Supplementary Material: Image Stitching and Rectification for Hand-Held Cameras

Bingbing Zhuang and Quoc-Huy Tran

NEC Labs America

This supplementary material contains additional contents that are not discussed in the main paper due to space limit. Specifically, it includes: derivation of GS differential homography, derivation of RS differential homography that takes into account the varying plane parameters due to the intra-frame motion, degeneracy discussion, additional experimental results, and two demo videos for image rectification.

S1 Additional Details

S1.1 Derivation of GS Differential Homography

Here, we provide a brief derivation of motion flow in terms of GS differential homography as defined in Eq. 3 of the main paper. First, suppose the camera moves with the instantaneous velocity $(\boldsymbol{\omega}, \boldsymbol{v})$, a 3D point $\boldsymbol{X} = [X, Y, Z]^\top$ in general position is observed by the camera to move with the velocity $\boldsymbol{V} = -\boldsymbol{v} - \lfloor \boldsymbol{\omega} \rfloor_\times \boldsymbol{X}$. Projecting this 3D velocity into the 2D image plane yields the motion flow $\boldsymbol{u} = [u_x, u_y]^\top$ at (x, y) [1]

$$u_x = \frac{v_z x - v_x}{Z} - \omega_y + \omega_z y + \omega_x xy - \omega_y x^2, \tag{S1}$$

$$u_y = \frac{v_z y - v_y}{Z} + \omega_x - \omega_z x - \omega_y xy + \omega_x y^2. \tag{S2}$$

Since the 3D points $\boldsymbol{X}$ now lie in a plane, i.e.

$$\boldsymbol{n}^\top \boldsymbol{X} = d, \tag{S3}$$

and from perspective projection $\boldsymbol{X} = Z[x, y, 1]^\top$ we have

$$\frac{1}{Z} = \frac{\boldsymbol{n}^\top \hat{\boldsymbol{x}}}{d}, \tag{S4}$$

where $\hat{\boldsymbol{x}} = [x, y, 1]^\top$. Plugging Eq. S4 into Eq. S2, it is trivial to verify that it becomes equivalent to the Eq. 3 in the main paper.

S1.2 Plane Parameters

Here, we derive the differential homography that takes into account the varying plane parameters due to the intra-frame motion. For convenience, we will again adopt the instantaneous motion model. Specifically, the instantaneous camera velocity $(\boldsymbol{\omega}, \boldsymbol{v})$ would induce an instantaneous rotation $\exp(-\lfloor\boldsymbol{\omega}\rfloor_\times)$ to the plane normal $\boldsymbol{n}$ and an instantaneous change $-\boldsymbol{v}^\top\boldsymbol{n}$ to the camera-plane distance d. When $(\boldsymbol{\omega}, \boldsymbol{v})$ is applied to describe the motion between the two first scanlines of the image pair (Sec. 4.1 in the main paper), we can obtain the plane parameters w.r.t. the scanline y_1 in frame 1 as $(\exp(-\beta_1(k, y_1)\lfloor\boldsymbol{\omega}\rfloor_\times)\boldsymbol{n}, d - \beta_1(k, y_1)\boldsymbol{v}^\top\boldsymbol{n})$. Combined with Eq. 9 and 10 in the main paper, we can write the homography between two scanlines y_1 and y_2 as

$$\boldsymbol{H}_{y_1y_2} = -\beta(k, y_1, y_2)\boldsymbol{K}\left(\lfloor\boldsymbol{\omega}\rfloor_\times + \frac{\boldsymbol{v}\left(\exp(-\beta_1(k, y_1)\lfloor\boldsymbol{\omega}\rfloor_\times)\boldsymbol{n}\right)^\top}{d - \beta_1(k, y_1)\boldsymbol{v}^\top\boldsymbol{n}}\right)\boldsymbol{K}^{-1}. \tag{S5}$$

For clarity, we shall omit the k, y_1 and y_2 for the derivation below. According to the first-order Taylor expansion, we have

$$\exp(-\beta_1\lfloor\boldsymbol{\omega}\rfloor_\times) \approx \boldsymbol{I} - \beta_1\lfloor\boldsymbol{\omega}\rfloor_\times, \tag{S6}$$

$$\frac{1}{d - \beta_1\boldsymbol{v}^\top\boldsymbol{n}} \approx \frac{1}{d} + \frac{\beta_1\boldsymbol{v}^\top\boldsymbol{n}}{d^2}. \tag{S7}$$

