Enhanced Video Mosaicing using Camera Motion Properties

Pulkit Parikh, C.V. Jawahar
Center for Visual Information Technology (CVIT)
International Institute of Information Technology (IIIT-H)
Gachibowli, Hyderabad - 500032, India
pulkitparikh@yahoo.com, jawahar@iiit.ac.in

Abstract

We propose a video mosaicing scheme which exploits the motion information, implicitly available in the video. The information about the camera motion is propagated to the homographies used for mosaicing. While some of the recent approaches make use of the information stemming from non-overlapping pairs of frames, the smoothness of the camera motion has gone largely under-capitalized. We present a technique which exploits this useful cue for refining homographies. Moreover, a generic framework which exploits the camera motion model, to relate homographies in a video, is also proposed. The analysis and results of the proposed algorithms demonstrate significant promise, in terms of accuracy and robustness.

1. Introduction

Mosaicing is an important step in many computer vision applications such as video stabilization, data compression, visualization of virtual environments, panoramic photography, etc. The conventional approach to video mosaicing extends the image mosaicing procedure to video by registering and stitching successive frames [5, 13, 14]. However, a video contains much more information than a set of isolated images, and little effort has gone into utilizing this information to provide better-quality mosaics. This work reveals how the redundant motion and texture information present in the video can be effectively used in building better quality mosaics.

Recently, graph-based global alignment approaches [6, 7, 10, 12] have become pupular wherein all overlapping frames, not just consecutive ones, are used for estimating homographies for mosaicing. First, homographies are computed for all pairs of images with sufficient overlap and a graph-based representation is built from it. In the second step, the problem of global registration is cast as the identification of an optimal structure (e.g., Minimum Spanning Tree) in the graph. The desired homographies are uniquely characterized by this graph structure (e.g., MST). The graph-based approaches are well-suited for bidirectional camera motions where the camera comes back to capture an object it has seen before. However, for unidirectional motions, it more or less degenerates to the conventional register-only-successive-video-frames approach. Another class of approches [2, 7] employs bundle adjustment wherein features from all frames are combined to optimize all the homography parameters at once. It helps overcome the phenomenon of error propagation i.e., the errors in registering the initial pairs of frames being carried forward to all the subsequent ones. Typically, the closed form estimate of the homography parameters is given as the initial estimate for bundle adjustment.

In spite of the above two approaches, there still remains scope for higher robustness and better accuracy in video mosaicing, leading to the exploration of a different cue which is the properties of the camera motion. It is observed that the trajectory of the camera, that captures the video to be mosaiced, is often smooth. Our approach is formulated to exploit this important piece of information for improved video mosaicing. In this paper, a simple but effective technique is proposed, which utilizes the decomposition of homography into *pose* $[R|T]$ and extrapolation in the pose space to refine the desired homography estimates.

In [1], homography is represented by three rotation angles and focal length to recognize images that are a part of the panorama and then, to generate the panorama. It is applicable for panoramic mosaicing where the camera rotates about its optical center, enabling them to use a 4 parameter homography representation. Our work is aimed at planar mosaicing and accounts for general camera motions, including translation.

Often, the camera motion that we encounter is not only smooth, but also follows a specific known motion model. If we have this *apriori* knowledge, the ensemble of homographies required for video mosaicing can be shown to be related by a well-defined model. This paper describes a generic approach that exploits this relationship for robust estimation of the homographies and thereby obtaining visually pleasing mosaics. Such algorithms are especially applicable when imaging is done by machine-controlled cameras, as in the case of robots and autonomous vehicles. Fig. 1(b) is the result of such an algorithm on a desert area wherein our approach can be observed to improve over the mosaic built using the conventional techniques.

2. Space of Homographies for Video Mosaicing

Almost all the mosaicing techniques [2, 5, 7, 10, 13, 14] in the literature consider the homographies between different pairs of images to be independent of each other. However, in a video, the homographies between different pairs of frames may often be tightly related, depending on the camera motion. The seminal work analyzing the relationship between homographies was done by Shashua and Avidan [11]. They considered a scenario wherein all possible planes are viewed from two fixed cameras, leading to a family of homographies. These homographies are shown to be constrained as given below.

