6th European Conference on Computer Vision, Dublin, Ireland*, 2000.

[11] F. Lang and W. Förstner. 3D-city modeling with a digital one-eye stereo system. In *Proceedings of the XVIII ISPRS-Congress, Vienna, Austria*, July 1996.

[12] D. Liebowitz and A. Zisserman. Metric rectification for perspective images of planes. In *Proceedings of the Conference on Computer Vision and Pattern Recognition, Santa Barbara, California, USA*, 1998.

[13] D.D. Morris and T. Kanade. Image-consistent surface triangulation. In *Proceedings of the Conference on Computer Vision and Pattern Recognition, Hilton Head Island, South Carolina, USA*, 2000.

[14] F. Schaffalitzky and A. Zisserman. Planar grouping for automatic detection of vanishing lines and points. *Image and Vision Computing*, 18(9):647–658, 2000.

[15] D. Sinclair and A. Blake. Quantitative planar region detection. *International Journal of Computer Vision*, 18(1):77–91, 1996.

[16] A. Streilein and U. Hirschberg. Integration of digital photogrammetry and CAAD: Constraint-based modeling and semi-automatic measurement. In *Proceedings of the International CAAD Futures Conference, Singapore*, September 1995.

[17] P.H.S. Torr and D.W. Murray. Outlier detection and motion segmentation. In P.S. Schenker, editor, *Sensor Fusion VI*, pages 432–442, Boston, 1993. SPIE volume 2059.

[18] P.H.S. Torr and A. Zisserman. Robust computation and parametrization of multiple view relation. In *Proceedings of the 6th International Conference on Computer Vision, Bombay, India*, 1998.