Robust Plane-Based Structure From Motion

Zihan Zhou
University of Illinois at Urbana-Champaign
zzhou7@illinois.edu

Hailin Jin
Adobe Systems Inc.
hljin@adobe.com

Yi Ma
Microsoft Research Asia
mayi@microsoft.com

Abstract

We introduce a new approach to structure and motion recovery directly from one or more large planes in the scene. When such a plane exists, we demonstrate how to automatically detect and track it robustly and consistently over a long video sequence, and how to efficiently self-calibrate the camera using the homographies induced by this plane. We build a complete structure from motion system which does not use any additional off-the-plane information about the scene, and show its advantage over conventional systems in handling two important issues which often occur in real world videos, namely, the plane degeneracy and the dynamic foreground problems. Experimental results on a variety of real video sequences verify the effectiveness and efficiency of our system.

1. Introduction

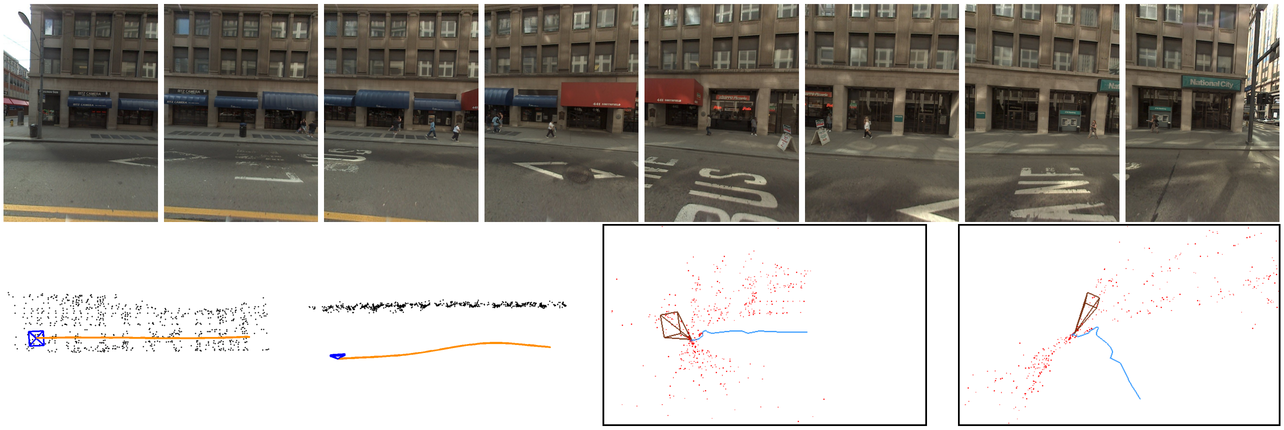
Structure from motion (SFM) has long been an active research topic in computer vision. Recently, thanks to the increasing demands of industrial applications such as virtual reality, navigation, robotics and film production, significant progresses in the SFM techniques have been made in terms of its scalability and reliability [15, 18]. In order to recover the 3D scene and camera motion from a video sequence, conventional SFM systems often rely on detecting, matching and tracking a number of *feature points* (e.g., corners or SIFT features) over frames [14]. One great advantage of working with point features is that the system can be somewhat oblivious to the scene: the scene could be of any shape or texture as long as the scene structure is *general* and motion is a single *rigid body*.

In practice, however, the scenes often exhibit strong structural regularities (or degeneracies), which are largely ignored by existing general-purpose SFM systems. Among all types of regularities, the presence of a planar surface (e.g. the ground or a building facade) is arguably the most common one in commercial or consumer videos, see Figure 1 for an example. Intuitively, the presence of such regularities provides opportunities for constraining and simplifying the reconstruction task. Therefore, if an SFM system can take the advantage of such information, it is natural to expect it to achieve more efficient, robust, and accurate reconstruction. Rather surprisingly, such regularities actually pose significant challenges for conventional SFM systems and could even greatly complicate the reconstruction process. For example, the presence of a dominant plane in the scene, which is very common in man-made environments and aerial videos, violates the general structure assumption of traditional methods, leading to ambiguous and even meaningless solutions.

In this paper, our goal is to develop a reliable SFM system that can explicitly take advantage of the presence of a (relatively dominant) plane in the scene. To this end, the first task obviously is to automatically detect such a plane in the scene from a given image sequence. This turns out to be not so trivial at all. For instance, one may attempt to detect planes between adjacent image pairs and combine the detection result across multiple pairs. However, since the camera motion between two adjacent frames is usually small, the detection result is very sensitive to noise. Further, the planes detected in different pairs of images may not be consistent with each other. Another practical difficulty is the presence of dynamic foreground in the scene. In fact, large majority of commercial or consumer videos consist of one or more moving objects, violating the single rigid body requirement, see Figures 2 and 3 for examples. Those objects, if not properly handled, could lead to huge detection and reconstruction errors in the final results. In this paper, we assume that for most cases of interests, a relatively dominant plane, if existing in the scene, belongs to a static background, and the foreground consists of out-of-plane structures and possibly other independently moving objects. Our goal is hence to robustly detect the plane and accurately recover the static part of the scene, despite severe corruption by dynamic outliers.

