Toward Automatic 3D Modeling of Scenes using a Generic Camera Model

Maxime Lhuillier

Toward Automatic 3D Modeling of Scenes using a Generic Camera Model

Maxime Lhuillier
LASMEA-UMR 6602 UBP/CNRS, 63177 Aubière Cedex, France.
maxime.lhuillier.free.fr

Abstract

The automatic reconstruction of 3D models from image sequences is still a very active field of research. All existing methods are designed for a given camera model, and a new (and ambitious) challenge is 3D modeling with a method which is exploitable for any kind of camera. A similar approach was recently suggested for structure-from-motion thanks to the use of generic camera models. In this paper, we first introduce geometric tools designed for 3D scene modeling with a generic camera model. Then, these tools are used to solve many issues: matching errors, wide range of point depths, depth discontinuities, and view-point selection for reconstruction. Experiments are provided for perspective and catadioptric cameras.

1. Introduction

The automatic reconstruction of photo-realistic 3D models of scenes from image sequences taken by a moving camera is still a very active field of research. Once the camera parameters of the image sequence are recovered by structure-from-motion, dense stereo and stereo merge into a single 3D model are successively applied. Currently, many 3D modeling systems exist for perspective cameras [12, 16, 10], catadioptric cameras [3, 9] and multi-camera rig [2] (among others) sometimes with the help of additional informations such as odometry. Even if the intrinsic parameters of the camera are unknown, the involved methods are dependent on a given camera model.

Recently, it was suggested that the first sub-problem (structure-from-motion) be solved using a generic camera model and generic tools which are exploitable for any kind of camera [5, 15], and the same challenge arises naturally for the complete 3D modeling process. The practical advantages would be obvious: a high ability to change one camera model for another or to mix different cameras (*e.g.* catadioptric camera for wide field of view and perspective camera for a few parts of the scene where higher reconstruction accuracy is needed). Many generic tools are already available for structure-from-motion: estimation of the generalized essential matrix [13, 15], pose calculation [14], bundle adjustment [17, 11] and generic camera calibration [5, 8]. We have no additional contributions for this sub-problem and assume that the camera parameters are known.

Dense stereo is the second sub-problem. It is recognized that this step is very difficult in practice for uncontrolled environments. This difficulty increases in the generic context since the use of the image projection function is prohibited (this function is specific to the kind of camera). The epipolar constraint is also unavailable since the camera may be non-central. Optical flow methods [7] remain since they do not use 3D. In this second step, a hypothesis is used to obtain better results for 3D modeling: we assume that epipolar constraints are locally available in the generic images such that the standard pair-wise stereo methods [18] may be applied after local rectifications.

The third sub-problem is the following: once cameras and matches between image pairs are known, how can a 3D model of the scene be recovered using generic tools ? This model is a list of textured triangles in 3D which approximates the visible part of the scene where the camera has moved. We have to reconstruct 3D points, approximate them by a mesh, and deal with matching errors (false negatives and false positives), depth discontinuities, and a wide range of accuracies for reconstructed points (due to close foreground and far background, or view-point selection).

1.1. Contributions and Paper Overview

Our generic camera model is slightly different from previous ones. Previous authors [5, 17] model arbitrary imaging systems by a set of virtual sensing elements called raxels: a raxel is central or perspective camera with a small part of the complete view field. In our case, a raxel is reduced to a single ray (a point origin and a direction in 3D) such that the raxel center is the ray origin. We know the calibration function, which maps image pixels to rays. This function also defines the choice of all ray origins ("the ray surface choice" [5]). A central camera is a special case, where all ray origins are the same point: the camera center.

Once the generic camera model is presented, Section 2 introduces a generic method to reconstruct points from

image matches by ray intersection, a generalization for a generic camera of virtual uncertainty [9], and a generalization for n views of the 2-view-angle reliability [4]. Both virtual uncertainty and reliability were introduced for catadioptric cameras to select reconstructed points which are retained in the final 3D model. Such selections are important if parts of the scene are to be reconstructed at very different accuracies depending on the view-point selected for reconstruction. This is also true for any other camera with a wide field of view. In this paper, both virtual uncertainty and reliability have closely related and coherent definitions, which is not the case in previous works [4, 9].

Section 3 describes how to obtain a (local) 3D model for a few generic images given their corresponding camera poses, the calibration function and point correspondences. First, a reference image is chosen and segmented by a 2D mesh using gradient edges and color information. Second, points are reconstructed by our generic ray intersection. Third, 2D triangles are back-projected in 3D to fit the reconstructed points as best possible by taking into account a wide range of point depths. Virtual uncertainty is useful here to weight the minimized scores, to define the connections between triangles in 3D, and to fill holes. Finally, triangles with the worst reliability are rejected.

Section 4 provides many experiments for central cameras. Global models are obtained by combining the local models with a simple view-point selection method [9] using our (generic) virtual uncertainty. Last, Section 5 concludes and explains what should be added for non-central cameras.

1.2. Assumptions

The proposed method involves many assumptions. First, the scene surface should be smooth enough to be approximated by a list of triangles in 3D. Second, the majority of occluding contours (and the tangent discontinuities of surfaces) should occur at gradient edges or color discontinuities in images. Third, the generic camera should not be too exotic to back-project connected image points (*e.g.* 2D triangles) to connected points in 3D (*e.g.* planar scene parts): we assume that the calibration function which maps pixels to rays in 3D is piecewise $\mathcal{C}^0$ continuous with known, smooth and polygonizable discontinuities. These discontinuities occur in practice for multi-camera rig (*e.g.* the line between two composite images in the generic image of the stereo-rig).

2. Geometric Tools for a Generic Camera

This Section presents a method to reconstruct points (Section 2.1), virtual uncertainty (Section 2.2), reliability (Section 2.3) and geometric tests (Section 2.4).

The calibration function of the camera is known and maps pixels of a generic image to optical rays. An optical ray is an oriented line defined by its origin and direction. Thanks to the knowledge of camera pose in the world coordinate system, origin $\mathbf{o}$ and direction $\mathbf{d}$ ($||\mathbf{d}|| = 1$) of this ray in the world coordinate system are also known and used throughout the paper.

