Calibration of Hand-held Camera Sequences for Plenoptic Modeling

Abstract

In this contribution we focus on the calibration of very long image sequences from a hand-held camera that samples the viewing sphere of a scene. View sphere sampling is important for plenoptic (image-based) modeling approaches that try to capture the appearance of a scene by storing images from all possible directions. The plenoptic approach is appealing since it allows in principle fast scene rendering of scenes with complex geometry and surface reflections, without the need for an explicit geometrical scene model. However, the acquired images have to be calibrated before plenoptic modeling can be used, and current approaches mostly use pre-calibrated acquisition systems. This limits the generality of the approach.

We propose a way out by using an uncalibrated hand-held camera only. The image sequence is acquired by simply waving the camera around the scene objects, creating a zigzag scan path over the viewing sphere. We extend the sequential camera tracking of an existing structure-from-motion approach to a simultaneous multi-viewpoint camera tracking. A mesh of camera viewpoints is computed that approximates the view sphere. The geometry and topology of the viewpoint mesh is computed automatically from the image sequence by weaving the sequential zigzag path into a connected viewpoint mesh. The viewpoint mesh is then used for view-dependent rendering. Novel views are generated by piecewise mapping and interpolating the new image from the nearest viewpoints according to the viewpoint mesh. Local surface geometry can further enhance the interpolation process. Extensive experiments with ground truth data and hand-held sequences confirm the performance of our approach.

Keywords: Sequence tracking, Camera calibration, Structure from Motion, Plenoptic modeling

1 Introduction

There is an ongoing debate in the computer vision and graphics community between geometry-based and image-based scene reconstruction and visualization methods. Both methods aim at realistic capture and fast visualization of 3D scenes from image sequences.

Image-based rendering approaches like plenoptic modeling [12], lightfield rendering [11] and the lumigraph [7] have lately received a lot of attention, since they can capture the appearance of a 3D scene from images only, without the explicit use of 3D geometry. Thus one may be able to capture objects with very complex geometry that can not be modeled otherwise. Basically one caches all possible views of the scene and retrieves them during view rendering.

Geometric 3D modeling approaches generate explicit 3D scene geometry and capture scene details mostly on polygonal (triangular) surface meshes. A limited set of camera views of the scene is sufficient to reconstruct the 3D scene. Texture mapping adds the necessary fidelity for photo-realistic rendering to the object surface. Recently progress has been reported on calibrating and reconstructing scenes from general hand-held camera sequences with a *Structure from Motion* approach [6, 13].

Somewhere in between both approaches is view-dependent texture mapping. Here an approximate geometrical model is combined with a set of view-dependent texture maps that are selected during rendering [3].

The problem common to all approaches is the need to calibrate the camera sequence. Typically one uses calibrated camera rigs mounted on a special acquisition device like a robot [11], or a dedicated calibration pattern is used to facilitate calibration [7].

In the case of lightfield generation from a hand-held camera, one typically generates very many (hundreds) of images, but with a specific distribution of the camera viewpoints. Since we want to capture the appearance of the object from all sides, we will try to sample the viewing sphere, thus generating a *mesh of view points*. To fully exploit hand-held sequences, we will also have to deviate from the restricted lightfield data structure and adopt a more flexible rendering data structure based on the viewpoint mesh.

In this contribution we tackle the problem of camera calibration from very many images under special consideration of dense viewsphere sampling. The necessary camera calibration and local depth estimates are obtained with a structure from motion approach. We will first give a brief overview of existing image-based rendering and geometric reconstruction techniques. We will then focus on the calibration problem for plenoptic sequences. Finally we will describe the image-based rendering approach that is adapted to our calibration. Experiments on calibration, geometric approximation and image-based rendering conclude this contribution.

2 Previous work

Plenoptic modeling describes the appearance of a scene through all light rays (2-D) that are emitted from every 3-D scene point, generating a 5D-radiance function [12]. Recently two equivalent realizations of the plenoptic function were proposed in form of the lightfield [11], and the lumigraph [7]. They handle the case when we observe an object surface in free space, hence the plenoptic function is reduced to four dimensions (light rays are emitted from the 2-dimensional surface in all possible directions).