1.1. Related Work

It has been known in the literature that prior knowledge about the scene planes can greatly facilitate the 3D reconstruction problem. For instance, [1] uses user-provided geometry about a piecewise planar scene to constrain the estimation of structure and motion parameters. [13] shows



(a) (b) (c) (d)

Figure 1. “Google Street View” example. **Top row:** Eight snapshots of the input video from Google Street View taken by a smoothly moving camera mounted on the car. **(a) and (b):** Frontal and top view of the reconstruction results of our plane-based SFM algorithm. **(c) and (d):** Incorrect reconstruction result from one of the state-of-the-art systems [21].

that the relationship between uncalibrated cameras and 3D scene points is linear with a known reference plane, and can be solved simultaneously via a linear algorithm. However, these methods require users to provide necessary information about the planes.

The problem of plane degeneracy in multi-view structure from motion has also been previously addressed. Several papers have tried to detect the degenerated frames and exclude them from the initial projective reconstruction by either fitting an average planar homography [14] between two frames or using some other statistic measures [19, 16]. Alternatively, [3] proposes a RANSAC-based algorithm for robust estimation of the epipolar geometry. These methods often substantially complicate the SFM system, and are not always reliable in practice, as noticed by [21]. Also, these methods assume the existence of enough out-of-plane structure, at least in certain part of the video, which may be unrealistic for many practical scenarios.

A popular method for detecting planar structure between two frames is to use RANSAC [5]. In this paper, we show how to extend this method to produce consistent plane models over long video sequences. Recently, [17] proposes a model selection method for multiple-frame plane detection using the Minimal Description Length (MDL) principle. While it focuses on discovering multiple plane models simultaneously, its robustness to gross outliers is unknown.

Finally, with the seminal work by Triggs [20], various approaches for camera self-calibration from a planar scene have been developed over the past decade. However, many of these methods requires additional assumptions on the data (e.g., fronto-parallelism of the key image) or user input to initialize the local optimization algorithm [11, 8, 12]. Assuming that only the constant focal length is unknown, a global solution is derived in [2]. But this method does not scale beyond a small number of views, hence not suitable for our purpose.

1.2. Contributions of this Paper

In this paper, we propose a novel and complete automatic SFM system specifically designed to exploit the useful properties of scene planes, meanwhile avoiding the aforementioned difficulties of conventional methods. We show how to automatically and robustly detect a scene plane (if present) and obtain accurate information about the cameras and structures directly from the plane, *without using any additional off-the-plane information about the scene*. Our method can handle multiple planes in the scene in a unified manner, and there is no need for images in the sequence to share a common plane (see the “Wall” and “Office Desk” sequences in Figure 5). As a result, our system produces clean, simple and visually plausible models for various challenging commercial or consumer videos on which conventional SFM systems often fail.

Figure 1 shows an example of successful reconstruction of a challenging sequence captured by Google Street View[1] using our method. Such sequences are of great importance to the computer vision community nowadays due to the increasing interest in building large-scale 3D models for urban area from the industry. However, conventional SFM systems often perform very poorly on them because (1) most of the tracked point trajectories lie on a plane (i.e., the building facade) in the scene and (2) there exists a significant amount of outliers due to the reflection of window glasses, moving objects, etc. As one can see in Figure 1, the reconstruction result by one of the state-of-the-art SFM systems [21] is obviously wrong.

The success of our system relies on several technical improvements over existing methods and systems, with the following notable advantages:

- We develop a novel method called TRASAC (TRAjectory SAmpling Consensus) for robust plane detec-

[1] www.google.com/streetview

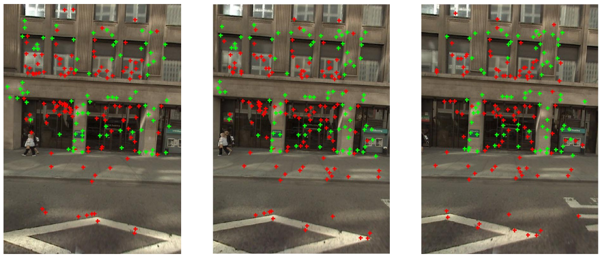


Figure 2. Three consecutive frames of the “Google Street View” sequence with the detected plane (building facade) using our method. Green dots correspond to the inlying points on the plane, red dots correspond to outliers. Note the outliers on the window glasses due to reflection.

tion and tracking from video sequences. This method generalizes the classical two-frame RANSAC to estimate consistent plane models across multiple views, and has a very high breakdown point to gross outliers. This ensures that our method is much more robust than conventional SFM methods which utilize epipolar geometry for outlier rejection or two-frame RANSAC.

- We propose a fully automatic plane-based self-calibration approach, which is fast, easy to implement and yet able to reliably handle practical sequences that have significant varying focal lengths. This makes our system very robust to initialization of the camera calibration and significantly enhances its applicability to commercial or consumer videos.
- Another advantage of our method is that the motion parameters for all the cameras are initialized globally. In contrast, most traditional SFM methods such as [6, 15, 21] employ an incremental method, i.e., they solve for progressively larger sets of images. Incremental methods are known to be sensitive to the initialization and amenable to local minima. Furthermore, our method is significantly more efficient than existing work for obtaining global initialization using a hybrid discrete-continuous optimization method [4].

Admittedly, our new method cannot yet handle all potential cases that arise from real applications, especially when there is a lack of clear planar structures in the scene. Hence, it should not be regarded as a replacement or competitor to the existing SFM systems. Rather, it should be considered complementary, and of great enhancement when properly integrated, to the general-purpose systems.

2. Overview of the Method

Before introducing our method, we review some notations and backgrounds of the multi-view geometry [9, 10]. Suppose a rigid scene is viewed by N cameras, we use $K_i \in \mathbb{R}^{3\times 3}$ to denote the intrinsic matrix of the i-th camera. Without loss of generality, we choose the world coordinate frame to be the camera frame of the first camera, and use $R_i \in SO(3)$ and $\boldsymbol{t}_i \in \mathbb{R}^3$ to denote the Euclidean transformation from the world coordinate frame to the i-th camera frame.

