Degeneracies in Rolling Shutter SfM

Cenek Albl[1], Akihiro Sugimoto[2(✉)], and Tomas Pajdla[1]

[1] Czech Technical University in Prague, Prague, Czech Republic
{alblcene,pajdla}@cmp.felk.cvut.cz
[2] National Institute of Informatics, Tokyo, Japan
sugimoto@nii.ac.jp

Abstract. We address the problem of Structure from Motion (SfM) with rolling shutter cameras. We first show that many common camera configurations, e.g. cameras with parallel readout directions, become critical and allow for a large class of ambiguities in multi-view reconstruction. We provide mathematical analysis for one, two and some multi-view cases and verify it by synthetic experiments. Next, we demonstrate that bundle adjustment with rolling shutter cameras, which are close to critical configurations, may still produce drastically deformed reconstructions. Finally, we provide practical recipes how to photograph with rolling shutter cameras to avoid scene deformations in SfM. We evaluate the recipes and provide a quantitative analysis of their performance in real experiments. Our results show how to reconstruct correct 3D models with rolling shutter cameras.

Keywords: Structure from motion · Rolling shutter · Degeneracy · Non-perspective cameras

1 Introduction

Structure from Motion (SfM) reconstructs geometry of scenes from their images while simultaneously estimating camera poses and (some of) their internal parameters [5]. SfM has many practical applications in scene modelling, 3D mapping, and visual odometry [12,16,18]. Typical SfM considers perspective cameras, incrementally performs [16] and includes relative and absolute camera pose computation and bundle adjustment (BA) [17]. Recently, rolling shutter cameras became very important [11] since the rolling shutter is present in vast majority of current CMOS image sensors in consumer cameras and smart-phones. In rolling shutter cameras, images are not captured at once. They are scanned either along image rows or columns [13]. Since different image lines are exposed at different times, camera movement during the exposure produces image distortions. It has been shown that rolling shutter distortion can severely influence SfM computation [6,14] and that special care has to be taken to achieve sensible results.

Authors of [6] addressed the problem of SfM from video sequences and presented specially adapted BA algorithm for rolling shutter videos [7]. In [9] a SfM pipeline for cellphone videos is presented using video sequences and fusion with

B. Leibe et al. (Eds.): ECCV 2016, Part V, LNCS 9909, pp. 36–51, 2016.
DOI: 10.1007/978-3-319-46454-1_3

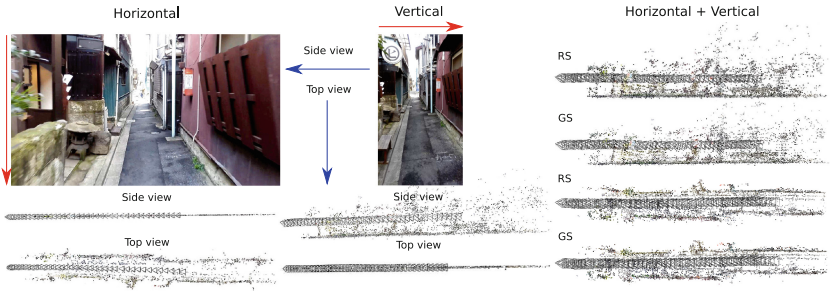

Fig. 1. SfM with rolling shutter model can deliver undesired results. (Left) A reconstruction from forward camera translation with vertical readout direction. (Middle) A reconstruction of the same scene from forward moving camera horizontal readout direction. In both cases, the scene collapses into a plane that is perpendicular to the readout direction. (Right) When both image directions are combined a correct reconstruction is obtained with rolling shutter (RS) projection model, which is close to a reconstruction with global shutter (GS) model.

inertial measurements. These works rely on the fact, that video sequences contain images separated closely in time and space and therefore we can interpolate between camera poses. Authors of [1] presented a technique for simultaneously estimating shape and motion of an object with rolling shutter stereo pair and pointed out a degenerate case.

Recently, techniques for computing absolute camera pose have been presented. In [10] authors estimate the camera pose using global optimization. A general minimal solution viable for incremental SfM is presented in [3]. Another minimal solution for translational movements has been shown in [15]. All of these works present camera models that improve the precision of camera pose estimation under rolling shutter image distortions and which are viable candidates for BA optimization in SfM reconstruction.

1.1 Motivation

Although state-of-the-art algorithms for rolling shutter absolute camera pose and BA have shown promising results, to our best knowledge, no one has yet addressed the task of running a complete RS SfM pipeline on general unordered sets of images. This is an important topic since almost all images taken today, even the still ones, can be affected with rolling shutter distortion. Also video sequences, where rolling shutter is most apparent, are often not desirable to be processed frame by frame, because that is a heavy computational load for longer sequences. The framerate available also could not be high enough for the interpolation used by [7]. Another issue is when combining data from different sources, where it is hard or impossible to enforce relationships between the camera poses and their motion. For these reasons, having SfM pipeline for rolling shutter images with no explicit constraints on the camera movement and temporal displacement is desirable.

Rolling shutter camera models describe the camera motion during image capture by various number of additional parameters. This introduces additional dimensions of freedom to the model. For bundle adjustment, a key component of SfM, this can introduce new and undesired local minimum. We observed in practice that the optimization tends to collapse into a degenerate solution which does not correspond to correct reconstruction in most of the cases (see Fig. 1). Although degenerate solutions have been studied for the case of perspective cameras, there has been no study for any of the rolling shutter camera models used today.

1.2 Contribution

The main purpose of this paper is to show the degeneracies introduced by rolling shutter camera models and to study them. The case of planar degeneracy which occurs most often in practice is explained and the reason why bundle adjustment always prefers this solution is given.

We show that the presence of the degenerate solution is dependent on the relative alignment of the input images. Cases where the scene can collapse into a plane for any number of cameras are shown as well as situations where it is not possible.

Our findings are backed by a number of both synthetic and real experiments that confirm the theory. We suggest a way to capture the images in practice such that the scene is reconstructed without any deformation. Again we verify the method on real data.