2.1 Lightfield data representation

The original 4-D lightfield data structure employs a two-plane parameterization. Each light ray passes through two parallel planes with plane coordinates (s, t) and (u, v). Thus the ray is uniquely described by the 4-tuple (u, v, s, t). The (s, t)-plane is the *viewpoint plane* in which all camera focal points are placed on regular gridpoints. The (u, v)-plane is the *focal plane* where all camera image planes are placed with regular pixel spacing. The optical axes of all cameras are perpendicular to the planes. This data structure covers one side of an object. For a full lightfield we would need to construct six such data structures on a cube around the object.

New views can be rendered from this data structure by placing a virtual camera on an arbitrary view point and intersecting the viewing ray r with the two planes at (s, t, u, v). The resulting radiance is a simple radiance lookup for r. This, however, applies only if the viewing ray passes through original camera view points and pixel positions. For rays passing in between the (s, t) and (u, v) grid coordinates an interpolation is applied that will degrade the rendering quality depending on the scene geometry. In fact, the lightfield contains an implicit geometrical assumption: The scene geometry is planar and coincides with the focal plane. Deviation of the scene geometry from the focal plane causes image warping.

2.2 The Lumigraph

The discussion above reveals two major problems when acquiring lightfields from real image sequences. First, the need to calibrate the camera poses in order to construct the viewpoint plane, and second the estimation of local depth maps for view interpolation.

The original lumigraph approach [7] already tackles both problems. A calibration of the camera is obtained by incorporating a background with a known calibration pattern into the scene. The known specific markers on the background are used to obtain camera parameter and pose estimation [17]. They provide no means to calibrate the images from image data only. For depth integration the object geometry is approximated by constructing a visual hull from the object silhouettes. The hull approximates the global surface geometry but can not deal with local concavities. Furthermore, the silhouette approach is not feasible for general scenes and viewing conditions since a specific background is needed. This approach is therefore confined to laboratory conditions and does not provide a general solution for arbitrary scenes. If we want to utilize the image-based approach for general viewing conditions we identify two main needs:

- the need to directly obtain camera calibration from the image data, and
- the need to estimate local depth for view interpolation.

2.3 The structure-from-motion approach to surface reconstruction

The problem of simultaneous camera calibration and depth estimation from image sequences has been addressed for quite some time in the computer vision community. In the uncalibrated case all parameters, camera pose and intrinsic calibration as well as the 3D scene structure have to be estimated from the 2D image sequence alone. Faugeras and Hartley first demonstrated how to obtain uncalibrated projective reconstructions from image sequences alone [4, 9]. Since then, researchers tried to find ways to upgrade these reconstructions to metric (i.e. Euclidean but unknown scale, see [5, 16]).

Recently a method was described to obtain metric reconstructions for fully uncalibrated sequences even for changing camera parameters with methods of self-calibration [13]. For dense structure recovery a stereo matching technique was applied between image pairs of the sequence to obtain a dense depth map for each viewpoint. From this depth map a triangular surface wire-frame is constructed and texture mapping from the image is applied to obtain realistic surface models [10]. The approach allows metric surface reconstruction in a 4-step approach:

1. projective calibration is obtained by robust tracking of salient feature points over the image sequence,
2. the metric structure of the scene and the cameras is reconstructed through camera self-calibration,
3. dense depth maps for all view points are computed from correspondences between adjacent image pairs of the sequence,
4. a 3-D surface mesh approximates the geometry, and surface texture is mapped onto it to enhance the visual appearance.

3 Calibration of viewpoint meshes

In this contribution we propose to extend the sequential structure-from-motion approach to the calibration of the viewpoint sphere. Plenoptic modeling amounts to a dense sampling of the viewing sphere that surrounds the object. One can interpret the different camera viewpoints as samples of a generalized surface which we will call the *viewpoint surface*. It can be approximated as a triangular viewpoint mesh with camera positions as nodes. In the specific case of lightfields this viewing surface is simply a plane and the sampling is the regular camera grid. If a programmable robot with a camera arm is at hand, one can easily program all desired views and record a calibrated image sequence. For sequences from a handheld videocamera however we obtain a general surface with possible complex geometry and non-uniform sampling. To generate the viewpoints with a simple video camera, one might want to sweep the camera around the object, thus creating a zig-zag scanning path on the viewing surface. The problem that arises here is that typically very long image sequences of several hundreds of views have to be processed. If we process the images strictly in sequential order as they come from the video stream, then images have to be tracked one by one. One can think of walking along the path of camera viewpoints given by the recording frame index. This may cause error accumulation in viewpoint tracking, because object features are typically seen only in a few images and will be lost after some frames due to occlusion and mismatching. It would therefore be highly desirable to detect the presence of a previously tracked but lost feature and to tie it to the new image.