For a piecewise planar scene with P planes, we assume that a 3D plane π_k ($1 \leq k \leq P$) has coordinates $\pi_k = (\boldsymbol{n}_k, d_k)^T$ with respect to the world coordinate frame, where $\boldsymbol{n}_k$ is the unit normal vector and $d_k > 0$ denotes the distance from the plane to the world origin. Therefore, for any point $X \in \mathbb{R}^3$ on it we have $\boldsymbol{n}_k^T X = d_k$.

Consider the situation in which we observe a set of trajectories $\mathcal{T} = \{T_j\}_{j=1}^M$ of M feature points. For each T_j, let p_j and q_j ($1 \leq p_j < q_j \leq N$) denote its starting and ending frames, respectively. We can therefore write $T_j = \{\boldsymbol{x}_j^i\}_{i=p_j}^{q_j}$, where $\boldsymbol{x}_j^i \in \mathbb{P}^2$ is the homogeneous coordinates of the j-th point as seen by the i-th camera. We also use $\mathcal{T}^{ab} = \{T_j \in \mathcal{T} : p_j \leq a, q_j \geq b\}$ to represent the set of trajectories which span the a-th and b-th frames.

Finally, if a tracked point lies on π_k, the coordinates of the first frame and the i-th frame are related by a planar homography $\boldsymbol{x}_j^i = H_i \boldsymbol{x}_j^1$ where H_i can be written as:

$$H_i \simeq K_i(R_i + \boldsymbol{t}_i \boldsymbol{n}_k^T / d_k) K_1^{-1}, \tag{1}$$

with the symbol $\simeq$ meaning “equality up to a scale”.

Our approach takes the feature point trajectories obtained by any standard tracking algorithm as input. To measure the fitness of a plane model to a trajectory T_j, we use the sum of the squares of the standard Euclidian image distance in the i-th image, $\|\boldsymbol{x}_j^i - H_i \boldsymbol{x}_j\|^2$, for all i’s between p_j and q_j. Note that here we use $\boldsymbol{x}_j$ as the (to be estimated) true feature point location in the first frame. This is different from $\boldsymbol{x}_j^1$, the (possibly noisy) 2D measurement of the same quantity.

Our goal is then to partition all the trajectories into groups, each corresponding to a plane in the scene, plus a set of trajectories which are labeled as *outliers*. We emphasize that an outlier may either come from non-planar structures of a static scene (e.g., trees), or dynamic foreground objects (e.g., moving cars). Define S_k as the set of indices of the trajectories which belong to the k-th plane, and S_0 as the set of outlying trajectories, we can now formulate our structure and motion recovery problem as minimizing the following geometric error function:

$$\sum_{k=1}^{P}\sum_{j\in S_k}\sum_{i=p_j}^{q_j}\|\boldsymbol{x}_j^i - K_i(R_i + \frac{\boldsymbol{t}_i}{d_k}\boldsymbol{n}_k^T)K_1^{-1}\boldsymbol{x}_j\|^2 + \sum_{j\in S_0}\sum_{i=p_j}^{q_j}\eta^2, \tag{2}$$

where η is the penalty for labeling a trajectory as an outlier.

In order to minimize this nonlinear function, we use an alternating method, which iterates between updating the plane models and assigning each trajectory to current plane candidates. Like other local methods, a set of good initial values of the unknowns are crucial for the algorithm to converge to the desired solution. In this paper, we propose to find such a good initialization using a two-stage approach. First, we detect and track each plane using a robust algorithm, yielding a set of inter-image homographies induced

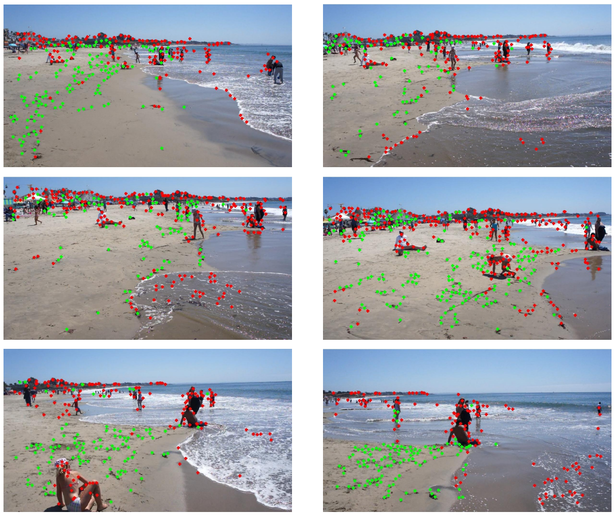


Figure 3. Selected frames of the "Beach" sequence with classified trajectories using TRASAC. Green: inliers. Red: outliers.

by the planes (Section 3). Second, we develop a plane-based self-calibration method which takes the homography matrices as the input and outputs the structure and motion parameters (Section 4). This is followed by the aforementioned alternating scheme which refines all the parameters (Section 5). We illustrate the performance of our method in Section 6 and conclude our discussion in Section 7.

3. Robust Plane Detection and Tracking

In this section, we describe a novel method called TRASAC, which is a generalization of the RANSAC estimator, for detecting and tracking *one* plane in the video sequence. To obtain all the planes one can simply apply this method sequentially by removing the inliers of the current plane after each iteration.