Hence, we have

$$\frac{\boldsymbol{v}\left(\exp(-\beta_1\lfloor\boldsymbol{\omega}\rfloor_\times)\boldsymbol{n}\right)^\top}{d - \beta_1\boldsymbol{v}^\top\boldsymbol{n}} = \frac{\boldsymbol{v}\boldsymbol{n}^\top}{d} - \frac{\beta_1\boldsymbol{v}(\lfloor\boldsymbol{\omega}\rfloor_\times\boldsymbol{n})^\top}{d} + \tag{S8}$$

$$+ \frac{\left(\beta_1\boldsymbol{v}^\top\boldsymbol{n}\right)\boldsymbol{v}\boldsymbol{n}^\top}{d^2} - \frac{\left(\beta_1^2\boldsymbol{v}^\top\boldsymbol{n}\right)\boldsymbol{v}(\lfloor\boldsymbol{\omega}\rfloor_\times\boldsymbol{n})^\top}{d^2}. \tag{S9}$$

Here, we can keep only the first-order term $\frac{\boldsymbol{v}\boldsymbol{n}^\top}{d}$ and ignore other higher-order terms (in terms of $\boldsymbol{\omega}$ and $\boldsymbol{v}$) that have relatively small impact. We then have

$$\boldsymbol{H}_{y_1y_2} = -\beta(k, y_1, y_2)\boldsymbol{K}\left(\lfloor\boldsymbol{\omega}\rfloor_\times + \frac{\boldsymbol{v}\boldsymbol{n}^\top}{d}\right)\boldsymbol{K}^{-1}. \tag{S10}$$

This is the differential RS-aware homography we use in the main paper; it allows us to derive the corresponding 5-point solver with only slightly higher complexity than its GS counterpart.

S1.3 Degeneracy

Here, we would like to discuss in more details on whether there are different pairs of k and $\boldsymbol{H}$ that lead to the same motion flow as defined in Eq. 11 of the main paper. Note that our interest is not in the well-known degeneracy [5, 4] that $\boldsymbol{H}$ itself can arise from two valid pairs of motion and 3D plane, but rather

in whether the RS effect induces additional degeneracy. Let us suppose $\{k_1, \boldsymbol{H}_1\}$ yields the same motion flow as does $\{k_2, \boldsymbol{H}_2\}$, i.e.,

$$\beta(k_1, y_1, y_2)(\boldsymbol{I} - \hat{\boldsymbol{x}}\boldsymbol{e}_3^\top)\boldsymbol{H}_1\hat{\boldsymbol{x}} = \beta(k_2, y_1, y_2)(\boldsymbol{I} - \hat{\boldsymbol{x}}\boldsymbol{e}_3^\top)\boldsymbol{H}_2\hat{\boldsymbol{x}}. \tag{S11}$$

for all the $\boldsymbol{x}$ in the image plane. Recall that $\beta(k, y_1, y_2) = (1 + \frac{\gamma(y_2 - y_1)}{h} + \frac{1}{2}k((1 + \frac{\gamma y_2}{h})^2 - (\frac{\gamma y_1}{h})^2))(\frac{2}{2+k})$. For a pair of $\boldsymbol{H}_1$ and $\boldsymbol{H}_2$, one can see that collecting the constraint Eq. S11 from many points $\boldsymbol{x}$ quickly leads to an over-constrained system on k_1 and k_2. The equations cannot be all satisfied in general except under special configurations of motion and 3D plane. Here, we attempt to identify such configurations yet do not find any practical situations under which the degeneracy exists. Considering that k_1 and k_2 are just two independent scales, we reckon that such degeneracy, if existing at all, is very rare.