The case of disappearing and reappearing features is very common in viewpoint surface scanning. Since we sweep the camera in a zigzag path over the viewpoint surface, we will generate rows and columns of an irregular mesh of viewpoints. Even if the viewpoints are far apart in the sequence frame index they may be geometrically close on the viewpoint surface. We should therefore exploit the proximity of camera viewpoints irrespectively of their frame index.

3.1 Viewpoint mesh weaving

In this section we will develop the multi-viewpoint tracking algorithm that allows to actually weave the viewpoint sequence into a connected viewpoint mesh.

Image pair matching. The basic tool for the viewpoint tracking is the two-view matcher. Image intensity features are detected with the Harris corner detector[8] and have to be matched between the two images I_j, I_k of the view points P_i, P_k. Here we rely on a robust computation of the Fundamental matrix F_{ik} with the RANSAC (RANdom SAmpling Consensus) method [15]. A minimum set of 7 features correspondences is picked from a large list of potential image matches to compute a specific F. For this particular F the support is computed from the other potential matches. This procedure is repeated randomly to obtain the most likely F_{ik} with best support in feature correspondence.

The next step after establishment of F is the computation of the 3×4 camera projection matrices P_i and P_k. The fundamental matrix alone does not suffice to fully compute the projection matrices. In a bootstrap step for the first two images we follow the approach by Beardsley *et al.* [1]. Since the camera calibration matrix K is unknown a priori we assume a approximate $\tilde{K}$ to start with. The first camera is then set to $P_0 = \tilde{K}[I|0]$ to coincide with the world coordinate system, and the second camera P_1 can be derived from the epipole e and F as

$$P_1 = \tilde{K}\left[[e]_x F + e a^T | r e\right] \text{ with } [e]_x = \begin{bmatrix} 0 & -e_3 & e_2 \\ e_3 & 0 & -e_1 \\ -e_2 & e_1 & 0 \end{bmatrix}$$

P_1 is defined up to a global scale r and the unknown plane $\pi_{\inf}$, encoded in a^T (see also [14]). Thus we can only obtain a projective reconstruction. The vector a^T should be chosen such that the left 3×3 matrix of P_i best approximates an orthonormal rotation matrix. The scale r is set such that the baseline length between the first two cameras is unity. K and a^T will be determined later during camera self-calibration.

Once we have obtained the projection matrices we can triangulate the corresponding image features to obtain the corresponding projective 3D object features. The object points are determined such that their reprojection error in the images is minimized. In addition we compute the point uncertainty covariance to keep track of measurement uncertainties. The 3D object points serve as the *memory* for consistent camera tracking, and it is desirable to track the projection of the 3D points through as many images as possible.

Sequential camera tracking. Each new view of the sequence is used to refine the initial reconstruction and to determine the camera viewpoint. Here we rely on the fact that two adjacent frames of the sequence are taken from nearby view points, hence many object features will be visible in both views. The procedure for adding a new frame is much like the bootstrap phase. Robust matching of $F_{i,i+1}$ between the current and the next frame of the sequence relates the 2D image features between views I_i and I_{i+1}. Since we have also the 2D/3D relationship between image and object features for view I_i, we can transfer the object features to view I_{i+1} as well. We can therefore think of the 3D features as self-induced calibration pattern and directly solve for the camera projection matrix from the known 2D/3D correspondence in view I_{i+1} with a robust (RANSAC) computation of P_{i+1}. In a last step we update the existing 3D structure by minimizing the resulting feature reprojection error in all images. A Kalman filter is applied for each 3D point and its position and covariance are updated accordingly. Unreliable features and outliers are removed, and newly found features are added.