1.3 Notation and Concepts

A *similarity* transformation $\mathcal{S}$ is a composition of rotation $\mathtt{R}$, translation $\mathbf{T}$ and uniform scaling s, i.e. $\mathcal{S}(\mathbf{X}) = s\,\mathtt{R}\,\mathbf{X} + \mathbf{T}$, where $\mathtt{R}$ is a rotation matrix, $\mathbf{T}$ is a translation vector and s is a scalar. *Image* j is a set of vectors $\mathbf{u}_i^j \in \mathbb{R}^3 \setminus \{\mathbf{0}\}$ with $i = 1, \ldots, n$, $j = 1, \ldots, m$. *Scene* is a set of vectors $\mathbf{X}_i \in \mathbb{R}^3$. We consider only finite scenes for simplicity.

Scene points are projected to image points by cameras as $\alpha_i^j \mathbf{u}_i^j = \pi(\mathtt{P}^j, \mathbf{X}_i)$, where $\mathtt{P}^j$ defines a particular camera projection model used and its parameters, and α_j^j are appropriate non-zero scales. For instance, when projecting by internally calibrated perspective cameras, the projection becomes $\alpha_i^j \mathbf{u}_i^j = \mathtt{R}^j \mathbf{X}_i + \mathbf{C}^j$, with rotation $\mathtt{R}^j$ and camera center $\mathbf{C}^j$.

A collection $\{\mathbf{X}_i, \mathtt{P}^j, \mathbf{u}_i^j\}$ such that $\alpha_i^j \mathbf{u}_i^j = \pi(\mathtt{P}^j, \mathbf{X}_i)$ for some α_i^j is called a *configuration*. We say that configuration $\{\mathbf{X}_i, \mathtt{P}^j, \mathbf{u}_i^j\}$ *explains* images $\mathbf{u}_i^j$. We say that configuration $\{\mathbf{X}_i, \mathtt{P}^j, \mathbf{u}_i^j\}$ is *related* to configuration $\{\mathbf{Y}_i, \mathtt{Q}^j, \mathbf{u}_i^j\}$ by a similarity transformation when there is a similarity transformation of points $\mathbf{Y}_i = \mathcal{S}(\mathbf{X}_i)$ and camera projection models $\mathtt{Q}^j = \mathcal{S}(\mathtt{P}^j)$ such that $\beta_i^j \mathbf{u}_i^j = \pi(\mathtt{Q}^j, \mathbf{Y}_i)$ for some β_i^j. For instance, for internally calibrated perspective cameras with $\mathtt{P}^j = (\mathtt{R}^j, \mathbf{C}^j)$, $\mathcal{S}(\mathtt{P}^j) = (\mathtt{R}^j\mathtt{R}^\top, s\mathbf{C}^j - \mathtt{R}^j\mathtt{R}^\top\mathbf{T})$ since $\pi(\mathcal{S}(\mathtt{P}^j), \mathcal{S}(\mathbf{X_i})) = \mathtt{R}^j\mathtt{R}^\top(s\mathtt{R}\mathbf{X}_i + \mathbf{T}) + s\mathbf{C}^j - \mathtt{R}^j\mathtt{R}^\top\mathbf{T} = s\mathtt{R}^j\mathbf{X}_i + s\mathbf{C}^j = s\alpha_i^j\mathbf{u}_i^j$.

The goal of 3D reconstruction is to explain images $\mathbf{u}_i^j$ by a configuration $\{\mathbf{X}_i, \mathtt{P}^j, \mathbf{u}_i^j\}$ with scene points $\mathbf{X}_i$ measured in a Cartesian coordinate system. Different choices of Cartesian coordinate systems and different choices of measurement units produce configurations that are related by similarity transformations. Moreover, it is well-known that internally calibrated perspective images of a generic scene can be explained by a set S of configurations that with every element C of S contains also all configurations related to C by a similarity transformation [5], i.e. scene points can be reconstructed only up to a similarity transformation.

Therefore, every two configurations related by a similarity transformation will be considered equivalent. This equivalence relation partitions the set of all configurations into equivalence classes. Two configurations in one class are related by a similarity while two configurations in different classes are not related by a similarity. The equivalence class containing all configurations with scene points measured in a Cartesian coordinate system will be termed *correct reconstruction*. All other equivalence classes will be termed *incorrect reconstructions*.

We say that images $\mathbf{u}_i^j$ are *critical* if they can be explained by two configurations that are not equivalent, i.e., by at least one configuration that is in the incorrect reconstruction. Notice that our concept of criticality is somewhat different from concepts used in [4,8], where they studied which scenes and cameras produce critical configurations for perspective images. Here we are interested in analyzing when images may be critical when using a rolling shutter models and we therefore modify the concept accordingly for that purpose.

2 Rolling Shutter Camera Model

In this paper, we consider internally calibrated rolling shutter (RS) camera models, which describe RS cameras that are realized as internally calibrated perspective cameras ($\mathtt{K} = \mathtt{I}$) with the row readout speed equal to one. To simplify the exposition, we will, hereafter, drop the adjective "internally calibrated". Therefore, "perspective model" means "internally calibrated perspective model" and "RS model" means "internally calibrated RS model". Calibrated perspective projection can be described by $\alpha_i \mathbf{u}_i = \mathtt{R}\,\mathbf{X}_i + \mathbf{C}$ where $\mathtt{R} \in SO(3)$ and $\mathbf{C} \in \mathbb{R}^3$ is the rotation and the translation transforming a 3D point $\mathbf{X}_i \in \mathbb{R}^3$ from a world coordinate system to the camera coordinate system with $\mathbf{u}_i = [c_i, r_i, 1]^\top$, and $\alpha_i \in \mathbb{R} \setminus \{0\}$.

In the RS [11] model, when the camera is moving during the image capture, every image row or image column is captured at a different time and hence at a different position. Here we assume that image is captured row by row and therefore the rotation $\mathtt{R}$ and the translation $\mathbf{C}$ are functions of the image row r_i: $\alpha_i \mathbf{u}_i = \alpha_i \left[c_i, r_i, 1\right]^\top = \mathtt{R}(r_i)\mathbf{X}_i + \mathbf{C}(r_i)$. Various models for $\mathtt{R}(r_i)$ and $\mathbf{C}(r_i)$ have been considered [2,3,7,11]. All of them used a linearized translational motion with constant velocity and direction $\mathbf{C}(r_i) = \mathbf{C}_0 + r_i\mathbf{t}$. This approximation can be justified by the fact that the readout times of one frame are short (tens of milliseconds) and there is not much acceleration over this period of time.