2.1. Point Reconstruction by Ray Intersection

Once point correspondences in images, calibration and successive poses of the camera are given, 3D points of the scene should be reconstructed. The standard method to reconstruct a point is the minimization of a sum of square of reprojection errors in pixels using the Levenberg-Marquardt method [6] (LM). However, these errors cannot be used in the generic context since they require the image projection function, which is specific to the kind of camera. Many rays $(\mathbf{o}_i, \mathbf{d}_i)$ corresponding to observations in the i^{th} image of the 3D point $\mathbf{P}$ to reconstruct should be used instead.

One solution is the reconstruction of $\mathbf{P}$ by minimizing the sum of squares of angles α_i between vectors $\mathbf{d}_i$ and $\mathbf{P} - \mathbf{o}_i$. In practice, the definition $\alpha_i(\mathbf{P}) = \arccos(\mathbf{d}_i^\top \frac{\mathbf{P}-\mathbf{o}_i}{||\mathbf{P}-\mathbf{o}_i||})$ lead to poor LM convergence. This is not surprising: the $\mathcal{C}^2$ continuity of α_i is recommended for a good (quadratic and final) convergence of LM and it can be shown [1] that α_i is never $\mathcal{C}^1$ continuous at point $\tilde{\mathbf{P}}$ such that $\alpha_i(\tilde{\mathbf{P}}) = 0$.

Let $\mathtt{R}_i$ be a rotation such that $\mathtt{R}_i\mathbf{d}_i = [0\ 0\ 1]^\top$ and π the function $\pi([x\ y\ z]^\top) = [x/z\ y/z]^\top$. Once rays $(\mathbf{o}_i, \mathbf{d}_i)$ are given ($i \in \{1, 2, \cdots I\}$), we estimate $\mathbf{P}$ as the minimizer of

$$E(\tilde{\mathbf{P}}) = \sum_{i=1}^{I} ||\alpha_i(\tilde{\mathbf{P}})||^2 \quad \text{with } \alpha_i(\tilde{\mathbf{P}}) = \pi(\mathtt{R}_i(\tilde{\mathbf{P}} - \mathbf{o}_i)). \quad (1)$$

Now α_i is $\mathcal{C}^2$ continuous and the LM convergence is good in practice. Furthermore, $||\alpha_i(\mathbf{P})||$ is the tangent of the angle between $\mathbf{d}_i$ and $\mathbf{P} - \mathbf{o}_i$. The tangent is a good angle approximation near the expected solution where the angles are small. $\mathbf{P}$ is retained if it is in front of the cameras (i.e. $\mathbf{d}_i^\top(\mathbf{P} - \mathbf{o}_i) > 0$) and if $E(\mathbf{P})/I$ is less than a threshold.

2.2. Virtual Covariance and Uncertainty

Assume that angle errors α_i defined in Eq. 1 follow independent and identical Gaussian noise $\mathcal{N}(\mathbf{0}_{2\times1}, \sigma_\alpha^2 \mathtt{I}_{2\times2})$. Let J be the Jacobian of the function $\tilde{\mathbf{P}} \mapsto [\alpha_1^\top\ \cdots\ \alpha_I^\top]^\top$. This noise propagates to a Gaussian noise for the estimated parameter $\mathbf{P}$ with standard covariance matrix [6]

$$C(\mathbf{P}) = \sigma_\alpha^2 (J(\mathbf{P})^\top J(\mathbf{P}))^{-1}. \quad (2)$$

An estimate of σ_α^2 is obtained from residuals $E(\mathbf{P})$ for all 3D reconstructed points $\mathbf{P}$.

Now, assume that a point $\mathbf{P}' \in \mathbb{R}^3$ and many ray origins $\mathbf{o}'_i \in \mathbb{R}^3, i \in \{1, 2, \cdots I\}$ are given. Let $\mathbf{d}'_i$ be the direction $\mathbf{d}'_i = \frac{\mathbf{P}'-\mathbf{o}'_i}{||\mathbf{P}'-\mathbf{o}'_i||}$. We can solve the minimization problem defined by Eq. 1 for the new rays $(\mathbf{o}'_i, \mathbf{d}'_i)$ instead

of rays $(\mathbf{o}_i, \mathbf{d}_i)$ and obtain the minimizer $\mathbf{P}$ of E with its standard covariance $C(\mathbf{P})$ in Eq. 2. However,

$$\alpha_i(\mathbf{P}') = \pi(\mathtt{R}_i(\mathbf{P}' - \mathbf{o}'_i)) = \pi(||\mathbf{P}' - \mathbf{o}'_i||\mathtt{R}_i\mathbf{d}_i) = 0$$

and we conclude that $E(\mathbf{P}') = 0$. Since the E minimizer is unique (if $\mathbf{P}'$ and the $\mathbf{o}'_i$ are not collinear points), we obtain $\mathbf{P}' = \mathbf{P}$. Thus, the "virtual covariance matrix" of $\mathbf{P}'$ for ray origins $\mathbf{o}'_i$ is defined by $C(\mathbf{P}')$ in Eq. 2.

At this point, the expression "virtual covariance matrix" is clearer: $C(\mathbf{P}')$ is the covariance matrix obtained by reconstructing $\mathbf{P}'$ from "virtual" rays $(\mathbf{o}'_i, \mathbf{d}'_i)$ using LM, i.e. rays which are not observation rays. In the special case where $\mathbf{P}'$ was reconstructed before by LM from other rays $(\mathbf{o}_i, \mathbf{d}_i)$ corresponding to real observations in images, the corresponding standard covariance is similar to the virtual covariance if $\mathbf{o}_i \approx \mathbf{o}'_i$ and $\mathbf{d}_i \approx \mathbf{d}'_i$.