Observation 1 *The space of all homographies (induced by any plane) between two fixed views is embedded in a four dimensional linear subspace.*

This implies that given any four *base* homographies (corresponding to four *base* planes), any new homography (corresponding to a new plane) can be expressed as their linear combination. This result, in the present from, is not very relevant for the mosaicing problem. In mosaicing, the problem on hand is the dual of the one described above. The plane is fixed while the camera views are changing. Another relevant work was done in [15] wherein a constraint was derived on relative homographies for a pair of planes, over multiple camera views. Each of these relative homographies corresponds to a fixed view and maps the image of one plane (captured from that fixed view) onto the other (captured from the same view).

Observation 2 *The collection of all relative homographies of a pair of planes (homologies) across multiple views, spans a 4-dimensional linear subspace.*

While Observation 2 provides useful insight, what we typically need for mosaicing is a contraint relating homographies mapping images of a single plane, captured from multiple views. This was done in [9] wherein the authors derived relationships between *incremental* (frame-to-next-frame) and reference (frame-to-reference) homographies, given certain camera motion models. Results in [9] can be generalized to arrive at Observation 3.

Observation 3 *The homographies induced by a fixed plane, between pairs of video-frames are related by a fixed set of parameters, depending on the camera motion model.*

This observation implies that a large number of homographies can be computed from a fixed, small number of parameters. Therefore, rather than using pair-wise information to compute the homographies individually, the information from all the frames can be pooled to estimate the parameters of the global homography model (i.e., model-fitting). The desired homographies can then be directly computed from these parameters. Note that estimation of a pair-wise homography from point correspondence can also be viewed as model-fitting, wherein an 8-parameter model (i.e. homography) is fitted to a subset of matching points. However, the ratio of the number of samples available to the number of parameters to be estimated, is much lower in this case.

3. Mosaicing in Presence of Continuous Camera Motion

There are certain real-life situations when the camera globally does not follow a specific motion model and the homographies can not be parameterized. In such cases, the smoothness of the camera trajectory can serve as a useful clue in improving the mosaicing process, especially if the camera motion is continuous.

3.1. Mosaicing using Smoothness of the Camera Trajectory

We detect and replace outlier homographies to improve the mosaicing accuracy. The most commonly encountered cause for the outlier homographies is poor quality (blurred, on most occasions) frames. It should be noted at the outset that in our approach, the emphasis is laid on replacing inaccurate homographies, and *not* on detecting and removing blurred frames (e.g., [8]). The substitute for an outlier homography is computed using the information encapsulated in the neighboring homographies. This involves the decomposition of each incremental homography to obtain the camera pose. The motivation for this decomposition is the fact that smoothness in the pose space does not translate into smoothness in the homography space due to the nonlinearity in the pose-to-homography relationship. Therefore, in order to exploit smoothness to refine an outlier homography, we move onto the pose space via homography decomposition, extrapolate in the pose space and finally, reconstruct the desired homography.

The relationship between homography and pose is the following [4]:

$$H = K\big(\tilde{R} - \frac{\tilde{T}\tilde{n}^T}{\tilde{d}}\big)K^{-1} \tag{1}$$

where, $[\tilde{R}|\tilde{T}]$ is the relative pose of the second camera assuming that the pose of the first camera is $[I|0]$. The vector $\tilde{n}$ is the normal of the plane of interest and $\tilde{d}$ is the perpendicular distance to the plane. Note that $\tilde{R}$, $\tilde{T}$, $\tilde{n}$ and $\tilde{d}$ are *relative* to the reference camera pose. For video mosaicing, we want to express the current incremental homography $H_{i,i+1}$ relating frame I_i to frame I_{i+1}, in terms of the *current* pose $[R_i|T_i]$, the *next* pose $[R_{i+1}|T_{i+1}]$, n and d, all defined in the (fixed) world coordinate system. Using the basics of rigid transformation, we express

$$H_{i,i+1} = K\left[R_{i+1}R_i^{-1} - (T_{i+1} - R_{i+1}R_i^{-1}T_i)\frac{n^T R_i^{-1}}{\tilde{d}}\right]K^{-1} \tag{2}$$

such that $\tilde{R}_i = R_{i+1}R_i^{-1}$, $\tilde{T}_i = T_{i+1} - R_{i+1}R_i^{-1}T_i$, $\tilde{n}^T = n^T R_i^{-1}$, $\tilde{d} = d - n^T R_i^{-1}T_i$ and K is the internal calibration matrix.