3.2 Viewpoint mesh tracking

The sequential approach as described above yields good results for the tracking of short sequences. New features are added in each image and the existing features are tracked throughout the sequence. Due to scene occlusions and inevitable measurement outliers, however, the features may be lost or wrongly initialized, leading to erroneous estimates and ultimately tracking failure. So far several strategies have been developed to avoid this situation. Recently Fitzgibbon et al. [6] addressed this problem with a hierarchical matching scheme that matches pairs, triplets, short subsequences and finally full sequences. However, they track along the linear camera path only and do not consider the extended relationship in a *mesh of viewpoints.* By exploiting the topology of the camera viewpoint distribution on the viewpoint surface we can extend the sequential tracking to a simultaneous matching of neighboring viewpoints. The viewpoint mesh is described by the node geometry (camera viewpoints) and the topology (which viewpoints are connected as nearest neighbors). Initially both topology and geometry are unknown and have to be retrieved. Fortunately, a special topology (sequential connectivity) is given by the frame index of the camera recording. This will serve as bootstrap to the recovery of the viewpoint surface.

Look-ahead and backtracking. Our goal is to recover topology and geometry of the viewpoint surface. We start sequentially with Single-stepping through the camera sequence as described above. This procedure computes the geometry of the camera from the connectivity with the previous viewpoint. To establish the connectivity to all nearest viewpoints we have now two possibilities: Look-ahead and backtracking. For look-ahead one tries to compute image relationships between the current frame and all future frames. Effectively this amounts to a full search between all possible image pairs. A reduced forward search would try to find the first best fit only. Still it will produce a large computational overhead.

For backtracking the situation is more fortunate, since for previous cameras we have already computed the geometry. It is therefore easy to compute the geometrical distance between the current and all previous cameras and to find the nearest viewpoints. Of course one has to account for the non-uniform viewpoint distribution and to select only viewpoints that give additional information. We have adopted a scheme to divide the viewing surface into angular sectors around the current viewpoint and to select the nearest cameras that are most evenly distributed around the current position. The search strategy is visualized in fig. 1. The camera produces a path whose positions have been tracked up to viewpoint $i-1$ already, resulting in a mesh of viewpoints (filled dots). The new viewpoint i is estimated from those viewpoints that are inside the shaded part of the sphere. The cutout section avoids unnecessary evaluation of nearby

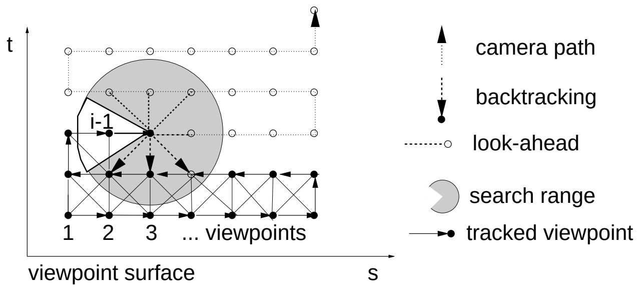

Figure 1: Search strategy to connect the current with previous viewpoints.

cameras $i-1, i-2, \ldots$. The cut-out section is always oriented along the connection between the viewpoints $i-1$ and i. The radius of the search sphere is adapted to the distance between the last two viewpoints.

Once we have found the local topology to the nearest view points we can update our current position by additional matching. In fact, each connecting edge of our viewpoint mesh allows the computation of F between the viewpoints. More important, since we are now matching with images way back in the sequence, we can *couple the 3D structure* much more effectively to image matches. A match that was lost during sequential tracking still lives in the previous images and will now be revived by the matching. Thus, a 3D feature lives typically much longer and is seen in more images than with simple sequential tracking. In addition to the revival of old features we obtain a much more stable estimate for the single viewpoint as well. Each image is now matched with (typically 3-4) images in different spatial directions with reduces the risk of critical or degenerate situations. The risk of feature loss due to occlusion is also minimized since a feature is checked in several images simultaneously.

4 Rendering from the viewpoint mesh

The previous section described how to acquire a calibrated viewpoint mesh. Now, virtual camera views are to be reconstructed from the set of calibrated views. The lumigraph approach [7] gives one solution to this problem. The regular grid structure of the lumigraph is synthesized from arbitrary views using approximated geometry which is reconstructed with a structure–from–silhouette technique. This so–called *rebinning* step also fills gaps by applying a multiresolution approach. Because of interpolating this regular structure from the original data, information is lost and blurring effects occur. The reconstruction of views is done by a look–up in this regular structure, considering depth corrections.