Finally, the virtual uncertainty $U(\mathbf{P}')$ is defined by the length of the major semi-axis of the uncertainty ellipsoid defined by $C(\mathbf{P}')$ and a probability p. This ellipsoid is

$$\Delta\mathbf{x}^\top C^- \Delta\mathbf{x} \leq \mathcal{X}_3^2(p), \quad C^- = \frac{1}{\sigma_\alpha^2}\sum_{i=1}^{I} \frac{\mathtt{I}_{3\times 3} - \mathbf{d}'_i\mathbf{d}'^\top_i}{||\mathbf{P}' - \mathbf{o}'_i||^2} \quad (3)$$

with $\mathcal{X}_3^2(p)$ the quantile function of the $\mathcal{X}^2$ distribution with 3 d.o.f, p a probability and C^- the inverse [1] of $C(\mathbf{P}')$. Using notation e for the smallest eigenvalue of C^-, we have

$$U(\mathbf{P}') = \sqrt{\frac{\mathcal{X}_3^2(p)}{e}}. \quad (4)$$

2.3. Reliability for 3D Modeling of a Scene

A point $\mathbf{P}$ reconstructed from observation rays $(\mathbf{o}_i, \mathbf{d}_i), i \in \{1, 2, \cdots I\}$ may be so inaccurate that the 3D model should not contain it. At first glance, we can decide that $\mathbf{P}$ is inaccurate for 3D modeling if $U(\mathbf{P})$ is larger than a given threshold U_0. In this Section, we introduce and justify a reliability definition $R(\mathbf{P})$ which is more adequate than $U(\mathbf{P})$ for thresholding.

If the generic camera is a central camera, the reconstruction is defined up to a global 3D scale and a scale change of the whole reconstruction (3D points and camera centers) implies the same scale change of the uncertainties. For a central camera, the threshold U_0 must be proportional to the scene scale to obtain a decision which is independent of the scale. This is a first reason to define

$$R(\mathbf{P}) = \frac{U(\mathbf{P})}{\min_i ||\mathbf{P} - \mathbf{o}_i||} \quad (5)$$

and decide that $\mathbf{P}$ is inaccurate for 3D modeling if $R(\mathbf{P})$ is larger than a given threshold R_0.

We see that the permitted maximal uncertainty by condition $R(\mathbf{P}) < R_0$ is proportional to the distance between point $\mathbf{P}$ and ray origins $\mathbf{o}_i$. More precisely, this inequality allows points with good accuracy (for 3D modeling of the scene) to have greater uncertainties if they are a long distance from the ray origins $\mathbf{o}_i$ and smaller uncertainties if they are close. As a consequence, we can expect to modelize both close foreground and far background of the scene. This is the second reason for this definition of $R(\mathbf{P})$. Furthermore, through this inequality it is possible to moderate the ellipsoid size (uncertainty $U(\mathbf{P})$) in comparison with the distance between ellipsoid center (the reconstructed point $\mathbf{P}$) and ray origins $\mathbf{o}_i$. It is not difficult to prove [1] that $R(\mathbf{P})$ is arbitrarily large in two cases: (1) nearly parallel $\mathbf{d}_i$ and (2) large values of $||\mathbf{P} - \mathbf{o}_i||$. Case (1) occurs if all $\mathbf{o}_i$ are collinear points and $\mathbf{P}$ goes toward the line of the $\mathbf{o}_i$. Case (2) occurs for distant point $\mathbf{P}$. These cases should be avoided for 3D modeling. This is a third reason for this definition of $R(\mathbf{P})$.

2.4. Geometric Tests

Let Π be the plane $\mathbf{n}^\top\mathbf{X} + d = 0$. Once virtual covariance matrix C is defined, Mahalanobis point-to-point and point-to-plane [1] squared distances are respectively

$$\begin{aligned} d^2(\mathbf{P}_1, \mathbf{P}_2) &= (\mathbf{P}_1 - \mathbf{P}_2)^\top C^{-1}(\mathbf{P}_1)(\mathbf{P}_1 - \mathbf{P}_2) \\ d^2(\mathbf{P}_1, \Pi) &= \min_{\mathbf{P}_2 \in \Pi} d^2(\mathbf{P}_1, \mathbf{P}_2) = \frac{(\mathbf{n}^\top\mathbf{P}_1 + d)^2}{\mathbf{n}^\top C(\mathbf{P}_1)\mathbf{n}} \end{aligned} \quad (6)$$

Here we introduce several tests which are systematically used by mesh operations for 3D modeling. The point-to-point neighborhood test $T(\mathbf{P}_1, \mathbf{P}_2)$ is true if $d^2(\mathbf{P}_1, \mathbf{P}_2) \leq \mathcal{X}_3^2(p)$ and $d^2(\mathbf{P}_2, \mathbf{P}_1) \leq \mathcal{X}_3^2(p)$. The point-to-plane neighborhood test $T(\mathbf{P}_1, \Pi)$ is true if $d^2(\mathbf{P}_1, \Pi) \leq \mathcal{X}_3^2(p)$. The planarity test $T(\{\mathbf{P}_i\})$ is true if there is a plane Π such that all $T(\mathbf{P}_i, \Pi)$ are true. In practice, Π is estimated by random samples of 3 points in the list $\{\mathbf{P}_i\}$.

These tests implicitly requires for each 3D point $\mathbf{P}_i$ the corresponding ray origins due to the virtual covariance definition in Section 2.2. If the generic camera is central or if the points are reconstructed by LM, the ray origins are known. They are unknown in other cases, unless we apply the projection functions to $\mathbf{P}_i$ (but this is not a generic method). More investigations are needed to estimate efficiently ray origins in the non-central case.

3. 3D Model from Generic Images

This Section describes how to obtain a 3D model for a few generic images given their corresponding camera poses, the calibration function and image point correspondences.

First, a reference image is chosen and segmented by a 2D mesh using gradient edges and color informations (Section 3.1). Second, points are reconstructed by intersection of observation rays as described in Section 2.1. Third, 2D triangles are back-projected in 3D to fit the reconstructed

points by taking into account 3D point uncertainties and depth discontinuities (Section 3.2). Last, the most unreliable parts of the resulting 2.5D mesh (Section 3.3) are rejected. Assumptions are given in Section 1.