Decomposition of the homography $H_{i,i+1}$ is done as in [3] wherein Singular Value Decomposition (SVD) is used to compute $\tilde{R}_i$, $\frac{\tilde{T}_i}{\tilde{d}_i}$ and $\tilde{n}_i$. Now, the next pose can be computed given the current pose, using the expressions for $\tilde{R}_i$, $\tilde{T}_i$, $\tilde{d}_i$ and $\tilde{n}_i$, given in Equation 2, along with d. Assuming the first camera pose to be $[I|0]$ allows us to reconstruct the complete camera motion, from only homographies, which are computed from image measurements. Note that no 3D information is assumed to be known other than an approximate estimate of d.

Since the camera motion is smooth, the rotation angles and camera position vary smoothly for successive pairs of frames. The rotation angles are obtained by decomposing R_is. The absolute position of the camera P_i is given by $-R_i^{-1}T_i$. Now, for each successive pair of frames, we use the rotation angles and P_i, along with the reprojection error, to determine if the homography for the pair is an outlier. The cumulative average reprojection error is defined as follows.

$$E = \frac{1}{m*(N-1)}\sum_{i,j}\|\Pi(X_{i+1}^j) - \Pi(H_{i,i+1}X_i^j)\|^2$$

where, $i \in \{0, 1, \ldots, N-1\}$ denotes the frame number and $j \in \{0, 1, \ldots, M-1\}$ is the feature point index and Π is the imaging function. If the homography is not found to be an outlier, we decompose it and retain the decomposed parameters to compute the next pose if required. If it is an outlier, we individually extrapolate in the space of rotation angles and in the position space to obtain a locally smooth estimate of R_{i+1} and P_{i+1}. Then T_{i+1} can be directly computed from them. These, along with the previously computed n, are substituted in Equation 2 to compute a reasonable estimate for the outlier homography. The whole procedure is summarized in Algorithm 1.

Algorithm 1 Compute homographies enforcing smoothness of the camera motion

```
for frames $i$ = 0 to $N-2$ do
    Compute $H_{i,i+1}$ from feature-correspondences (or other methods).
    $[\tilde{R}_i, \frac{\tilde{T}_i}{\tilde{d}_i}, \tilde{n}_i] \leftarrow$ Decompose($H_{i,i+1}$).
    Compute $R_{i+1}$ and $T_{i+1}$ from the above (decomposed) parameters and $[R_i\ T_i]$ i.e. the pose computed in the previous iteration.
    (Initial pose is $[I\ \mathbf{0}]$)
    Compute Position $P_{i+1}$ and Rotation Angles from $R_{i+1}$ and $T_{i+1}$.
    Check reliability of $H_{i,i+1}$ using the cumulative average projection error and displacement in the camera position.
    if $H_{i,i+1}$ is an outlier homography then
        Extrapolate in the space of $P$ and Rotation Angles to obtain reliable estimates of $R_{i+1}$ and $T_{i+1}$.
        Compute the new estimate of $H_{i,i+1}$ using $R_{i+1}$, $T_{i+1}$ and calibration matrix $K$.
    end if
end for
```

3.2. Mosaicing using a Motion Model

Our second method does not need the camera calibration. It is suitable for situations when the camera motion is controlled. To facilitate better understanding of the model-based approach, we begin with a simple model - the linear, translational camera motion model. We know from [9] that under linear translational motion, all incremental (frame-to-next frame) homographies are related by an eleven parameter model:

$$H_{i,i+1} = I + \frac{C}{c_1 + i.c_2} \tag{3}$$

where, C is a 3×3 matrix and c_1 and c_2 are scalars. This directly leads to

$$X_{i+1}^j = \left[I + \frac{C}{c_1 + i.c_2}\right]X_i^j$$

where, X denotes a feature point, i denotes the frame number and j is the feature point index in the given frame.

It can be seen from the above equation that each corresponding pair of points gives two linearly independent equations in the model parameters C and c_1, c_2. Thus, what we have is an over-determined linear system of equations