To prevent the disadvantages of the rebinning step, our goal is to render views from the originally recorded images directly. In the simplest way this is achieved by projecting all images onto a common plane of "mean geometry" by a 2D projective mapping. Having a full triangulation of the viewpoint surface, we project this mesh into the virtual camera. For each triangle of the mesh, only the views that span the triangle are contributing to the color values inside. Each triangle acts as a window through which the three corresponding mapped textures are seen in the virtual camera. The textures are overlapped by applying alpha blending with barycentric weights depending on the distance to the corresponding triangle corner. As the whole mapping procedure is a 2D projective mapping, it can be done in real time using the texture mapping and alpha blending facilities of graphics hardware.

4.1 Combining images and geometry

The rendering approach can be refined using more detailed geometric information. Depending on the virtual camera position, a plane of mean geometry can be assigned adaptively to each image triplet which forms a triangle. Adaptive to the size of each triangle and the complexity of geometry, further subdivision of each triangle may improve the accuracy of the reconstruction. For this use of geometry, local correspondence maps or depth maps are sufficient, so no consistent 3D–model needs to be created, which would require the registration of different views. Ultimately this approach will result in a system that can handle geometric as well as image-based representations simultaneously by exploiting viewpoint-adaptive depth and texture maps.

5 Experimental results on camera calibration

To evaluate our approach, we recorded a test sequence with known ground truth from a calibrated robot arm. The camera is mounted on the arm of a robot of type SCORBOT-ER VII. The position of its gripper arm is known from the angles of the 5 axes and the dimensions of the arm. Optical calibration methods were applied to determine the eye-hand calibration of the camera w.r.t. the gripper arm. We achieve a mean absolute positioning error of 4.22 mm or 0.17 degrees, respectively [2]. The repetition error of the robot is 0.2 mm and 0.03 degrees, respectively. Because of the limited size of the robot, we are restricted to scenes with maximal size of about 100 mm in diameter.

For the ground truth experiment the robot sampled a 8×8 spherical viewing grid with a radius of 230 mm. The viewing positions enclosed a maximum angle of 45 degrees which gives an extension of the spherical viewpoint surface patch of 180×180 mm^2. The scene consists of a cactus and some metallic parts on a piece

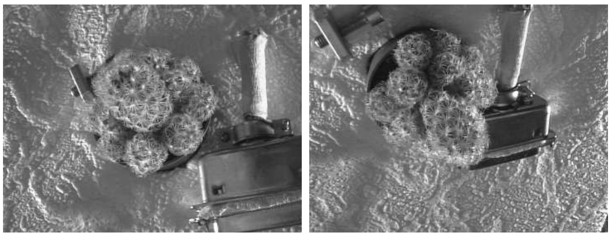

Figure 2: Image 1 and 64 of the 8×8 original camera images of the sphere sequence.

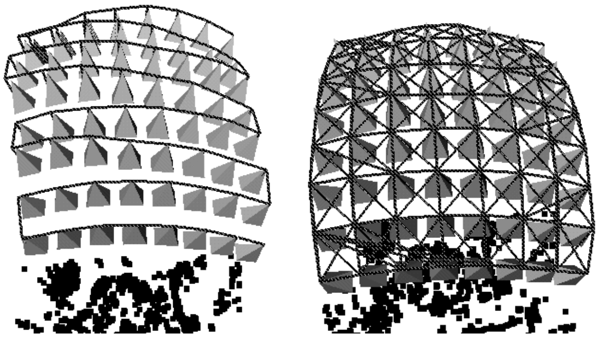

Figure 3: Left: Camera track and view points for sequential tracking. Right: Camera topology mesh and view points for viewpoint mesh weaving. The cameras are visualized as pyramids, the black dots display some of the tracked 3D points.

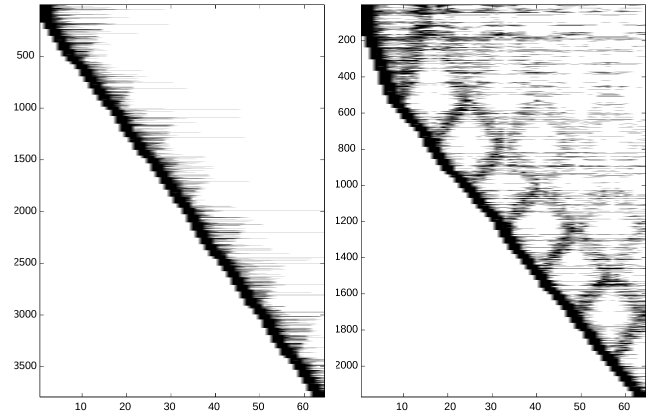

Figure 4: Distribution of tracked 3D points (vertical) over the images (horizontal). Left: Sequential tracking. Right: viewpoint mesh weaving. Please note the specific 2D pattern in the right graph that indicates how a tracked point is lost and found back throughout the sequence.

of rough white wallpaper. Two of the original images are shown in fig. 2. Please note the occlusions and the reflections and illumination changes in the images.