The same justification is used for camera rotational velocity, which is considered constant during the frame capture. Rotation $\mathtt{R}(r_i)$ was often modeled as a composition $\mathtt{R}(r) = \mathtt{R}_r(r_i)\mathtt{R}_0$ of a static part $\mathtt{R}_0$ and a motion part $\mathtt{R}_r(r_i)$. The motion part $\mathtt{R}_r(r_i)$ has been parameterized by SLERP [6,7], Rodriguez formula [2,10] or was linearized by the first order Taylor expansion [2,10,11]. Here we will concentrate on the linearized model of rotation [2,10,11], which approximates rotation $\mathtt{R}_r(r_i)$ as

$$\mathtt{R}_r(r_i) = \begin{bmatrix} 1 & -r_i\omega_z & r_i\omega_y \\ r_i\omega_z & 1 & -r_i\omega_x \\ -r_i\omega_y & r_i\omega_x & 1 \end{bmatrix}. \tag{1}$$

Putting all the above together, brings us to the following RS camera model

$$\alpha_i \mathbf{u}_i = \alpha_i \left[c_i, r_i, 1\right]^\top = \mathtt{R}_r(r_i)\mathtt{R}_0\mathbf{X}_i + \mathbf{C}_0 + r_i\mathbf{t} \tag{2}$$

with $\mathtt{P}(r) = [\mathtt{R}_r(r), \mathtt{R}_0, \mathbf{C}_0, \mathbf{t}]$.

3 Bundle Adjustment with Independent RS Models

In this paper we consider Bundle Adjustment (BA) with independent RS models. This is more general than BA developed in [7] for (video) sequences of regularly spaced cameras where the camera motion during the image capture was constrained to be along the global camera trajectory. Our approach is necessary when reconstructing scenes from unorganized RS images.

Bundle adjustment [17] minimizes the sum of squares of reprojection errors which are, in our case, expressed as

$$\mathbf{e}_i^j = \tilde{\mathbf{u}}_i^j - \mu(\pi(\mathtt{P}^j(\tilde{r}_i), \mathbf{X}_i)), \tag{3}$$

where $\tilde{\mathbf{u}}_i^j = \left[\tilde{c}_i^j, \tilde{r}_i^j\right]^\top$ is the measured image point, $\mu([x, y, z]^\top) = [x/z, y/z]^\top$ is the perspective division and $\mathtt{P}^j(\tilde{r}_i)$ is an RS projection model of the j-th camera.

Non-linear least squares methods are used to find a solution $(\mathtt{P}^{j*}, \mathbf{X}_i^*)$ that (locally) minimizes the error over all the visible projections (i, j)

$$(\mathtt{P}^{j*}, \mathbf{X}_i^*) = \arg\min \sum_{(i,j)} \|\mathbf{e}_i^j\|^2.$$

When the set of images $\mathbf{u}_i^j$ is critical, it might happen that the bundle adjustment algorithm finds a local minimum producing an incorrect reconstruction. We will see that this indeed often happens.

4 Ambiguities in 3D Reconstruction with RS Camera Models

Ambiguities in 3D reconstruction with the perspective projection model have been extensively studied in [8]. It has been found there that two perspective

cameras and any number of scene points on certain ruled quadrics containing the cameras centers are in a critical configuration, as well as that for three perspective cameras, there is always a quartic curve of scene points such that they are in a critical configuration. Hence, there are situations when a set of perspective images become critical. However, the critical perspective images are very special and therefore do not in general pose problems for 3D reconstruction in practical situations with many points in generic scenes.

RS models are more general than the perspective model and therefore we expect to see more critical image sets when reconstructing with RS models. In particular, every perspective image can be explained by an RS model (2) with $\mathbf{t} = \mathbf{0}$ and $\mathtt{R}_r(r_i) = \mathtt{I}$. Therefore, every set of images that is critical for the perspective projection model is also critical for RS model (2).

RS cameras produce images with large variation of RS effects. Photographing static scenes with static RS cameras produces perspective images while images taken by RS cameras on a fast train exhibit pronounced RS effects. It is therefore desirable to look for RS SfM that can deal with all levels of RS effects. In particular, it is important that any practical RS SfM can handle perspective images.

When RS images are not truly perspective, it is often possible to treat the rolling shutter effect as (perhaps systematic) image error and explain RS images with perspective cameras, distorted scene, and somewhat higher image error. Therefore, it is important to analyze when a set of perspective images become critical w.r.t. RS model (2).

We will next show that, in many practical situations, images taken by perspective cameras become critical when reconstructed with RS model (2) and, even worse, when image noise is present, images can be explained by incorrect reconstructions with smaller error than is the smallest error of a correct reconstruction. Hence, in such situations, BA often prefers incorrect reconstructions.

We will use the RS camera model (2) with the rotation parameterized by the linearized model (1), which was used in [2,3,10,11], since it is simple to show the ambiguities algebraically with this model. The linearized rotation model is an approximation to all the other models used in the literature and therefore images that are critical w.r.t. to model (2) will be close to critical for all other models if a cameras make turns by a small angle during the image capture.. For other models, the derivations we show will not hold exactly but they will be very close for many practical situations. We have observed in experiments that BA converges to incorrect reconstructions for all RS camera models in all the cases we have tested.

4.1 Single Camera

We will start with showing how we can arbitrarily rotate the projection rays of a single RS camera and even collapse them in a single plane.