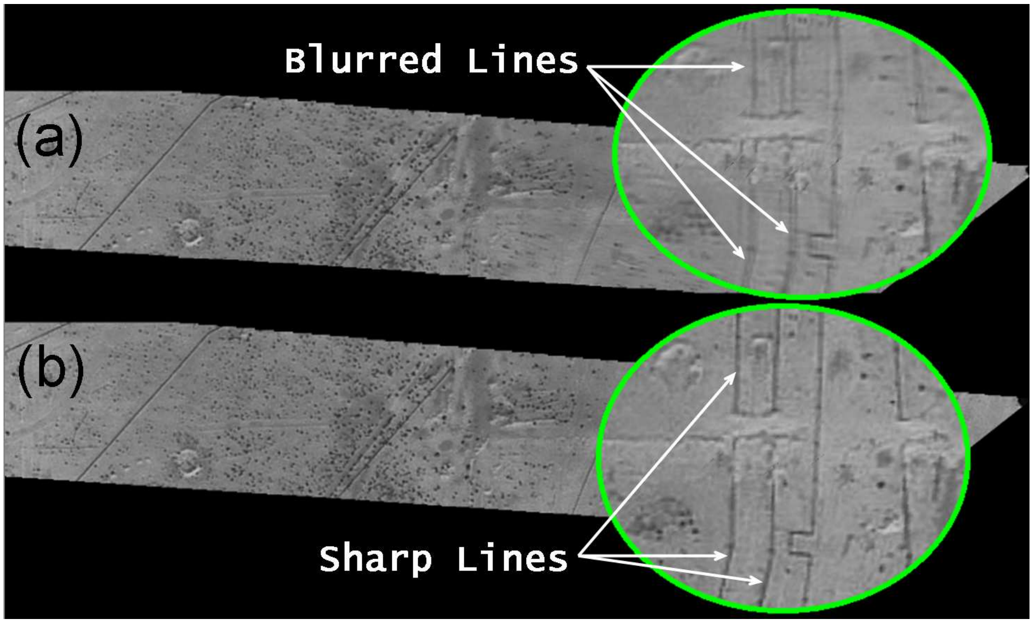

Figure 1. Mosaic of an Aerial Video over a Desert. (a) Bundle Adjustment, (b) Proposed Model-based Approach. Our algorithm is able to produce good quality mosaics even if features to track are rare.

in the model parameters and we can use SVD to solve it. Mostly, the solution given by this linear technique is satisfactory. If not, it may be used as the initial guess to a non-linear, iterative algorithm like the Levenberg and Marquardt algorithm. This simple technique is generalized to cater to any arbitrarily complex homography model. The steps of this generic algorithm are given in Algorithm 2.

Algorithm 2 Build mosaic using a camera motion model

```
1: Establish correspondences between all pairs of frames.
2: Estimate the homography model parameters $M_p$ such that
     $$E = \sum_{i,j} \|(\Pi(X_i) - \Pi(H(M_p, i, j)X_j))\|^2$$
   is minimized. (Use a non-linear optimization technique such as the Levenberg-Marquardt algorithm)
3: Identify samples, for which reprojection error is greater than twice the average reprojection error, as the outliers.
4: Re-estimate the model parameters $M_p$ from the inliers alone.
5: Compute the reference homographies from the model and build the mosaic.
```

In Algorithm 2, first, the global model (M_p) is estimated from the measurements from the individual frames and the homography is computed directly from. i.e., $H_{i,j} = f(i, j, M_p)$, where M_p is the set of parameters of the global homography model. Note that the noisy image measurements will have far lesser influence on the estimation of $f(\cdot)$, compared its influence on the individual homography estimates. This helps us in providing better mosaics, by not worrying about the failure in correspondence in a specific frame pair.

4. Results and Discussions

Now, we analyze the performance of the proposed mosaicing schemes in various situations. Experiments done in this section are carefully designed on synthetic data, to study the quantitative performance in a systematic manner. First, we consider the problem of mosaicing a video captured by a camera undergoing a smooth motion. For the purpose of mosaicing, frame to frame homographies were computed between every pair of successive frames. Some of these homography estimates were poor. This can be seen by large errors in the reprojection values at certain frames in Fig. 2(a). Camera trajectory is computed from the estimated homographies by their decomposition. As expected, the homography errors propagate into this procedure. In Fig. 2(b), one can see the the homography errors reflected in the camera trajectory. Notice that this camera trajectory is not smooth in many places. After applying our approach of extrapolation in the pose space, the camera trajectory be-