The novelty of our method is that instead of independently sampling point correspondences between every two frames, it directly samples the feature point trajectories. By doing so, we assume that if a trajectory is classified as an inlier within any pair of frames, it remains as an inlier to the same plane for all the other frames it spans. Compared to the two-frame RANSAC, the advantage of our new method is two-folded: First, it directly generates a consistent plane model over the entire sequence – no linking is needed as a post-processing step. Second, it enables us to use only trajectories with *known* membership to estimate the homographies induced by the same plane in the rest of the frames. In this way, we derive an efficient algorithm with very high tolerance to (possibly dominant) outliers in the scene.

Algorithm 1 (TRASAC)

```
Input: A set of $M$ trajectories $\mathcal{T}$ over $N$ frames. A distance threshold $\epsilon$.
repeat for $n$ trials:
    Select a random pair of frames $(F_{i-1}, F_i)$ from $\mathcal{C}$.
    Select a random sample of four trajectories from $\mathcal{T}^{(i-1)i}$ and compute the homography $H_{(i-1)i}$.
    Classify each $T_j \in \mathcal{T}^{(i-1)i}$ into $\mathcal{T}_{in}$ or $\mathcal{T}_{out}$ according to $H_{(i-1)i}$.
    while not all pairs in $\mathcal{C}$ are processed
        Select a new pair of frames $(F_{k-1}, F_k)$.
        if $|\mathcal{T}_{in} \bigcap \mathcal{T}^{(k-1)k}| \leq 4$; break; end if
        Compute $H_{(k-1)k}$ using two-frame RANSAC estimation from trajectories in $\mathcal{T}_{in} \bigcap \mathcal{T}^{(k-1)k}$.
        Classify all the unclassified trajectories in $\mathcal{T}^{(k-1)k}$ into $\mathcal{T}_{in}$ or $\mathcal{T}_{out}$ according to $H_{(k-1)k}$.
    end while
end repeat
Choose the set of homographies $\{H_{(i-1)i}\}_{i=2}^{N}$ from the trial with the largest number of inliers $|\mathcal{T}_{in}|$.
Compute the homography between the first and the $i$-th frame recursively using $\{H_{(i-1)i}\}_{i=2}^{N}$: $H_1 = I_{3\times3}, H_i = H_{(i-1)i}H_{i-1}, i = 2, \ldots, N$.
Output: A set of inter-image homographies $\{H_i\}_{i=1}^{N}$.
```

We now discuss our method in full details. Since our method is based on sampling consensus, it consists of multiple trials of the same procedure followed by a selection of the best result from these trials. We first describe the procedure of one trial, which contains two steps (Step 1 and 2 below). Note that given an input sequence, our method operates in an incremental manner, processing two adjacent frames at a time. Therefore, for each trial, we maintain the sets of trajectories which are classified as inliers and outliers, $\mathcal{T}_{in}$ and $\mathcal{T}_{out}$, respectively. They are both empty at the beginning, and expanded accordingly after processing each image pair.

Step 1: Random sampling. Given an input sequence of N frames $\{F_i\}_{i=1}^{N}$, we form $N-1$ pairs of adjacent frames $\mathcal{C} = \{(F_1, F_2), (F_2, F_3), \ldots, (F_{N-1}, F_N)\}$. Our algorithm starts with a randomly chosen pair in $\mathcal{C}$, say (F_{i-1}, F_i). Then, a putative plane model between these two frames is generated using a random minimum subset of samples. More precisely, 4 randomly chosen trajectories in $\mathcal{T}^{(i-1)i}$ are used to estimate a homography matrix $H_{(i-1)i}$. Then, given a fixed threshold ϵ, we classify each trajectory $T_j \in \mathcal{T}^{(i-1)i}$ into $\mathcal{T}_{in}$ or $\mathcal{T}_{out}$ by comparing the projection error $\|\boldsymbol{x}_j^i - H_{(i-1)i}\boldsymbol{x}_j^{i-1}\|$ with ϵ.

Step 2: Computation of the consensus. Next, we choose a new pair of frames which is adjacent to the previous pair, say (F_i, F_{i+1}),[2] and compute the set of trajectories in $\mathcal{T}^{i(i+1)}$ which have already been labeled as inliers, i.e., $\mathcal{T}_{in} \bigcap \mathcal{T}^{i(i+1)}$. These trajectories are then used as candidates to estimate the homography $H_{i(i+1)}$. In the ideal case, any 4 or more samples from this set should do the job equally well because they are all inliers. However, to en-

[2] The other adjacent pair is (F_{i-2}, F_{i-1}). We do not make any preference among these two choices.

sure the estimation quality in the presence of image noise, we generate a small number of model hypotheses and select the one with the largest number of inliers. We repeat this step for each pair of adjacent frames, until all the frames are processed or there are not enough inliers to proceed.

Selection of the best model. After repeating Steps 1 and 2 for enough times, the plane model (i.e., a set of homographies) estimated from the trial with the largest total number of inliers across the entire sequence is kept as the output.

We summarize the complete procedure as Algorithm 1. The only parameter for our method is the distance threshold ϵ. Since our goal is to detect those large scene planes, we find that a fixed value $\epsilon = 4$ (pixels) works well enough in practice. Figure 3 shows an example of the detected plane (the ground) in the "Beach" sequence by TRASAC. As one can see, the plane detected by our method is consistent despite large number of outliers in certain frames.