In order for a 3D point $\mathbf{X}_i$ to project into coordinate $[c_i, r_i, 1]^\top$ in the image, it has to lie on a plane defined by the row r_i and the camera center. All points that lie in such a plane can be therefore described as

$$\mathbf{X}(c, r_i, \alpha) = \left(\mathtt{R}_r(r_i)\,\mathtt{R}_0\right)^{-1}\left(\alpha\left[c, r_i, 1\right]^\top - \mathbf{C}_0 - r_i\mathbf{t}\right).$$

To obtain an equation representing the plane, we need three non-collinear points, e.g.

$$\begin{aligned}
\mathbf{X}(1, r_i, 0) &= \left(\mathtt{R}_r(r_i)\,\mathtt{R}_0\right)^{-1}\left(-\mathbf{C}_0 - r_i\mathbf{t}\right),\\
\mathbf{X}(1, r_i, 1) &= \left(\mathtt{R}_r(r_i)\,\mathtt{R}_0\right)^{-1}\left(\left[1, r_i, 1\right]^\top - \mathbf{C}_0 - r_i\mathbf{t}\right),\\
\mathbf{X}(0, r_i, 1) &= \left(\mathtt{R}_r(r_i)\,\mathtt{R}_0\right)^{-1}\left(\left[0, r_i, 1\right]^\top - \mathbf{C}_0 - r_i\mathbf{t}\right).
\end{aligned}$$

The plane $\mathbf{n}(r_i)$ determined by these three points is the solution of the following homogeneous equation system:

$$\begin{bmatrix}
\left(-\mathbf{C}_0 - r_i\mathbf{t}\right)^\top \left(\mathtt{R}_r(r_i)\,\mathtt{R}_0\right)^{-\top} & 1\\
\left(\left[1, r_i, 1\right]^\top - \mathbf{C}_0 - r_i\mathbf{t}\right)^\top \left(\mathtt{R}_r(r_i)\,\mathtt{R}_0\right)^{-\top} & 1\\
\left(\left[0, r_i, 1\right]^\top - \mathbf{C}_0 - r_i\mathbf{t}\right)^\top \left(\mathtt{R}_r(r_i)\,\mathtt{R}_0\right)^{-\top} & 1
\end{bmatrix}\mathbf{n}(r_i) = \mathtt{A}(r_i)\mathbf{n}(r_i) = \mathbf{0} \qquad (4)$$

The solution of this system always spans at least one dimensional space, which is the null-space of $\mathtt{A}(r_i)$, since the rank of $\mathtt{A}(r_i)$ is at most three.

We set $\mathbf{C}_0 = [0, 0, 0]^\top$ and $\mathtt{R}_0 = \mathtt{I}$ for simplicity and disregard the translational motion $\mathbf{t}$. We then set $\omega_y = \omega_z = 0$ to simulate the rotation around the x-axis alone. The 3D point projected on a row r_i is now written as $\mathbf{X}(c, r_i, \alpha) = \alpha\mathtt{R}_r(r_i)^{-1}\left[c, r_i, 1\right]^\top$. We again choose the triplet $\mathbf{X}(1, r_i, 0)$, $\mathbf{X}(1, r_i, 1)$ and $\mathbf{X}(0, r_i, 1)$ to determine the plane $\mathbf{n}(r_i)$, from which Eq. (4) yields

$$\begin{bmatrix}
0 & 0 & 0 & 1\\
1 & 0 & 0 & 0\\
0 & \frac{r_i(\omega_x+1)}{r_i^2\omega_x^2+1} & -\frac{r_i^2\omega_x-1}{r_i^2\omega_x^2+1} & 1
\end{bmatrix}\mathbf{n}(r_i) = 0.$$

We see that $\mathbf{n}(r_i)$ below is a solution:

$$\mathbf{n}(r_i) = \left[0, 1, \frac{r_i\,(\omega_x+1)}{r_i^2\omega_x - 1}, 0\right]^\top.$$

We can see that if we set $\omega_x = -1$ then $\mathbf{n}(r_i)$ becomes the plane $y = 0$ for any r_i. This indicates that there exists a rotational motion (linearized) making all the projected planes $\mathbf{n}(r_i)$ coplanar (see Fig. 2).

We will now extend this example to a camera whose center lies in a plane $y = 0$ and whose corresponding $\mathbf{n}(0)$ is also contained in this plane. Such a camera has $\mathbf{C} = [C_x, 0, C_z]^\top$ and can be rotated around the y-axis by angle ϕ. We will now consider the translational motion $\mathbf{t} = [t_x, t_y, t_z]^\top$ as well. The null-space of the matrix $\mathtt{A}(r_i)$ then changes to

$$\left[-\sin(\phi)(\omega_x+1), \frac{\omega_x r_i^2 - 1}{r_i}, \cos(\phi)(\omega_x+1), C_z - t_y + r_i t_z\right]^\top.$$

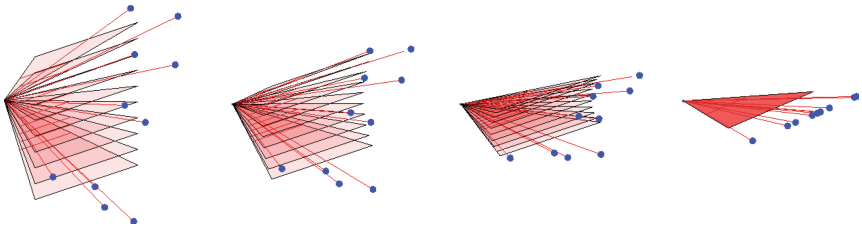

Fig. 2. Changing the rotational velocity ω_x around the x-axis for a rolling shutter camera model changes the alignment of projection rays that correspond to each image row. From left to right there is $\omega_x = 0$, $\omega_x = -0.3$, $\omega_x = -0.6$ and $\omega_x = -1$. For $\omega_x = -1$ all projection rays collapse into a single plane and any image can be explained by 3D points in a plane.

It is clear that by setting $\omega_x = -1$, $t_z = 0$ and $t_y = C_z$, we obtain again the plane $y = 0$ for any r_i. We remark that we need the non-zero t_y which is dependent on the camera position in the plane. The reason for this is that in this camera model we express the camera center in the camera coordinate system, which is changing for each r_i due to ω_x. We show that for the linearized rotation model the rotational velocity $\omega = [\omega_x, 0, 0]^\top$ can be compensated by translational velocity $\mathbf{t} = [0, C_z, 0]^\top$ to fix the camera center in the world coordinate system.