3.1. 2D Mesh

The 2D mesh in the generic reference image should satisfy many contradictory constraints: gradient edges at mesh edges, small enough mesh edges for good approximation of gradient edges, large enough mesh triangles for stable estimation of triangles in 3D and efficient rendering, uniform sampling of the field of view, and good aspect ratio for triangles. A compromise is obtained as follows.

Mesh Initialization First, a Delaunay triangulation is initialized such that the solid angles of any triangles are roughly the sames. In practice, simple checkerboards with two triangles for each rectangular cell are good enough for standard cameras like perspective, catadioptric, or stereorig. The $\mathcal{C}^0$ discontinuities of the calibration function define the borders of independent 2D meshes in the reference image. Borders enforce constrained edges on the Delaunay and enforce the global shape of the checkerboards (in the catadioptric case, cell rows are concentric rings and cell columns are radial sections). The mesh resolution is defined by a mean length of cell edges equal to 8 pixels.

Gradient Edge Integration Second, the gradient edges are integrated in the mesh by moving mesh vertices slightly and forcing mesh edges to be constrained. We have not taken into account all gradient edges since the mesh resolution has been previously fixed. So they are integrated in a best first order. A contour is a list of connected pixels which have maximum local image gradient. Its score is equal to the sum of gradient modulus for all its pixels. We pick the contour with the highest score, and find the list of closest vertices to its pixels such that the vertices have not been used before for any other contour. Then two consecutive vertices are moved slightly to approximate the contour if the part of the contour between vertex ends is a segment. Once all contours have been considered by decreasing score, a completion step is used in order to try to constrain new mesh edges if they approximate a contour in their immediate neighborhood.

Mesh Refinement Third, the 2D mesh is refined by alternating continuous improvements (move vertices to minimize a global cost combining color variance in triangles and mesh smoothness) and discrete improvements (flip edges and merge vertices to improve aspect ratio of triangles).

The continuous mesh improvement is useful for many reasons. First, few (parts of) gradient edges may be missed by the previous step, and minimizing the sum of color variances for each triangle is an other way to increase the probability that the gradient edges are on the mesh edges. Second, the gradient edge integration deformed the initial mesh only locally such that the constraint of a same solid angle for all triangles is highly violated. Minimizing the mesh smoothness (sum of squared modulus of an umbrella operator) is a way to incite incident triangles to have similar solid angles. Minimizing the mesh smoothness is also useful to improve triangle aspect ratio and regularize the minimization of color variance.

The cost function is defined by

$$e_{2d}(\{p_v\}) = \sum_{p \in t \in T} ||c_p - \sum_{p' \in t} \frac{c_{p'}}{|t|}||^2 + \lambda \sum_{v \in V} || \sum_{v' \in \mathcal{N}_v} p_v - p_{v'}||^2$$

with T the list of mesh triangles, $|t|$ the area of triangle t, V the list of mesh vertices, $\mathcal{N}_v$ the list of vertices which are connected to v by a mesh edge. Color c_p at pixel p is RGB, p_v is the image location of vertex v and λ is equal to 1000. The cost function is minimized using a simple descent method with vertex locations $\{p_v\}$ as parametrization. All mesh vertices are allowed to move in 2D, except vertices which are incident to a constrained edge (vertices at gradient edges). The latter are only allowed to move in 1D along the detected gradient edges. This gives the priority to detected gradient edges over the minimization of color variance, which may sometimes be contradictory.

3.2. 2.5D Mesh

Assume that we have $I \geq 2$ camera poses and a dense list of 3D points $\mathbf{P}$ reconstructed by intersection of I observation rays (one for each pose) as described in Section 2.1. A 2D mesh in a reference image is also given.

First, the 2.5D mesh is initialized as a list of fully disconnected triangles in 3D. Then, this mesh is refined by alternating discrete improvements "Triangle Connection", "Hole Filling", "Triangle Removal", "Triangle Damping" and continuous improvements "Mesh Refinement". These mesh improvements are defined below thanks to the virtual covariance for the I poses (Sections 2.2 and 2.4).

At any step, the 2.5D mesh in 3D is a back-projection of the 2D mesh in the reference image. In other words, each triangle t^{2d} of the 2D mesh corresponds to (at most) one triangle t^{3d} of the 2.5D mesh with vertices $\mathbf{v}_i \in \mathbb{R}^3$. The t^{3d} vertices are parameterized by depths $z_i > 0$ such that $\mathbf{v}_i = \mathbf{o}_i + z_i \mathbf{d}_i$ with $(\mathbf{o}_i, \mathbf{d}_i)$ the observation rays of t^{2d} vertices. Vertices in the 2D mesh may have many depths depending on current connections between triangles in 3D.

Mesh Initialization In this step, each triangle t^{2d} of the 2D mesh is individually back-projected to fit the 3D points as best possible with a RANSAC procedure.

First, all 3D points reconstructed from matched pixels inside t^{2d} are collected in a list $L_{t^{2d}}$. Second, planes are calculated for random samples of three 3D points taken in

$L_{t^{2d}}$. Let Π be the plane minimizing

$$E^2_{t^{2d}} = \sum_{\mathbf{P} \in L_{t^{2d}}} \min\{\mathcal{X}^2_3(p), d^2(\mathbf{P}, \Pi)\} \quad (7)$$

with $\mathcal{X}^2_3(p)$ and $d(\mathbf{P}, \Pi)$ introduced in Eq. 3 and 6.

Then, we estimate depths z_i at the 3 vertices of t^{2d} such that $\mathbf{o}_i + z_i\mathbf{d}_i \in \Pi$ with $(\mathbf{o}_i, \mathbf{d}_i)$ the observation rays of these vertices. The triangle in 3D with vertices $\mathbf{o}_i + z_i\mathbf{d}_i$ is added in the 2.5D mesh if $z_i > 0$.

Pair-Wise Triangle Connection Triangles in 3D should be interconnected to obtain a more realistic 3D model.