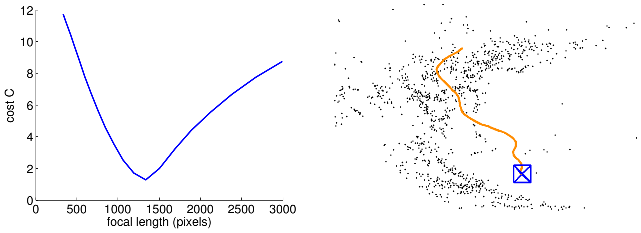
4. Plane-Based Self-Calibration

In this section, we discuss how to self-calibrate a camera using only the set of homographies $\mathcal{H} = \{H_i\}_{i=1}^N$ induced by a scene plane. We assume the camera to have a zero pixel skew and known aspect ratio, which is true for most modern digital cameras. We also assume that the principal point coincides with the image center, as the error introduced by this approximation is normally well within the region of convergence of the subsequent nonlinear optimization. As a result, the self-calibration problem is reduced to finding the focal length for each frame. Inspired by the work of [7], we propose to enumerate the inherently bounded space of focal lengths and examine the tentative metric reconstruction produced by each sample. In the rest of this section, we first describe our method for the constant focal length case in details. Then we will show how to generalize this method to handle varying focal length.

4.1. Self-Calibration with Constant Focal Length

Our self-calibration method is based on two important observations. First, if the focal length f (or equivalently the matrix K) is given, then there are at most two physically possible solutions for a decomposition of any H into parameters $\{R, \tilde{\boldsymbol{t}}, \boldsymbol{n}\}$ where $\tilde{\boldsymbol{t}} = \boldsymbol{t}/d$ (see e.g. [10]). Second, the space of possible values of f is inherently bounded by the finiteness of the acquisition devices. We assume $f \in [0.3f_0, 3f_0]$ where f_0 is defined as the sum of half width and half height of the image and propose the following two-stage method:

1. Given a guess on f, compute the plane normal $\boldsymbol{n}$ from the homography induced by any two frames.[3] This yields at most two physically possible normals. For each $\boldsymbol{n}$, estimate $\{R_i, \tilde{\boldsymbol{t}}_i\}_{i=2}^N$ for all cameras.
2. Enumerate the space of focal length (a subset of $\mathbb{R}$) and score each focal length f based on how well the recovered structure and motion parameters fit the homographies.



(a) (b)

Figure 4. (a) Score C as a function of f for the "Beach" sequence. It is minimized at the true focal length. (b) Reconstruction result.

The best solution is then obtained according to the scores. We now elaborate each step in details.

Planar homography decomposition. Given an estimate for the focal length, we can compute the Euclidean homography matrix as: $\hat{H}_i = K^{-1}H_iK$. $\hat{H}_i$ is related to $\{R_i, \tilde{\boldsymbol{t}}_i, \boldsymbol{n}\}$ as follows:

$$\hat{H}_i = \lambda_i(R_i + \tilde{\boldsymbol{t}}_i\boldsymbol{n}^T). \tag{3}$$

It turns out that there are only four solutions for decomposing $\hat{H}_i$ to $\{R_i, \tilde{\boldsymbol{t}}_i, \boldsymbol{n}\}$. The positive depth constraint can be imposed to reduce the number of physically possible solutions to two. We refer the reader to [10] for more details.

Estimation of the focal length. As mentioned before, our self-calibration algorithm determines the focal length f by enumerating all of its possible values and checking how well the resulting camera parameters $\{R_i, \tilde{\boldsymbol{t}}_i\}_{i=2}^N$ and plane normal $\boldsymbol{n}$ fit the homographies $\{\hat{H}_i\}_{i=2}^N$ where $\hat{H}_i = K^{-1}H_iK$. Once a set of parameters are obtained for a given f, there are several ways to score them. In this paper, we adopt the cost function used in [11], which compares the normalized difference of the two non-zero singular values σ_i^1 and σ_i^2 ($\sigma_i^1 \geq \sigma_i^2$) of the matrix $\hat{H}_i\hat{\boldsymbol{n}}$:

$$C = \sum_{i=2}^{N} \frac{\sigma_i^1 - \sigma_i^2}{\sigma_i^1}. \tag{4}$$

The computational complexity of our self-calibration algorithm is linear in the number of samples of f. Figure 4(a) shows a plot of the score as a function of focal length for the "Beach" sequence. As one can see, the correct focal length can be easily determined as the minimizing point on the curve. Once the camera is calibrated, the camera motion and scene points can be recovered as shown in Figure 4(b).

4.2. Handling Varying Focal Length

Our method can be easily generalized to handle the varying focal length case. Instead of sampling $f \in \mathbb{R}$, we sample all possible values of $(f_1, f_N) \in \mathbb{R}^2$ (the first and last

[3] In this paper, we always choose the homography H_N between the first frame and the last frame for computing $\boldsymbol{n}$.