4.2 Two Cameras

Using the findings in the previous section, that arbitrary RS camera can be collapsed in a plane, we will now argue that every two images can be explained by two RS cameras and a planar scene such that the reprojection error (3) is zero.

Since each image can be explained by a camera whose center lies in plane $y = 0$ and this plane also contains all their projection rays, every two rays must intersect at least in one point. We can show this algebraically by using the equations for triangulating 3D points with known camera parameters. We can write the projection matrix parameterized by r_i as

$$\mathtt{P}^j(r_i) = \left[\mathtt{R}^j(r_i^j)\mathtt{R}_0^j\,\mathbf{C}_0^j + \mathbf{C}_r^j(r_i^j)\right] = \left[\mathbf{p}_1^j(r_i^j), \mathbf{p}_2^j(r_i^j), \mathbf{p}_3^j(r_i^j)\right]^\top.$$

Then for a 3D point corresponding to two image measurements $\tilde{\mathbf{u}}_1 = \left[\tilde{c}_i^1, \tilde{r}_i^1\right]^\top$ and $\tilde{\mathbf{u}}_2 = \left[\tilde{c}_i^2, \tilde{r}_i^2\right]^\top$ in two cameras having parameters $\mathtt{P}^1(\tilde{r}_i^1)$ and $\mathtt{P}^2(\tilde{r}_i^2)$ the following system of equations must hold with $\lambda (\in \mathbb{R} \setminus \{0\})$

$$\mathtt{M}_i\mathbf{X}_i = \begin{bmatrix} \tilde{c}_i^1\mathbf{p}_3^1(\tilde{r}_i^1)^\top - \mathbf{p}_1^1(\tilde{r}_i^1)^\top \\ \tilde{r}_i^1\mathbf{p}_3^1(\tilde{r}_i^1)^\top - \mathbf{p}_2^1(\tilde{r}_i^1)^\top \\ \tilde{c}_i^2\mathbf{p}_3^2(\tilde{r}_i^2)^\top - \mathbf{p}_1^2(\tilde{r}_i^2)^\top \\ \tilde{r}_i^2\mathbf{p}_3^2(\tilde{r}_i^2)^\top - \mathbf{p}_2^2(\tilde{r}_i^2)^\top \end{bmatrix} \begin{bmatrix} \lambda x_i \\ \lambda y_i \\ \lambda z_i \\ \lambda \end{bmatrix} = \mathbf{0}. \tag{5}$$

In order for a 3D point $[x_i, y_i, z_i]^\top$ to exist, the null-space of the 4×4 matrix $\mathtt{M}_i$ has to be at least one-dimensional, i.e., the rank must be at most 3. To calculate the triangulated 3D point coordinates we can compute the null-space. For perspective cameras in general configuration, the null-space will be either zero dimensional for non-intersecting camera rays or one dimensional, corresponding to a single 3D point.

Let us apply Eq. (5) to the above example with two RS cameras whose centers both lie in a plane $y = 0$. The rotation matrices $\mathtt{R}_0^1$ and $\mathtt{R}_0^2$ will be rotations around y axis by angles ϕ^1 and ϕ^2. Camera centers will lie anywhere in $y = 0$: $\mathbf{C}_0^1 = \left[C_x^1, 0, C_z^1\right]^\top$ and $\mathbf{C}_0^2 = \left[C_x^2, 0, C_z^2\right]^\top$. To collapse the projection rays of both cameras we will set, as shown in the previous section, the rotational velocities $\omega_x^1 = -1$ and $\omega_x^2 = -1$ and translational velocities $\mathbf{t}^1 = \left[0, t_y^1, 0\right]^\top$ and $\mathbf{t}^2 = \left[0, t_y^2, 0\right]^\top$. We then obtain the following matrix

$$\mathtt{M}_i = \begin{bmatrix} -\cos(\phi^1) - \tilde{c}_i^1 \sin(\phi^1) & -\tilde{c}_i^1 \tilde{r}_i^1 & \tilde{c}_i^1 \cos(\phi^1) - \sin(\phi^1) & C_z^1 \tilde{c}_i^1 - C_x^1 \\ 0 & -(\tilde{r}_i^1)^2 - 1 & 0 & 0 \\ -\cos(\phi^2) - \tilde{c}_i^2 \sin(\phi^2) & -\tilde{c}_i^2 \tilde{r}_i^2 & \tilde{c}_i^2 \cos(\phi^2) - \sin(\phi^2) & C_z^2 \tilde{c}_i^2 - C_x^2 \\ 0 & -(\tilde{r}_i^2)^2 - 1 & 0 & 0 \end{bmatrix}.$$

The rank of $\mathtt{M}_i$ is at most 3 and, therefore, the rays always intersect at least in one point. For any pair of image projections the null-space of $\mathtt{M}_i$ and thus the subspace where the 3D point can lie is $[a, 0, b, 1]^\top$ and therefore all points could be reconstructed in plane $y = 0$.

4.3 Projecting onto a Plane

Before we proceed to analysis of multiple RS cameras we need to explain an important fact, that is, the rotation induced by $\omega = [-1, 0, 0]^\top$ in the linearized RS model is actually a projection onto a plane $y = 0$. Let us see what happens to an arbitrary point on a camera ray. Any 3D point that lies on a camera ray can be expressed in the camera coordinate system as $\alpha\,[c_i, r_i, 1]^\top$. For the sake of simplicity we will consider $\mathtt{R}_0 = \mathtt{I}$ and $\mathbf{C}_0 = [0, 0, 0]^\top$ now. We can express the 3D point in a world coordinate system by

$$\mathbf{X} = \mathtt{R}(r_i)^{-1} \begin{bmatrix} \alpha c_i \\ \alpha r_i \\ \alpha \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{1+r_i^2} & \frac{-r_i}{1+r_i^2} \\ 0 & \frac{r_i}{1+r_i^2} & \frac{1}{1+r_i^2} \end{bmatrix} \begin{bmatrix} \alpha c_i \\ \alpha r_i \\ \alpha \end{bmatrix} = \begin{bmatrix} \alpha c_i \\ 0 \\ \alpha \end{bmatrix},$$

which shows that the x and z coordinates remain the same while the y coordinate is dropped. An illustration of this is in Fig. 3.