Let t^{3d}_a and t^{3d}_b be two 3D triangles such that the associated triangles t^{2d}_a and t^{2d}_b in the 2D mesh have a common edge (t^{2d}_a and t^{2d}_b are "weakly" connected). This edge has two vertices 0 and 1 in 2D, which correspond to triangle vertices $\{\mathbf{v}^a_0, \mathbf{v}^b_0\}$ and $\{\mathbf{v}^a_1, \mathbf{v}^b_1\}$ in 3D. The connection between t^{3d}_a and t^{3d}_b is effective if the point-to-point neighborhood tests $T(\mathbf{v}^a_0, \mathbf{v}^b_0)$ and $T(\mathbf{v}^a_1, \mathbf{v}^b_1)$ defined in Section 2.4 are true.

The connection between t^{3d}_a and t^{3d}_b is defined as follows. Let z^a_i and z^b_i be depths such that $\mathbf{v}^a_i = \mathbf{o}_i + z^a_i\mathbf{d}_i$ and $\mathbf{v}^b_i = \mathbf{o}_i + z^b_i\mathbf{d}_i$ with $(\mathbf{o}_i, \mathbf{d}_i)$ the observation rays of 2D vertices $i \in \{0, 1\}$. New values of z^a_i and z^b_i are set to former value of $\frac{1}{2}(z^a_i + z^b_i)$. Henceforth, the 2.5D mesh parameters z^a_i and z^b_i are linked by constraints $z^a_i = z^b_i$ for further processing.

Group-Wise Triangle Connection The "Pair-Wise Triangle Connection" above connects any triangle pair in 3D if they satisfy neighborhood conditions. Here we introduce the "Group-Wise Triangle Connection", which connects any k-group of triangles in 3D if they satisfy a planarity condition (typically $k \in \{2, 3, 4\}$).

A k-group of triangles in 3D is a list of k triangles t^{3d}_j such that the corresponding triangles t^{2d}_j are "strongly" connected in the 2D mesh. Two triangles are strongly connected if they have a common edge which is not constrained in the 2D mesh. We avoid constrained edges since they are potential surface discontinuities in 3D. Section 2.4 defines the planarity condition by $T(\{\mathbf{v_i}\})$ with $\{\mathbf{v_i}\}$ the list of all triangle vertices of the k-group.

Any triangle pair $\{t^{3d}_a, t^{3d}_b\}$ in 3D is connected as in the pair-wise case if it is included in a k-group satisfying the planarity condition and if the corresponding t^{2d}_a, t^{2d}_b in 2D have a common edge.

Triangle Removal A smooth surface is expected to be approximated by a list of connected triangles in 3D. If a triangle is not connected to (at least) one of its neighbors after trials of triangle connections, we have some doubt as to its quality and may decide to remove it from the 2.5D mesh. They are many reasons for fully disconnected and bad triangles in 3D: false positive matches in images (*e.g.* in the neighborhood of occluding contours), triangle estimations using 3D points in both close foreground and far background, too few points for reliable estimation.

Triangle Damping The main drawback of "Triangle Removal" is the lack of triangles in scene parts which are not smooth such as tree foliages. If a triangle t^{3d} without connection is not removed, it may produce a major degradation of visual quality if it is very stretched in 3D in the direction $\mathbf{d}_i$ of rays which goes across t^{3d} vertices. In this case, the angle θ between t^{3d} normal $\mathbf{n}$ and $\mathbf{d}_i$ is greater than a threshold θ_0.

Thus, "Triangle Damping" reduces such degradations as follows: if $\theta_0 < \theta$, the t^{3d} depths z_i are disturbed such that (1) the t^{3d} center is fixed and (2) $\mathbf{n}$ is replaced by $cos(\theta_0)\mathbf{d}_i + sin(\theta_0)\frac{\tilde{\mathbf{d}}_i}{||\tilde{\mathbf{d}}_i||}$ with $\tilde{\mathbf{d}}_i = \mathbf{n} - (\mathbf{n}^\top \mathbf{d}_i)\mathbf{d}_i$. "Triangle Damping" may be preferred to "Triangle Removal" to obtain more triangles in the 3D model.

Hole Filling In our context, a hole is a connected component of triangles t^{2d}_j in the 2D mesh without corresponding triangles t^{3d}_j in the 2.5D mesh. "Hole Filling" is the definition of the lacking t^{3d}_j by interpolation of depths available in the hole border. Holes are mainly due to false negative matches in low textured areas and degrade the visual quality of 3D model rendering if they are not properly filled.

The main risk is depth interpolation between foreground and background which also degrades the rendering quality, especially if foreground and background have different colors. We have the choice between strong connectivity (used in "Group-Wise Triangle Connection") and weak connectivity (used in "Pair-Wise Triangle Connection") between two triangles in the 2D mesh to define a hole as a connected component. The former is preferred to the latter which includes potential surface discontinuities at constrained edges too easily in the hole. As a consequence, the hole border is a list of edges in the 2D mesh such that (1) edges are constrained or (2) edges are not constrained and have depths at their two vertices. All 3D points corresponding to these vertices with depths are collected in a list $\{\mathbf{v_i}\}$. We also define r as the ratio between the sum of 2D lengths of edges of type (2) and the sum of 2D lengths of all border edges.

To obtain a well defined interpolation and reduce the risk of depth interpolation between foreground and background, we require that the hole border is planar thanks to the planarity condition $T(\{\mathbf{v_i}\})$ defined in Section 2.4. We also request enough 3D information at the hole border by r thresholding ($0.5 < r$). If $T(\{\mathbf{v_i}\})$ is true, there is a plane Π which approximates the $\mathbf{v_i}$ and "Hole Filling" is defined as follow. Each vertex in the hole (including border) has a corresponding observation ray $(\mathbf{o}_i, \mathbf{d}_i)$ and a depth z_i defined by $\mathbf{o}_i + z_i\mathbf{d}_i \in \Pi$. Any hole triangle t^{2d} with positive z_i at its vertices defines a new triangle t^{3d} in the 2.5D mesh.

Depth constraints are set for further processing such that these vertices have only one depth.