We compared the viewpoint mesh weaving algorithm with the sequential tracking and with ground truth data. Fig. 3 shows the camera path and connectivity for the sequential tracking (left) and viewpoint weaving (right). Weaving generates the topological network that tightly connects all neighboring views. On average each node was linked to 3.4 connections.

Table 1: Statistics for camera tracking over 64 images.

Algorithm:	sequential tracking	viewpoint weaving
# Pts	3791	2169
# Im/Pts(ave.)	4.8	9.1
# Im/Pts(max.)	28	48
# Pts/Im(ave.)	286	306
# Min Pts	1495	458

The graph in fig. 4 illustrates very clearly the survival of 3D points. A single point may be tracked throughout the sequence but is lost occasionally due to occlusion. However as the camera passes near to a previous position in the next sweep it is revived and hence tracked again. This results in fewer 3D points (# Pts) which are tracked in more images (# Im/Pts). Some statistics of the tracking are summarized in table 1. A minimum amount of 3 images is required before a feature is kept as 3D point. For viewpoint weaving, 3D points are usually tracked in the double amount of images as compared to sequential tracking, and the average number of image matches (#Pts/Im) is increased. Important is also that the number of points that are tracked in 3 images only (# Min Pts) drops sharply. These points are usually unreliable and should be discarded.

A quantitative evaluation of the tracking was performed by comparing the estimated metric camera pose with the known Euclidean robot positions. We anticipate two types of errors: 1) a stochastic measurement noise on the camera position, and 2) a systematic error due to a remaining projective skew from imperfect self-calibration. For comparison we transform the measured metric camera positions into the Euclidean robot coordinate frame. With a projective transformation we can eliminate the skew and estimate the measurement error. We estimated the projective transform from the 64 corresponding camera positions and computed the residual distance error. The distance error was normalized to relative depth by the mean surface distance of 250 mm. The mean residual error dropped from 1.1% for sequential tracking to 0.58% for viewpoint weaving (see table2). The position repeatability error of the robot itself is 0.08%.

If we assume that no projective skew is present then a similarity transform will suffice to map the coordi-

Table 2: Ground truth comparison of 3D camera positional error between the 64 estimated and the known robot positions [in % of the mean object distance of 250 mm].

Camera position	projective		similarity	
Tracking Error[%]	mean	dev	mean	dev
sequential	1.08	0.69	2.31	1.08
2D viewpoints	0.57	0.37	1.41	0.61

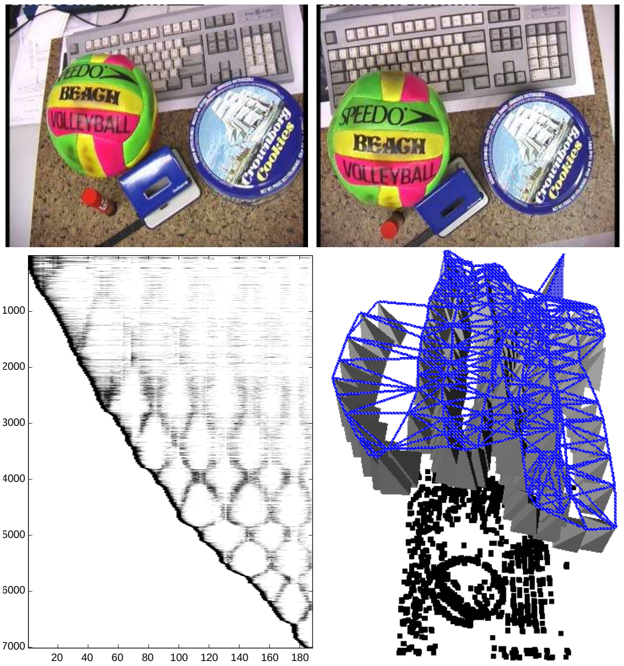

Figure 5: Top: Two images from hand-held office sequence. Bottom left: Distribution of 3D feature points (7014 points, vertical) over the image sequence (187 images, horizontal). Bottom right: Viewpoint mesh (in blue) with cameras as pyramids and 3D points (black).

nate sets onto each other. A systematic skew however will increase the residual error. To test for skew we computed the similarity transform from the corresponding data sets and evaluated the residual error. Here the mean error increased with a factor of about 2.4 to 1.4% which still is very good for pose and structure estimation from fully uncalibrated sequences.