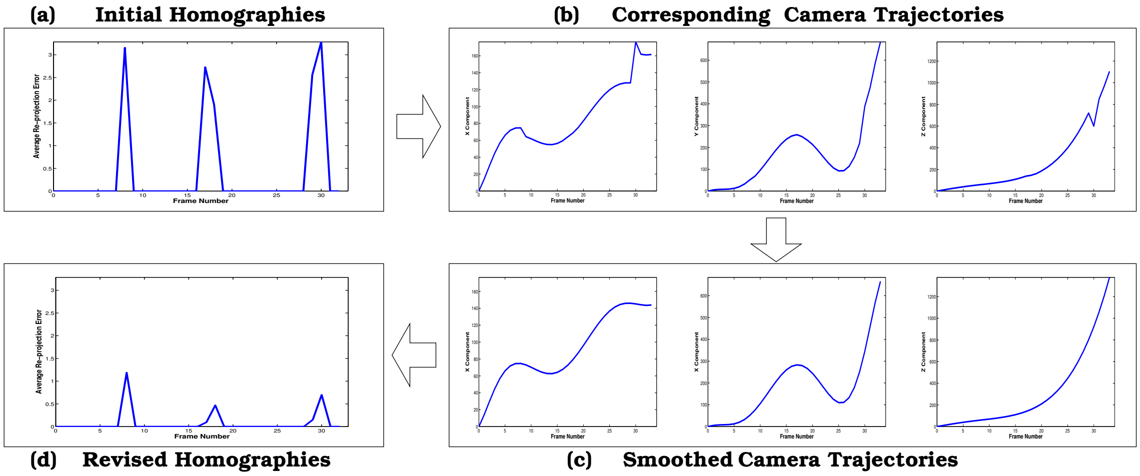

Figure 2. Experiments on a video of 34 frames. Noisy homographies were computed at frames 8-9,17-18,18-19,29-30 and 30-31. (a) and (d) show the reprojection errors for incremental homographies, before and after applying Algorithm 1, respectively. (b) and (c) show the camera trajectories in x, y and z directions, from top to bottom. The originally computed trajectory ((b)) is smoothened and shown in (c). Then, the homographies are refined using (c), as indicated by the low errors in (d).

comes smooth (Fig. 2(c)). We use these reconstructed poses to compute substitutes for the outlier homographies, and then, perform the mosaicing. The reduction in the reprojection errors (Fig. 2(d)), after applying our extrapolation-based algorithm, may be observed.

We analyzed the sensitivity of our model-based approach to noise in the image measurements. Uniform noise was added to the set of point correspondences. To compare the homographies computed using our approach and the conventional approach, wherein homographies are individually computed using RANSAC, we compute the cumulative average re-projection error for the noise-free points. Fig. 4 shows that for our approach, the error increases at a much slower rate with the level of noise, compared to the conventional approach.

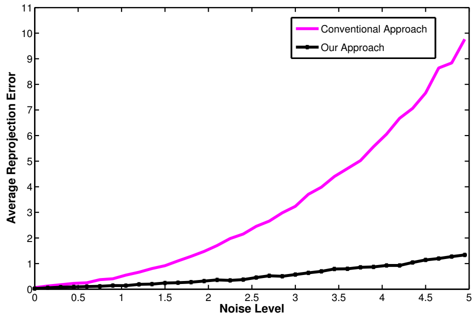

Figure 4. Cumulative Average Reprojection Error Vs Noise. The error for Algorithm 2 is always lower than that for the conventional approach.

In Fig. 1, we present the mosaic of an aerial video of a desert area. There is little texture information in the input frames. Thus, as expected, the feature extractor was not able to find enough (accurate) points in the frames. Therefore, the conventional technique, using RANSAC on successive pairs of frames, could not perform accurate registration. Applying bundle adjustment on top of it, using an objective function similar to the one used in [7, 12], reduced the reprojection error marginally but made no visible difference to the appearence of the mosaic. However, our model-based approach performed very well since it combines information from all over-lapping frame pairs for computing each homography. A part of the mosaic is zoomed and shown. Note that the graph-based approach is also likely to fail here because of the uni-directional motion, as mentioned in Section 1.