4.4 Multiple Cameras with Parallel y (readout) Directions

We will now use the result from previous subsection to make the following statement.

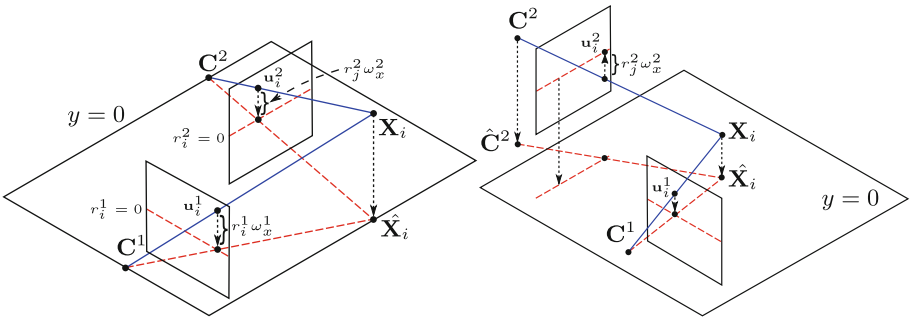

Fig. 3. (Left) Two possible configurations of a scene from image projections $\mathbf{u}_i^1$ and $\mathbf{u}_i^2$. One is represented by two perspective cameras and point $\mathbf{X}_i$ and the other by linearized RS cameras with $\omega_x^1 = \omega_x^2 = -1$ and point $\hat{\mathbf{X}}_i$. This figure illustrates that changing ω_x parameter to -1 equals to a projection into a plane $y = 0$. (Right) This projection is possible even for cameras that do not lie in the plane $y = 0$ but their readout direction is parallel.

Theorem: *Assume any number of images taken by perspective cameras with parallel y (readout) directions in space. Then, if there exists a reconstruction for such cameras using the perspective camera model, then there also exists a reconstruction using the RS camera model (2) with all cameras and 3D points lying in plane* $y = 0$.

This statement can be proven by combining the previous statements. The perspective reconstruction gives a set of 3D points $\mathbf{X}_i = [x_i, y_i, z_i]^\top$ and the cameras whose centers are $\mathbf{C}_0^j = \left[C_x^j, C_y^j, C_z^j\right]^\top$ and whose y axes are aligned with the y axis in the world coordinate system, where j is the index for cameras and i for 3D points. If we project the rays connecting $\mathbf{C}_0^j$ and $\mathbf{X}_i$ onto the plane $y = 0$ we will obtain the rays that pass through $\hat{\mathbf{C}}_0^j = \left[C_x^j, 0, C_z^j\right]^\top$ and $\hat{\mathbf{X}}_i = [x_i, 0, z_i]^\top$ which we have shown that is a configuration that is easily achieved by setting $\omega_x^j = -1$ (see Fig. 3). It follows that if there exists a perspective reconstruction for such images with zero reprojection error, the reconstruction projected to $y = 0$ will also have a zero reprojection error.

4.5 The Effect of Planar Projection in the Presence of Image Noise

The mere existence of the planar representation of the scene is not a reason BA should converge to such a solution. In practice, however, measured image points are affected by noise, and this noise leads to non-zero reprojection error $\mathbf{e}_i^j$ in BA (see Eq. (3)). In this section we show that the planar projection always reduces the reprojection error and therefore it always provides a superior solution in BA.

Suppose measured image points $\tilde{\mathbf{u}}_i^j = \left[\tilde{c}_i^j, \tilde{r}_i^j\right]^\top$ are now affected by noise such that $\mathbf{e}_i^j = \tilde{\mathbf{u}}_i^j - \mu(\mathrm{P}^j(\tilde{r}_i)\mathbf{X}_i)$. For perspective projection, i.e. $\omega^j = [0,0,0]^\top$ and $\mathbf{t}^j = [0,0,0]^\top$ the error can be expressed as

$$\mathbf{e}_i^j = \tilde{\mathbf{u}}_i^j - \mu\left(\mathtt{R}_0^j\mathbf{X}_i + \mathbf{C}^j\right) = \begin{bmatrix} \tilde{c}_i^j - \frac{C_x^j + x\cos(\phi^j) + z\sin(\phi^j)}{C_z^j + z\cos(\phi^j) - x\sin(\phi^j)} \\ \tilde{r}_i^j - \frac{y}{C_z^j + z\cos(\phi^j) - x\sin(\phi^j)} \end{bmatrix}$$

whereas the reprojection error using the linearized RS camera model with $\omega^j = \left[\omega_x^j, 0, 0\right]^\top$ and $\mathbf{t}^j = \left[0, C_z^j, 0\right]^\top$ is

$$\mathbf{e}_i^j = \begin{bmatrix} e_{ix}^j \\ e_{iy}^j \end{bmatrix} = \tilde{\mathbf{u}}_i^j - \mu\left(\mathtt{R}_r^j(\tilde{r}_i^j)\mathtt{R}_0^j\mathbf{X}_i + \mathbf{C}^j + \tilde{r}_i^j\mathbf{t}^j\right) = \begin{bmatrix} \tilde{c}_i^j - \frac{C_x^j + x\cos(\phi^j) + z\sin(\phi^j)}{C_z^j + z\cos(\phi^j) - x\sin(\phi^j)} \\ 0 \end{bmatrix}.$$

The e_{iy}^j component of the reprojection error is eliminated and the e_{ix}^j component remains unchanged by the projection to $y = 0$; therefore the overall error is reduced. This is always true for images taken by the perspective cameras with identical y directions in space.

4.6 What Does It Mean in Practice?

We have shown the reason why the planar projection reduces the reprojection error in the case where all images are captured by perspective cameras with identical y direction in space. This case is in practice hardly achieved exactly, but we can often come very close to this scenario, for example when taking handheld pictures while walking or taking pictures with a camera mounted on a car.