5.1 Hand-held office sequence

We tested our approach with an uncalibrated hand-held sequence. A digital consumer video camera (Sony DCR-TRV900 with progressive scan) was swept freely over a cluttered scene on a desk, covering a viewing surface of about 1 m^2. The resulting video stream was then digitized on an SGI O2 by simply grabbing 187 frames at more or less constant intervals. No care was taken to manually stabilize the camera sweep. Fig. 5(top) displays two images of the sequence. The camera viewpoints are tracked and the viewpoint mesh topology is constructed with the viewpoint mesh weaving. Fig. 5(bottom) shows the statistics of the tracked 3D feature points (left) and the resulting camera viewpoint mesh with 3D points (right). The point distribution (left) shows the characteristic weaving structure when points are lost and found back throughout the sequence. The camera tracking illustrates nicely the zigzag scan of the hand movement as the camera scanned the scene. The viewpoint mesh is irregular due to the arbitrary hand movements. On the bottom half one can see the the reconstructed 3D scene points.

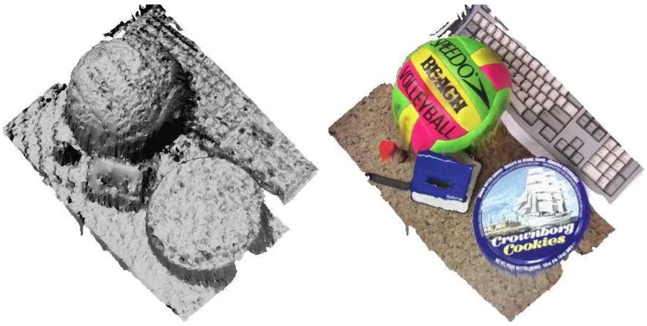

Figure 6: 3D surface model of office scene rendered with shading (left) and surface texture (right).

The statistical evaluation gives an impressive account on the tracking abilities. The camera was tracked over 187 images with at average 452 matches/image. A total of 7014 points were generated and matched on the average in 12 images each. A single 3D point was even tracked over 181 images, with image matches in 95 images.

5.1.1 Scene reconstruction and viewpoint mesh rendering

From the calibrated sequence we can compute any geometric or image based scene representation. As an example we show in fig. 6 a geometric surface model of the scene with approximate local scene geometry that was generated by dense surface matching. Even fine details like the keyboard keys are modeled.

Some results of the proposed image-based render-

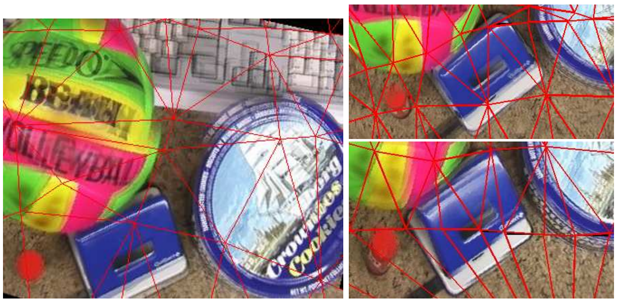

Figure 7: Left: novel scene view rendered far away from the viewpoint mesh. The red lines indicate the projection of the viewpoint mesh into the novel view. Right: Two closeup views from different viewing directions. Please note the changing surface reflection on the object surface.

ing from the viewpoint mesh are shown in Fig. 7. These views were rendered without local geometry. Only a mean plane was fitted through the scene which causes interpolation shadowing artifacts. In the closeup views (right) a detail was viewed from different directions. The changing surface reflections are rendered correctly due to the view-dependent imaging. This shows the potential of the method. We expect to achieve very realistic scene reconstructions in combining the view-dependent rendering with the local geometry estimates.