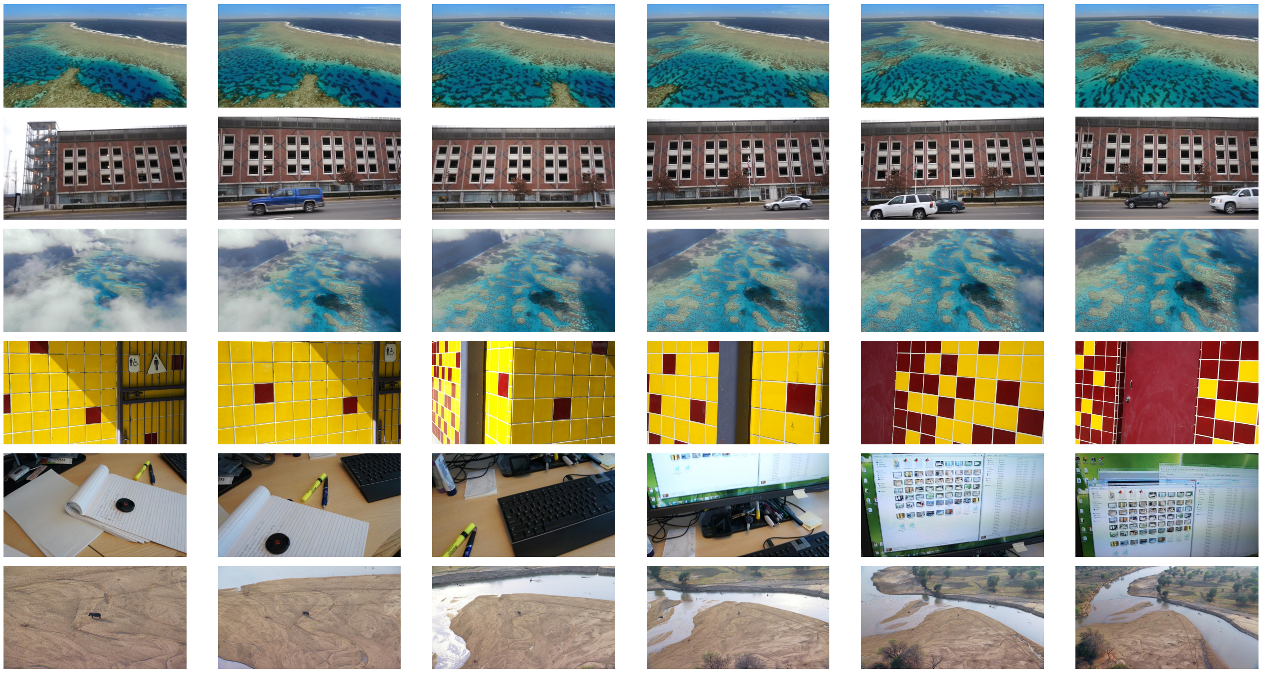


Figure 5. Snapshots of several testing sequences. **From top to bottom:** “Seashore”, “Street”, “Shallow Sea”, “Wall”, “Office Desk” and “Lonely Hippo”.

cameras are chosen for convenience) and compute the plane normal $\boldsymbol{n}$ as described before. Compared to the constant focal length case, the extra work required is to compute the focal length for other images $f_2, \dots, f_{N-1}$. We note that $K_i^{-1}H_iK_1$ has to preserve the length of any vectors inside the subspace perpendicular to $\boldsymbol{n}$ (see details in [10]). Let $\boldsymbol{u}, \boldsymbol{v}$ be two unit vectors in that subspace, the length constraint dictates

$$\|K_i^{-1}H_iK_1\boldsymbol{v}\|^2 = \|K_i^{-1}H_iK_1\boldsymbol{u}\|^2. \tag{5}$$

Equation (5) is a linear equation in f_i^2 which can be easily solved to obtain f_i.

5. Optimal Structure and Motion Recovery

With a good initialization of all parameters, we solve the global optimization problem (2) using an alternating algorithm. On one hand, given the labeling $\{S_k\}_{k=0}^P$, (2) becomes:

$$\begin{aligned}&\min f(\boldsymbol{x}_j, K_i, R_i, \boldsymbol{t}_i, \boldsymbol{n}_k, d_k)\\ &= \sum_{k=1}^{P}\sum_{j\in S_k}\sum_{i=p_j}^{q_j} \|\boldsymbol{x}_j^i - K_i(R_i + \boldsymbol{t}_i\boldsymbol{n}_k^T/d_k)K_1^{-1}\boldsymbol{x}_j\|^2,\end{aligned}$$

which can be solved via the Levenberg-Marquardt (LM) method. On the other hand, given the structure and motion parameters, we can update the index sets $\{S_k\}_{k=0}^P$. For the trajectory T_j, let

$$f_j(k) = \sum_{i=p_j}^{q_j} \|\boldsymbol{x}_j^i - K_i(R_i + \boldsymbol{t}_i\boldsymbol{n}_k^T/d_k)K_1^{-1}\boldsymbol{x}_j\|^2,$$

we assign T_j to class k^* using the following rule:

$$k^* = \begin{cases} 0 & \text{if } \min_k f_j(k) > (q_j - p_j + 1)\eta^2 \\ \arg\min_k f_j(k) & \text{otherwise} \end{cases}$$

Full 3D reconstruction. To recover the full 3D structure, we back-project all the points on the plane to obtain their 3D positions. In addition, we can triangulate the positions of the off-the-plane points. We employ the standard 3D bundle adjustment to get the optimal estimates of all structure and motion parameters.

6. Experiments

We have tested our algorithm on more than 50 video sequences captured by a variety of cameras. These sequences cover a wide range of scenes with one or more large planes, from both natural and indoor/outdoor man-made environments. In the section, we report the reconstruction results of our method on several representative examples, which are shown in Figure 5. In terms of speed, for a typical sequence such as “Beach” with 660 frames, our system chooses 44 keyframes and reconstructs 1479 3D points, which takes about 50 seconds on a desktop PC with Intel Xeon 2.67GHz CPU and 24GB memory.