When images are captured with the y directions not parallel, we are still able to reduce e_{iy}^j to zero, but at the cost of increasing the e_{ix}^j component. It follows that BA will try to reduce e_{iy}^j as far as the increase in e_{ix}^j does not exceed the reduction in e_{iy}^j.

The amount of increase in e_{ix}^j depends on camera poses when images are taken and it is complicated to analyze in general. We have, however, practically observed the following fact.

Observation: *For three or more images by perspective cameras with pairwisely different y directions, the deformation of the scene by BA due to using the RS model is directly dependent on the angle between the y axes.*

In synthetic experiments, we show that when the smallest angle between the three pairs of y directions is at least 30 degrees, the reconstruction is recovered correctly. I real experiments, on the other hand, we show that capturing the scene with sufficient amount of images with two distinct y directions that are perpendicular with each other, i.e. taking portrait as well as landscape images provides a correct reconstruction.

5 Experiments

5.1 Synthetic Experiments

In Sect. 4 we have shown that images captured with parallel readout directions used in BA with linearized RS camera model can be explained by a planar scene

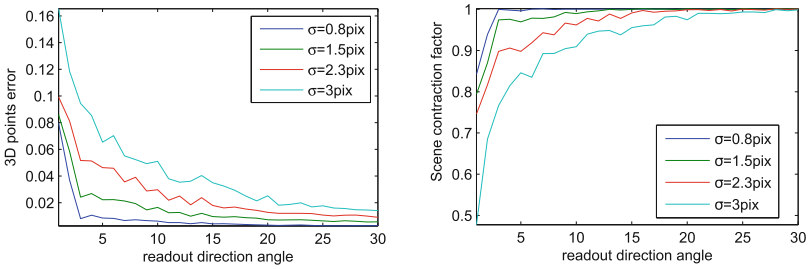

Fig. 4. Experiment with three randomly initialized cameras. The x axis shows the minimal readout direction angle among the three cameras. The figure on the left shows the mean spatial error over all 3D points after the optimization and the figure on the left shows the contraction factor of the scene compared to the ground truth. The lower the contraction factor the more deformation is in the scene. Results are shown for several values of error in the observations, expressed by the variance σ of their zero mean normal distribution.

and that this configuration has lower reprojection error. In synthetic experiments we verified this also for SLERP and Rodriguez parameterization.

Further investigation was aimed at the case when image readout directions were not parallel during capture. We studied the amount of minimal angular difference between the three readout directions needed for the scene to be reconstructed correctly. To express the "correctness" of the reconstruction we introduce a measure which we call the scene contraction factor.

We calculate scene contraction factor as the ratio between the third principal component of the 3D points' coordinates before and after BA. The optimized 3D points after BA are first fitted to the initial 3D points by a similarity transform and then the principal components are calculated. If the scene is deformed, the third principal component will be different. A correctly reconstructed scene will have contraction factor close to 1 whereas completely flat scene will have contraction factor equal to 0.

We sampled three cameras randomly on a sphere with radius of 1 and pointing towards the a cubical scene. We measured the mean distance of initial 3D points from the resulting ones and also the scene contraction factor. Altogether 10,000 samples were generated and we categorized them based on the minimal angle between the three pairs of readout directions.

For each of these samples the same analysis as in previous experiment was done using 1000 different initializations with increasing image noise. We show the results for several values of the image noise in Fig. 4. From these experiments we can predict that if the minimal readout direction angle among the three camera pairs is at least 30 degrees, the reconstruction should be correct.

5.2 Real Data Experiments

To test our hypotheses under real conditions we captured several datasets using smart-phone camera under various angles. In order to have three mutually

distinct RS readout directions we captured the same scene in vertical, horizontal and tilted position of the phone. Images were extracted from short videos captured handheld while moving around the objects or walking.

An incremental SfM pipeline similar to [16,18] was used to provide a baseline reconstruction. This pipeline was then adapted to use R6P [3] solver for absolute camera pose computation and used either linearized camera rotation model, SLERP or Rodriguez rotation model in bundle adjustment. Since we observed identical behavior for all rotation models, we present only the results of the linearized one. This rolling shutter aware version of the pipeline is denoted in the experiments as RS and the original as global shutter (GS) camera, which is equivalent with the perspective camera.

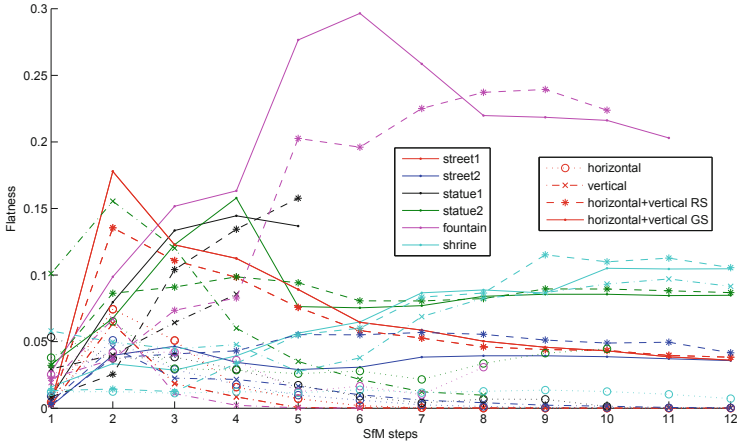

Fig. 5. Analysis of the criticality in real datasets. The degeneracy is shown as flatness of the scene, where zero means completely flat. For either horizontal or vertical datasets, the degeneracy is apparent as the scene usually collapses to a plane completely. The results of the RS pipeline on datasets with both horizontal and vertical images show the same scene dimensions as the ones with GS pipeline.

For each dataset we ran the two pipelines on several subsets of data – horizontal images, vertical images, horizontal+vertical and horizontal+vertical+tilted. According to our expectations, for the subsets containing only one readout direction the scene was collapsing to a plane as the RS incremental pipeline was progressing. We have calculated the flatness of the scene using principal component analysis (Fig. 5). Note that flatness in Fig. 5 is not the same as scene contraction factor used in synthetic experiments, value 0 still means the scene is completely flat, but the maximum is different for each dataset.