Mesh Refinement The parameters of the 2.5D mesh is the list of depths z_i for each triangle vertex in 3D with many constraints (equalities) between the z_i. Improvements "Hole Filling" and "Pair/Group-Wise Triangle Connection" are useful to increase the rendering quality of the 3D model, but they reduce the number of independent z_i and disturb the initial values of z_i obtained from the 3D point cloud. The consequence is an increasing discrepancy between 3D points and the 2.5D mesh. This problem is reduced by minimizing a global cost function including a discrepancy term and a smoothness term. The smoothness term is useful to reduce noise and enforce a prior knowledge of a smooth surface on the 2.5D mesh.

The cost function to minimize is defined by

$$e_{3d}(\{z_i\}) = \sum_{t \in T} E_t^2 + \lambda \sum_{\{t_1,t_2\} \in E} \frac{1}{2}(|t_1|+|t_2|)(\mathbf{n}_{t_1} - \mathbf{n}_{t_2})^2$$

with T the list of 2D mesh triangles which has a triangle in 3D, $\{t, t_1, t_2\} \subset T$, $\{t_1, t_2\}$ the edge between triangles t_1 and t_2, $|t|$ the surface (in pixels) of t, E the list of unconstrained edges in the 2D mesh, and $\mathbf{n}_t$ the normal of the 3D triangle corresponding to the 2D triangle t. Weight λ is equal to 1 and E_t^2 is defined in Eq. 7. The cost function is minimized by a descent method with depths $\{z_i\}$ as parametrization. Depths have a wide range due to close foreground and far background, and this should be taken into account to reduce the cost efficiently. Given a depth value z_i^n at iteration n of the descent method, we choose the value $z_i^{n+1} \in \{z_i^n - \delta_i(z_i^n), z_i^n, z_i^n + \delta_i(z_i^n)\}$ which minimizes the partial function $z_i \mapsto e_{3d}(z_i)$. Virtual uncertainty is used to scale the increment δ_i by $\delta_i(z) = \epsilon U(\mathbf{o}_i + z\mathbf{d}_i)$ with $\epsilon = 0.02$.

Algorithm Summary Many combinations of the mesh operations above are possible and have been the subject of experiments. Our favorite strategy currently is

1. Mesh Initialization
2. apply Group-Wise Triangle Connection ($k = 4$), Hole Filling and Mesh Refinement alternatively
3. Triangle Removal or Triangle Damping ($\theta_0 = \frac{7}{20}\pi$)
4. apply Pair-Wise Triangle Connection, Hole Filling and Mesh Refinement alternatively.

Step 2 merges triangles with strong conditions before step 3. Once step 3 has removed improbable and unconnected triangles, step 4 connects triangles with weaker conditions.

3.3. Unreliable Parts

Once the 2.5D mesh is obtained, the reliability for 3D modeling (Eq. 5) allows the detection of unreliable vertices $\mathbf{v}_i$ by thresholding such as $R_0 < R(\mathbf{v}_i)$. Any triangle which has an unreliable vertex is removed.

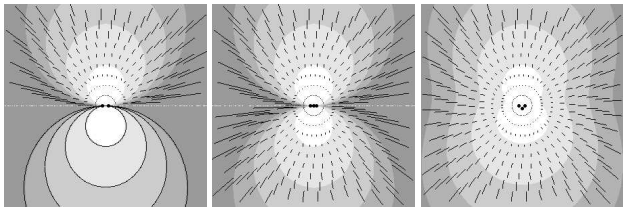

Figure 1. Virtual uncertainty U and reliability R in a plane for a generic (central) camera in three cases: two camera poses on the left, three collinear poses in the middle, and three non collinear poses on the right. Camera poses are black points in this plane. Black edges are the main axes of uncertainty ellipsoids centered at some points $\mathbf{P}$ (their length is $2U(\mathbf{P})$. Every point $\mathbf{P}$ has a gray level depending on the interval where $R(\mathbf{P})$ lies: $[0, \frac{1}{40}[, [\frac{1}{40}, \frac{2}{40}[, [\frac{2}{40}, \frac{3}{40}[, [\frac{3}{40}, \frac{4}{40}[, [\frac{4}{40}, +\infty]$ (darkest gray levels for largest reliabilities). On the left, we check that curves $R(\mathbf{P}) = R_0$ with $R_0 \in \{\frac{1}{40}, \frac{2}{40}, \frac{3}{40}, \frac{4}{40}\}$ are very similar to circles (in black). Typical values are obtained for U and R with $\sigma_\alpha = 0.001$ radian and $\mathcal{X}_2^3(0.9) = 6.25$.

4. Experiments

Once camera poses and matching between image pairs are given, all experiments are done for the generic camera model restricted to central cameras (ray origins at the centre). Actually, specific pose methods [10, 9] are prefered to generic method [11] since they also estimate calibration and have more successful automatic matching. Furthermore, the dense matching method between image pairs involves local rectifications which are specific to camera model [9].

4.1. Properties of $U(\mathbf{P})$ and $R(\mathbf{P})$

In the first case (on the left of Figure 1), $U(\mathbf{P})$ and $R(\mathbf{P})$ are shown for two camera locations or ray origins $\mathbf{A}$ and $\mathbf{B}$. $U(\mathbf{P})$ and $R(\mathbf{P})$ are defined everywhere (except on the line defined by $\mathbf{A}$ and $\mathbf{B}$). Due to the symmetry of the problem, U and R are the same for any plane in 3D containing $\mathbf{A}$ and $\mathbf{B}$. As expected, they increase in two cases: (1) if $\mathbf{P}$ goes toward line defined by $\mathbf{A}$ and $\mathbf{B}$ or (2) if $\mathbf{P}$ goes long away from $\mathbf{A}$ and $\mathbf{B}$. We also see at the bottom that our reliability is very similar to the reliability given in [4]: curves implicitly defined by $R(\mathbf{P}) =$ constant are very similar to circles defined by $angle(\mathbf{A}, \mathbf{P}, \mathbf{B}) =$ constant.