To better understand the reconstruction quality and the advantage of our method, we further compare our method against one of the state-of-the-art general-purpose SFM system, ACTS [21]. We have also tested Bundler [18] and Voodoo Camera Tracker[4] on these sequences. However,

[4] www.digilab.uni-hannover.de/docs/manual.html

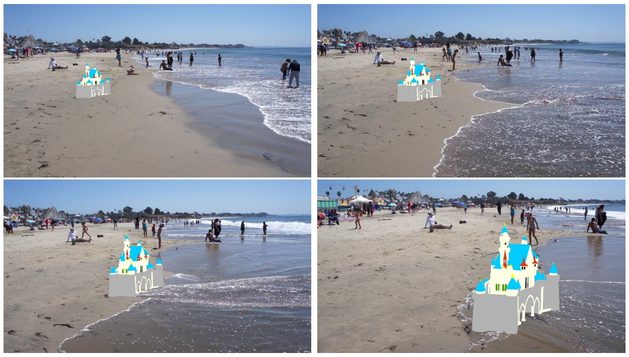
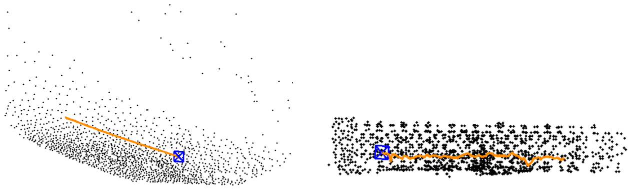


Figure 6. Some augmented images of the “Beach” sequence using the reconstruction result obtained by our method.



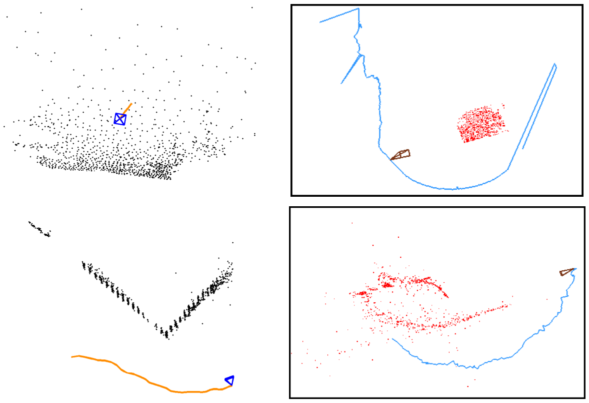
(a) Seashore (b) Street

Figure 7. Some reconstruction results of our method.

Bundler is designed for unordered large-baseline images and computes point correspondences between each pair of images, hence is very inefficient for our purpose. Also, it assumes known camera intrinsic parameters. For Voodoo, we found that its performance is generally worse than ACTS. Therefore, in the interest of space, we do not report their results in the paper.

According to the performance of ACTS, we roughly partition the test sequences into two categories. The first category consists of planar scenes with no or little 3D structure throughout the entire sequence, whereas the second category contains videos with certain 3D structure in at least a fraction of the frames (e.g., the “Beach” and “Office Desk” sequences). As expected, while sequences in the first category are considered easy to our method, ACTS performs poorly on them, generating incomplete or obviously wrong results. For the second category, ACTS is able to obtain reasonable solutions, thanks to its ability to detect key frames with enough 3D structures for initialization. For these sequences, we further demonstrate the reconstruction quality of our method by inserting virtual objects to the videos.

The “Beach” sequence. This is a representative example with both large dynamic foreground (sea waves, running people) and planar scene structure (Figure 3). We have already seen the reconstruction result of our method in Figure 4(b). Here, we further examine the reconstruction result of our method by augmenting the video with a synthetic object. As one can see in Figure 6, the castle in our result remains firmly registered to the scene, implying the reconstruction by our method is very accurate.



(a) our method (b) ACTS

Figure 8. Comparison of reconstruction results. **First row:** The “Shallow Sea” sequence. **Second row:** The “Wall” sequence.

The “Seashore” sequence. This sequence is taken by an aerial camera moving forward along the seashore. Because the scene is completely flat, ACTS crashes on this example. The reconstruction result of our method is shown in Figure 7(a).

The “Street” sequence. This is an example of planar scene in man-made environments with dynamic foregrounds (i.e., cars). The planar structure and camera motion are easily obtained by our method, as shown in Figure 7(b), while ACTS generates completely wrong result.

The “Shallow Sea” sequence. This is another example of a planar scene with large dynamic foreground (i.e., the clouds). In this sequence the camera is smoothly moving forward, which is correctly recovered by our method, as shown in Figure 8. However, ACTS fails in this case possibly due to lack of static 3D structure in the scene.

The “Wall” sequence. We use this somewhat extreme example to test the ability of both systems in handling multiple planes. As one can see in Figure 8, the structure recovered by our system is very accurate, with a clean right angle between the two walls. In contrast, ACTS generates incorrect structure in this case.

The “Office Desk” sequence. This scene also contains two large planes, the desk and the computer monitor. In addition, as one can see in Figure 9, the synthetic object in our method’s augmented video remains very steady throughout the sequence. This further evidences the advantage of using information encoded by scene planes for accurate reconstruction.

The “Lonely Hippo” sequence. Lastly, we test our method on a sequence with a smoothly zooming-out camera. It is very challenging in that the focal length changes by a factor of 8 between the first frame and the last frame. Figure 10 shows the estimated focal lengths as well as the reconstruction result by our method, verifying its effectiveness in handling varying focal length.