6 Further Work and Conclusions

We have proposed a camera calibration algorithm for geometric and plenoptic modeling from uncalibrated hand-held image sequences. During image acquisition the camera is swept over the scene to sample the viewing sphere around an object. The new algorithm considers the two-dimensional topology of the viewpoints and weaves a viewpoint mesh with high accuracy and robustness. It significantly improves the existing sequential structure-from-motion approach and allows to fully calibrate hand-held camera sequences that are targeted towards plenoptic modeling. The calibrated viewpoint mesh was used for the reconstruction of geometric surface models and for image-based rendering, which even allows to render reflecting surfaces.

Currently we are concentrating to fully integrate calibration, geometrical reconstruction, and image-based rendering. The calibration delivers the viewpoint mesh as basic data structure, which can be interpreted as a generalized viewpoint plane in a lightfield data structure. The reconstructed surface geometry will likewise generalize the lightfield focal plane. We are currently able to render novel views from the viewpoint mesh, but so far no local geometry has been used to improve the view interpolation. The synergy of camera sequence tracking, local geometric interpolation and image-based rendering will allow very realistig scene representations from hand-held camera sequences.

References

[1] P. Beardsley, P. Torr and A. Zisserman: 3D Model Acquisition from Extended Image Sequences. *ECCV 96*, LNCS 1064, vol.2, pp.683-695.Springer 1996.

[2] R. Beß: Kalibrierung einer beweglichen, monokularen Kamera zur Tiefengewinnung aus Bildfolgen. In: Kropatsch, W. G. and Bischof, H. (eds.), Informatics Vol. 5, Mustererkennung 1994, 524 – 531, Springer Verlag Berlin, 1994.

[3] P. Debevec, Y. Yu, G. Borshukov: Efficient View-Dependent Image-Based Rendering with Projective Texture Mapping. Proceedings SIGGRAPH '98, ACM Press, New York, 1998.

[4] O. Faugeras: What can be seen in three dimensions with an uncalibrated stereo rig. *Proc. ECCV'92*, pp.563-578.

[5] O. Faugeras, Q.-T. Luong and S. Maybank: Camera self-calibration - Theory and experiments. *Proc. ECCV'92*, pp.321-334.

[6] A. Fitzgibbon and A. Zisserman: Automatic Camera Recovery for Closed or Open Image Sequences. Proceedings ECCV'98. LNCS Vol. 1406, Springer, 1998.

[7] S. Gortler, R. Grzeszczuk, R. Szeliski, M. F. Cohen: The Lumigraph. Proceedings SIGGRAPH '96, pp 43–54, ACM Press, New York, 1996.

[8] C.G. Harris and J.M. Pike: 3D Positional Integration from Image Sequences. 3rd Alvey Vision Conf, pp. 233-236, 1987.

[9] R. Hartley: Estimation of relative camera positions for uncalibrated cameras. *ECCV'92*, pp.579-587.

[10] R. Koch, M. Pollefeys, and L. Van Gool: Multi Viewpoint Stereo from Uncalibrated Video Sequences. *Proc. ECCV'98*, Freiburg, June 1998.

[11] M. Levoy, P. Hanrahan: Lightfield Rendering. Proceedings SIGGRAPH '96, pp 31–42, ACM Press, New York, 1996.

[12] L. McMillan and G. Bishop, "Plenoptic modeling: An image-based rendering system", *Proc. SIGGRAPH'95*, pp. 39-46, 1995.

[13] M. Pollefeys, R. Koch and L. Van Gool: Self-Calibration and Metric Reconstruction in spite of Varying and Unknown Internal Camera Parameters. Proc. ICCV'98, Bombay, India, Jan. 1998.

[14] M. Pollefeys, R. Koch, M. Vergauwen and L. Van Gool: Metric 3D Surface Reconstruction from Uncalibrated Image Sequences. In: 3D Structure from Multiple Images of Large Scale Environments. LNCS Series Vol. 1506, pp. 139-154. Springer-Verlag, 1998.

[15] P.H.S. Torr: Motion Segmentation and Outlier Detection. PhD thesis, University of Oxford, UK, 1995.

[16] B. Triggs: The Absolute Quadric. *Proc. CVPR'97.*

[17] R.Y.Tsai: A Versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology using off-the-shelf Cameras and Lenses. IEEE Journal Robotics and Automation RA-3,4 (Aug. 1987), 323-344.