In the second case (in the middle), a camera pose $\mathbf{C}$ is added in the middle of $\mathbf{A}$ and $\mathbf{B}$. The result is unexpected: there is no improvement (i.e. U or R decrease) by adding the third camera pose. In fact, the results are nearly the same. In the third case (on the right), $\mathbf{C}$ is moved toward the bottom. As expected, the improvement is noticeable in the neighborhood of the line defined by $\mathbf{A}$ and $\mathbf{B}$. In these two last cases, our R definition is naturally derived from U for any numbers of views. This was not the case for the R definition of [4], which was only defined for two views.

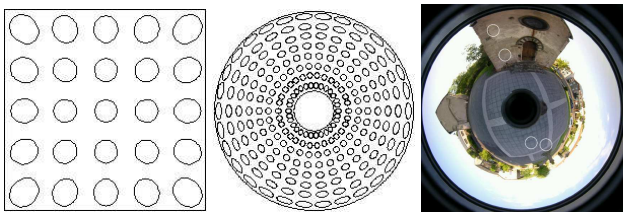

Figure 2. Image projections of circular cones (with apex at camera center) for perspective (left) and catadioptric (middle) cameras. The former is a 35mm camera with a 30mm lens. The latter is equiangular and has a field of view of $\pm 50^\circ$ above and below the plane orthogonal to the symmetry axis. Cone apertures are $\frac{\pi}{25}$ radian. An image taken by this catadioptric camera is also shown.

4.2. Comparing Specific and Generic Cameras

It is wished that the virtual covariance obtained from the generic error (angle) be the same as the virtual covariance obtained from the specific error (image reprojection). We prove a simple condition [1] in image space to check this.

Assume that a point $\mathbf{P}$, a specific camera model and I camera poses are given. We also consider any point $\mathbf{X}$ such that the image projection $p_i(\mathbf{X})$ is in the immediate neighborhood of $p_i(\mathbf{P})$ in the i-th image. The condition is

$$||\alpha_i(\mathbf{X})|| = \frac{\sigma_\alpha}{\sigma_p}||p_i(\mathbf{X}) - p_i(\mathbf{P})|| + o(||p_i(\mathbf{X}) - p_i(\mathbf{P})||) \quad (8)$$

with α_i defined from camera center $\mathbf{o_i}$ and direction $\mathbf{d_i} = \frac{\mathbf{P}-\mathbf{o_i}}{||\mathbf{P}-\mathbf{o_i}||}$ as described in Eq. 1. In other words, the projections of all circular cones (with apex at camera center) of aperture 2ϵ radians should be circles of radius $\epsilon\frac{\sigma_p}{\sigma_\alpha}$ pixels.

Figure 2 draws some of these projections for perspective and catadioptric cameras. In both cases, we note that the main differences between specific and generic virtual covariances occur at the borders of view fields where the circles have the largest distortions.

4.3. 3D Model from Catadioptric Images

The field of view and an image taken by the catadioptric camera are also shown in Figure 2. The definition of the calibration is completed by the radii of the large and small circles: 563 and 116 pixels respectively. The image sequence has 208 images (closed turn around a church). Once the camera parameters and dense matching between image pairs are estimated, each local model is reconstructed from 3 consecutive views with $\sigma_\alpha = 0.0011$ radian and $\mathcal{X}^2_3(0.9) = 6.25$ as described in Section 3.

Figure 3 shows 3D models obtained with this sequence. The local model in the first row is obtained with the reference image given in Figure 2. The most unreliable parts drawn on the left are discarded in the middle with $R_0 = 0.08$. A part of the ground (circular hole) and the upper part of the facade are in the blind cones defined by the small and large circles of the catadioptric images and can not be reconstructed for this reason. The second row shows views of an other local model. In both examples, we see that a successful gradient edge integration in the meshes allows sharp modeling of C^0 and C^1 depth discontinuities (in spite of the low resolution of catadioptric images).

The third row of Figure 3 shows a top view and a height map of the global model. The global model is obtained from 208 local models around the church as follows. Let $U_l(\mathbf{P})$ be the virtual uncertainty defined at point $\mathbf{P}$ with ray origins defined by the centers of cameras of a local model l. A triangle of local model l_0 is retained in the global model if $U_{l_0}(\mathbf{P}) \leq \beta \min_l U_l(\mathbf{P})$ with $\beta = 1.1$ and $\mathbf{P}$ a vertex of the triangle. In other words, the triangle is retained if l_0 provides one of the best (smallest) virtual uncertainties available from all local models [9]. This condition ignores the visibility of $\mathbf{P}$ in the images since $U_l(\mathbf{P})$ is well-defined everywhere in the generic context. In practice, the result is improved by taking into account the visibility as follows: we reset $U_l(\mathbf{P}) = +\infty$ if $\mathbf{P}$ is not in the view fields of l. The global model contains 567757 triangles.

The video shows a walkthrough in the scene. The reconstruction is difficult in several parts including trees and low textured areas (*e.g.* street parts) which are not filled. Furthermore, the simple triangle selection above has two weakness referenced in [9]: the model redundancy increases with depth and the self-occlusions of the surface are ignored. We have noted that the triangle selection confines the case (1) of bad reliability (Section 2.3) at the ends of image sequences, and case (1) does not occur for a closed sequence like this. Here we choose to not reject the most unreliable areas and include the far background (case (2) of bad reliability).

4.4. 3D Model from Perspective Images

Figure 4 shows results of the method applied to 28 (816×1088) images taken by a perspective camera with a lateral motion. The focal length and radial distortion estimations are $f = 1234$ and $\rho = -0.073$. Local and global models are reconstructed as in the catadioptric case with $\sigma_\alpha = 0.00059$ radian, $\beta = 1.02$ and $R_0 = 0.05$. The global model has 129635 triangles. Table 1 provides an estimate of the 3D noise for a few planes of the global models. The perspective camera has smaller noise than the catadioptric camera.