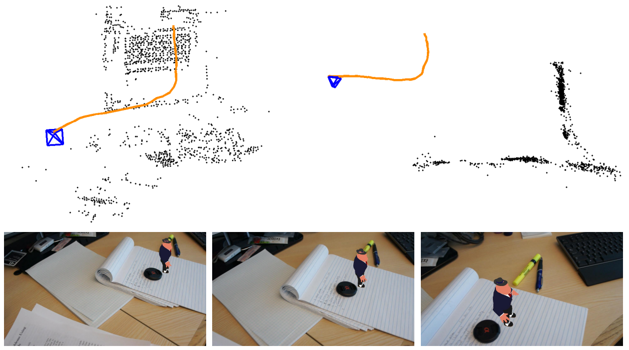


Figure 9. Reconstruction results of the "Office Desk" sequence. **First row:** Two views of the result of our method. **Second row:** Augmented frames by our method.

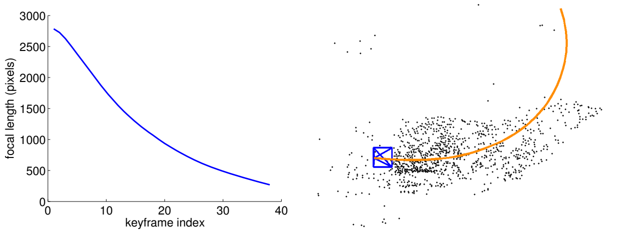


Figure 10. The "Lonely Hippo" sequence with varying focal length. **Left:** Estimated focal lengths. **Right:** Reconstruction result.

7. Conclusion

In this paper, we have proposed a novel and complete SFM system which produces very accurate reconstruction result by directly analyzing the geometry information encoded by large scene planes. The system consists of two main components, namely, a new method to detect and track the planes consistently across the entire sequence and an efficient multiple-view self-calibration algorithm based on the homographies induced by the scene plane. We show that by taking advantage of the presence of planar structures in the scene, our method avoids the difficulties of conventional SFM techniques in handling plane degeneracy and dynamic foreground, hence highly complements those techniques in processing real-world commercial and consumer videos.

8. Acknowledgement

This work is supported in part by Adobe Systems Incorporated. Zihan Zhou and Yi Ma gratefully acknowledge support by ONR N00014-09-1-0230, NSF CCF 09-64215, and NSF IIS 11-16012.

References

[1] A. Bartoli and P. Strum. Constrained struture and motion from multiple uncalibrated views of a piecewise planar scene. *IJCV*, 52(1):45–64, 2003.

[2] B. Bocquillon, P. Gurdjos, and A. Crouzil. Towards a guaranteed solution to plane-based self-calibration. In *ACCV*, pages 11–20, 2006.

[3] O. Chum, T. Werner, and J. Matas. Two-view geometry estimation unaffected by a dominant plane. In *CVPR*, pages 772–779, 2005.

[4] D. Crandall, A. Owens, N. Snavely, and D. Huttenlocher. Discrete-continuous optimization for large-scale structure from motion. In *CVPR*, 2011.

[5] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395, 1981.

[6] A. W. Fitzgibbon and A. Zisserman. Automatic camera recovery for closed or open image sequences. In *ECCV*, 1998.

[7] R. Gherardi and A. Fusiello. Practical autocalibration. In *ECCV*, 2010.

[8] P. Gurdjos and P. Sturm. Methods and geometry for plane-based self-calibration. In *CVPR*, 2003.

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

[10] Y. Ma, J. Košecká, S. Soatto, and S. Sastry. *An Invitation to 3-D Vision, From Images to Models*. Springer-Verlag, New York, 2004.

[11] E. Malis and R. Cipolla. Camera self-calibration from unknown planar structures enforcing the multi-view constraints between collineations. *PAMI*, 24(9):1268–1272, 2002.

[12] J. F. Menudet, J. M. Becker, T. Fournel, and C. Mennessier. Plane-based camera self-calibration by metric rectification of images. *Image Vision Comput.*, 26:913–934, July 2008.

[13] R. K. Nicolas, N. Dano, and R. Hartley. Plane-based projective reconstruction. In *ICCV*, pages 420–427, 2001.

[14] 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):207–232, 2004.

[15] M. Pollefeys, D. Nisté, J.-M. Frahm, A. Akbarzadeh, P. Mordohai, B. Clipp, C. Engels, D. Gallup, S.-J. Kim, P. Merrell, C. Salmi, S. Sinha, B. Talton, L. Wang, Q. Yang, H. Stewéius, R. Yang, G. Welch, and H. Towles. Detailed real-time urban 3d reconstruction from video. *IJCV*, 78:143–167, 2008.

[16] M. Pollefeys, F. Verbiest, and L. V. Gool. Surviving dominant planes in uncalibrated structure and motion recovery. In *ECCV*, pages 837–851, 2002.

[17] J. Prankl, M. Zillich, B. Leibe, and M. Vincze. Incremental model selection for detection and tracking of planar surfaces. In *BMVC*, pages 1–12, 2010.

[18] N. Snavely, S. Seitz, and R. Szeliski. Modeling the world from internet photo collections. *IJCV*, 80:189–210, 2008.

[19] P. H. S. Torr, A. W. Fitzgibbon, and A. Zisserman. The problem of degeneracy in structure and motion recovery from uncalibrated image sequences. *IJCV*, 32(1):27–44, 1999.

[20] B. Triggs. Autocalibration from planar scenes. In *ECCV*, 1998.

[21] G. Zhang, X. Qin, W. Hua, T.-T. Wong, P.-A. Heng, and H. Bao. Robust metric reconstruction from challenging video sequences. In *CVPR*, 2007.