Fig. 3 demonstrates how our approach tackles the frequently encountered repetitive texture problem. The conventional RANSAC-based technique, even after bundle adjustment, failed to compute accurate homographies for many frames of this video because of several false matches in the feature matching step. However, since our approach accumulates correspondence from many frames, it was less affected by the mis-matched points and was able to yield

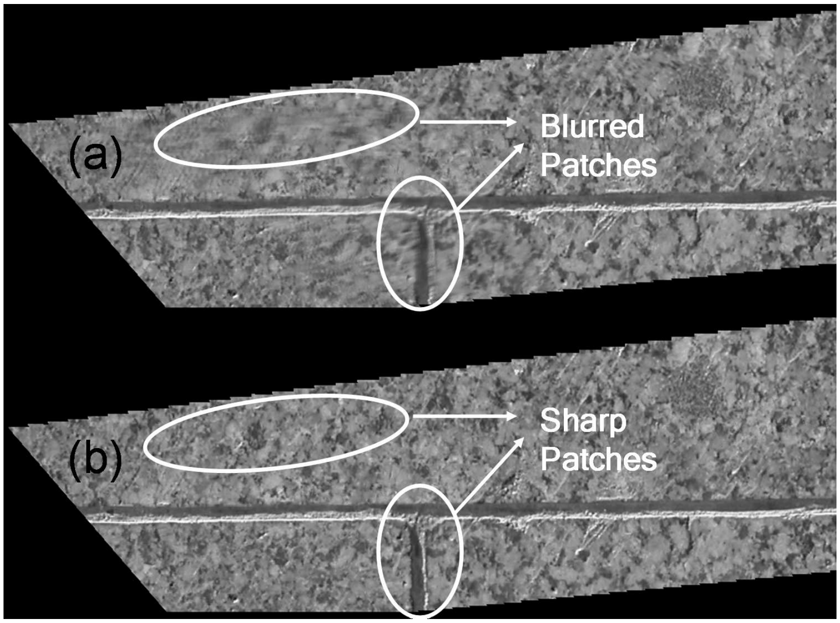

Figure 3. Mosaicing in presence of Repetitive Texture. (a) Bundle Adjustment, (b) Our Approach. (b) shows that our approach stays unaffected under false feature matching.

quite accurate homographies and consequently, a better mosaic.

5. Conclusions

We have presented video mosaicing algorithms with applicability to a variety of real-life scenarios, including uncalibrated and calibrated camera as well as parameterized and unparameterized motion models. Experiments were carried out keeping in mind the frequently occurring mosaicing problems and results show the robustness of our algorithms.

References

[1] M. Brown and D. G. Lowe. Recognising panoramas. In *ICCV*, pages 12–18, Washington, DC, USA, 2003. IEEE Computer Society.

[2] D. Capel and A. Zisserman. Automatic mosaicing with super-resolution zoom. In *CVPR*, pages 885–891, 1998.

[3] O. Faugeras and F. Lustman. Motion and structure from motion in a piecewise planar environment. *IJPRAI*, 2:485–508, 1988.

[4] R. Hartley and A. Zisserman. *Multiple view geometry in computer vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2000.

[5] M. Irani, P. Anandan, and S. Hsu. Mosaic based representations of video sequences and their applications. In *ICCV*, pages 605–611, 1995.

[6] E.-Y. Kang, I. Cohen, and G. G. Medioni. A graph-based global registration for 2d mosaics. In *ICPR*, pages 1257–1260, 2000.

[7] R. Marzotto, A. Fusiello, and V. Murino. High resolution video mosaicing with global alignment. In *CVPR (1)*, pages 692–698, 2004.

[8] D. Nister. Frame decimation for structure and motion. In *SMILE00: In Proc. 2nd Workshop on Structure from Multiple Images of Large Environments*, pages 17–34, 2000.

[9] P. Parikh and C. V. Jawahar. Motion constraints for video mosaicing. In *IEE International Conference on VIE*, pages 494–499, 2006.

[10] H. S. Sawhney, S. C. Hsu, and R. Kumar. Robust video mosaicing through topology inference and local to global alignment. In *ECCV (2)*, pages 103–119, 1998.

[11] A. Shashua and S. Avidan. The rank 4 constraint in multiple view geometry. In *ECCV '96: Proceedings of the 4th European Conference on Computer Vision-Volume II*, pages 196–206, London, UK, 1996. Springer-Verlag.

[12] H.-Y. Shum and R. Szeliski. Construction of panoramic mosaics with global and local alignment. *Int. J. Comput. Vision*, 36(2):101–130, 2000.

[13] D. Steedly, C. Pal, and R. Szeliski. Efficiently registering video into panoramic mosaics. In *ICCV*, pages 1300–1307, 2005.

[14] R. Szeliski. Video mosaics for virtual environments. *IEEE CG&A*, pages 22–30, March 1996.

[15] L. Zelnik-Manor and M. Irani. Multiview constraints on homographies. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(2):214–223, 2002.