It is important to realize that in practice only few iterations of BA are allowed for the sake of performance and therefore the scene collapses gradually as new cameras are added and BA step is repeated. For small datasets with only small number of BA steps (up to 20 cameras) the deformation was not so apparent but

Fig. 6. Reconstructions using SfM pipeline for unorganized RS images. (Left) Horizontal image set sample and its reconstruction below. (Middle) Vertical image set sample and its reconstruction next to it. (Right) Reconstruction from both horizontal and vertical images. Notice the deformations when only one image direction is used. Two perpendicular directions provide correct results.

it was extremely critical in larger datasets. When three distinct image readout directions were present in the dataset, we did not notice any deformation of the model caused by the RS pipeline compared to the GS pipeline, which confirms our predictions.

Even more important, however, are the experiments with two distinct readout directions (horizontal+vertical), which also do not show any deformation compared to the baseline GS reconstruction. This shows that in practice having horizontal as well as vertical images of the scene should be sufficient to successfully reconstruct the scene using RS pipeline. We show the results in (a rather complex) Fig. 6.

6 Conclusion

We tackled the topic of SfM with RS cameras. Recent works have shown that accounting for the camera movement in RS images can greatly improve the result and presented several practical RS camera models. We show that such models when used without constraints on the camera motion lead to incorrect reconstructions.

We analyzed the cases in which incorrect reconstruction arises and the reasons why it is so. We prove that any two perspective images can be explained by the linearized RS camera model and a planar scene. Further we prove that a set of images taken with parallel readout directions that can be explained by perspective cameras can also be explained by RS cameras and a scene all lying in a single plane. Moreover, we prove that the reprojection error is always reduced in such a case and, therefore, BA tends to prefer such solution.

This is a consequence of the linearized rotation being a mere projection on a plane. Since the linearized rotation model is a close approximation to all the other models it is expected that the other models will exert similar effects in BA. We have observed this both in synthetic and real data.

We show that in order to obtain a correct reconstruction using unconstrained RS SfM pipeline the input images should be captured with different readout directions. Synthetic experiments suggest that for 3 or more cameras, the minimal mutual angle between the readout directions should be at least 30 degrees. The experiments on real data confirm our predictions and in addition show that having two image sets with perpendicular readout directions is enough to obtain a correct reconstruction using SfM pipeline with the RS camera model.

Acknowledgment. This research was in part supported by Czech Ministry of Education under Project RVO13000, by Grant Agency of the CTU Prague project SGS16/230/OHK3/3T/13 and by Grant-in-Aid for Scientific Research of the Ministry of Education, Culture, Sports, Science and Technology of Japan.

References

1. Ait-Aider, O., Berry, F.: Structure and kinematics triangulation with a rolling shutter stereo rig. In: IEEE 12th International Conference on Computer Vision, pp. 1835–1840, September 2009

2. Ait-aider, O., Andreff, N., Lavest, J.M., Blaise, U., Ferr, P.C., Cnrs, L.U.: Simultaneous object pose and velocity computation using a single view from a rolling shutter camera. In: Proceedings of the European Conference on Computer Vision, pp. 56–68 (2006)
3. Albl, C., Kukelova, Z., Pajdla, T.: R6p - rolling shutter absolute pose problem. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2292–2300, June 2015
4. Hartley, R., Kahl, F.: Critical configurations for projective reconstruction from multiple views. IJCV **71**, 5–47 (2006)
5. Hartley, R., Zisserman, A.: Multiple View Geometry in Computer Vision. Cambridge University Press, New York (2003)
6. Hedborg, J., Ringaby, E., Forssen, P.E., Felsberg, M.: Structure and motion estimation from rolling shutter video. In: 2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops), pp. 17–23 (2011)
7. Hedborg, J., Forssèn, P.E., Felsberg, M., Ringaby, E.: Rolling shutter bundle adjustment. In: CVPR, pp. 1434–1441 (2012)
8. Kahl, F., Hartley, R.: Critical curves and surfaces for euclidean reconstruction. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002. LNCS, vol. 2351, pp. 447–462. Springer, Heidelberg (2002). doi:10.1007/3-540-47967-8_30
9. Klein, G., Murray, D.: Parallel tracking and mapping on a camera phone. In: 8th IEEE International Symposium on Mixed and Augmented Reality, ISMAR 2009, pp. 83–86, October 2009
10. Magerand, L., Bartoli, A., Ait-Aider, O., Pizarro, D.: Global optimization of object pose and motion from a single rolling shutter image with automatic 2D-3D matching. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012. LNCS, vol. 7572, pp. 456–469. Springer, Heidelberg (2012). doi:10.1007/978-3-642-33718-5_33
11. Meingast, M., Geyer, C., Sastry, S.: Geometric models of rolling-shutter cameras. Comput. Res. Repository (2005)
12. Moulon, P., Monasse, P., Marlet, R.: Global fusion of relative motions for robust, accurate and scalable structure from motion. In: 2013 IEEE International Conference on Computer Vision (ICCV), pp. 3248–3255, December 2013
13. Oth, L., Furgale, P., Kneip, L., Siegwart, R.: Rolling shutter camera calibration. In: 2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1360–1367, June 2013
14. Saurer, O., Koser, K., Bouguet, J.Y., Pollefeys, M.: Rolling shutter stereo. In: 2013 IEEE International Conference on Computer Vision (ICCV), pp. 465–472, December 2013
15. Saurer, O., Pollefeys, M., Lee, G.H.: A minimal solution to the rolling shutter pose estimation problem. In: 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 1328–1334, September 2015
16. Snavely, N., Seitz, S.M., Szeliski, R.: Photo tourism: exploring photo collections in 3d. In: ACM SIGGRAPH 2006 Papers, pp. 835–846. ACM, New York (2006)
17. Triggs, B., McLauchlan, P.F., Hartley, R.I., Fitzgibbon, A.W.: Bundle adjustment — a modern synthesis. In: Triggs, B., Zisserman, A., Szeliski, R. (eds.) IWVA 1999. LNCS, vol. 1883, pp. 298–372. Springer, Heidelberg (2000). doi:10.1007/3-540-44480-7_21
18. Wu, C.: VisualSFM: A Visual Structure from Motion System (2011)