5. Conclusions

This paper presents geometric tools and results for 3D scene modeling using a generic camera model. First, virtual uncertainty and reliability are extended for generic cameras and compared with those of previous works. Second, these tools are systematically applied in the 3D model generation: fit and connect triangles in 3D, fill the holes due to matching errors, set depth resolution for 2.5D mesh optimization, re-

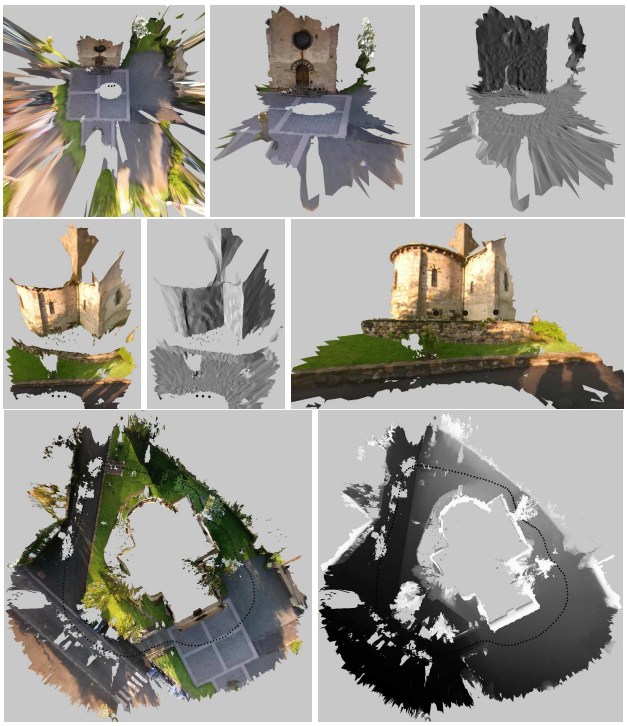

Figure 3. Several views of 3D models obtained with the catadioptric camera. Top and middle: views of two local models (3 consecutive poses) with rejection of unreliable triangles ($R_0 = 0.08$), except at the top left corner. Triangle orientations are also drawn using gray levels. Bottom: top view and height map of the global model (208 poses) with rejection ($R_0 = 0.05$).

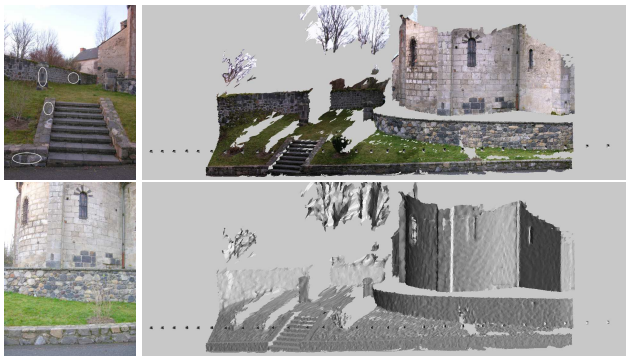

Figure 4. Left: two images (among 28) taken by the perspective camera. Right: the global 3D model.

ject the most unreliable triangles, and view-point selection to obtain a global model. Finally, 3D models of a scene are obtained for both perspective and catadioptric cameras.

A problem should be solved to apply the method on non-central camera naturally: the ray origin calculation for a 3D point expressed in the camera coordinate system. Future works also include efficient matching methods in the generic context.

Camera	per.	per.	per.	per.	cat.	cat.	cat.	cat
Vertex	208	74	153	139	78	71	405	157
Depth	188	246	358	524	212	283	319	358
RMS	0.23	0.37	0.55	0.74	0.92	0.98	1.77	1.21

Table 1. 3D noise for planar parts of global models. Each column provides information about a part: camera (perspective or catadioptric), number of vertices, mean distance between vertex and closest camera (cm), RMS of distances between vertex and estimated plane (cm). Vertices are selected for a part if they are projected in an ellipse of an image (white ellipses in Fig. 2 and 4). Distance between two consecutive camera poses is about 30 cm.

References

[1] File proofs.pdf in the supplementary material.

[2] A. Akbarzadeh, J. Frahm, P. Mordohai, B. Clipp, C. Engels, D. Gallup, P. Merell, 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 *3DPTV'06*.

[3] R. Bunschoten and B. Krose. Robust scene reconstruction from an omnidirectional vision system. *IEEE Transactions on Robotics and Automation*, pages 351–357, 2003.

[4] P. Doubek and T. Svoboda. Reliable 3d reconstruction from a few catadioptric images. In *OMNIVIS'02*.

[5] M. Grossberg and S. Nayar. A general imaging model and a method for finding its parameters. In *ICCV'01*.

[6] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000.

[7] D. F. J. Barron and S. Beauchemin. Performance of optical flow techniques. *IJCV*, 12(1), 1992.

[8] J. Kannala and S. Brandt. A generic camera model and calibration method for conventional, wide-angle and fish-eye lenses. *IEEE PAMI*, 28(8), 2006.

[9] M. Lhuillier. Toward flexible 3d modeling using a catadioptric camera. In *CVPR'07*.

[10] M. Lhuillier and L. Quan. A quasi-dense approach to surface reconstruction from uncalibrated images. *IEEE PAMI*, 27(3), 2005.

[11] E. Mouragnon, M. Lhuillier, M. Dhome, F. Dekeyser, and P. Sayd. Generic and real time structure from motion. In *BMVC'07*.

[12] D. Nister. *Automatic Dense Reconstruction from Uncalibrated Video Sequence.* PhD thesis, Royal Institute of Technology KTH, Stockholm, Sweden, 2001.

[13] D. Nister. An efficient solution for the five-point relative pose problem. *IEEE PAMI*, 26(6), 2004.

[14] D. Nister and H. Stewenius. A minimal solution to the generalized 3-point pose problem. *JMIV*, 27(1), 2007.

[15] R. Pless. Using many cameras as one. In *CVPR'03*.

[16] M. Pollefeys, L. V. Gool, M. Vergauwen, F. Verbiest, K. Cornelis, J. Tops, and R. Koch. Visual modeling with a hand-held camera. *IJCV*, 59(3), 2004.

[17] S. L. S. Ramalingam and P. Sturm. A generic structure-from-motion framework. *CVIU*, 103(3), 2006.

[18] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *IJCV*, 47(2), 2002.