S2 Additional Results

S2.1 Two-View Experiments

First, we present in Fig. S1 the full results that we did not give in Fig. 6 of the main paper due to space constraints. We observe that 'Adobe AE' has clearly inferior rectification results compared to both 'DiffSfM' and 'RS-APAP & Rectification'. Second, we give additional two-view qualitative results in Fig. S2 and S3. In the example shown in Fig. S2(a), we observe that both 'DiffSfM' and 'Adobe AE' introduces undesirable distortions in the image boundary. We reckon that the distortions in 'Adobe AE' are perhaps due to its attempt to perform inpainting-like tasks in the boundary. However, it affects those pixels within the field of view as well. We have more examples later to demonstrate this observation. In the example shown in Fig. S2(b), we would like to highlight that, although all the three rectification methods are able to rectify the building to be mostly upright and it is hard to tell which performs best in this case, there are evident performance gap in the image stitching as highlighted. Especially, 'DiffSfM' brings significant hindrance to the stitching (i.e., 'DiffSfM+APAP' or [9]+[8]). This is presumably because of the discrepancy in the motion models — the rectification applies the differential motion model while the stitching adopts the discrete one. Our 'RS-APAP' obtains consistent results in rectification and stitching, thanks to its holistic nature. In the example shown in Fig. S3(a), we again observe that 'DiffSfM' introduces undesirable distortions in the image boundary. The images returned by 'Adobe AE' remain somewhat distorted; see the shape of the doors as highlighted. The superiority of 'RS' and 'RS-APAP' in image stitching is also clear. Example in Fig. S3(b) presents a challenging case, wherein the images contain large texture-less regions, i.e., windows, as well as slight motion blurs in frame 1. As expected, 'DiffSfM' causes chaotic perturbation to these regions due to the difficulty in establishing correspondences. 'Adobe AE' rectifies the images slightly, but again, produces large distortion in the boundary. 'RS-APAP & Rectification' rectifies the building to be upright, without introducing noticeable artifacts.

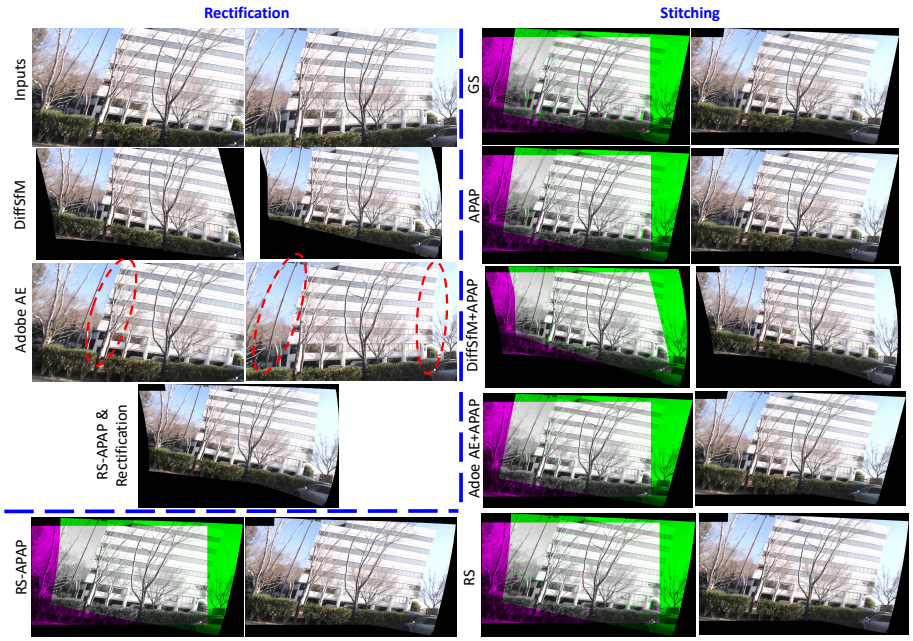

Fig. S1. Comparison of rectification/stitching on real RS images. Best viewed in screen.

S2.2 Test on Data from [9]

We give in Fig. S4 the rectification results on the remaining four images from [9], which we did not provide in the main paper due to space limit.

S2.3 Sensitivity to Keypoints Distribution

As mentioned in the main paper, we observe that APAP might become unstable in cases of keypoints concentrating in a small region. Note that this is independent of the RS effect, and poses challenges to all keypoint-based methods. Nevertheless, we observe that our differential homography mitigates this issue to some extent as compared to the discrete homography. This may well be due to the higher stability of the differential motion model under small motion, which is exactly the motivation of its use in classical SfM [3, 4]. An example is shown in Fig. S5. As highlighted by the red circle, the APAP results might contain non-smooth artifacts near the small region where crowded keypoints are closely located in. This concentration of keypoints makes the local discrete homography estimation unstable, especially so with smaller value of σ. This further implies the hardness for APAP in such case to select a good value of σ that can compensate for local deformation (smaller σ desired) while leading to stable results. Our differential homography is more stable against the choice of σ.

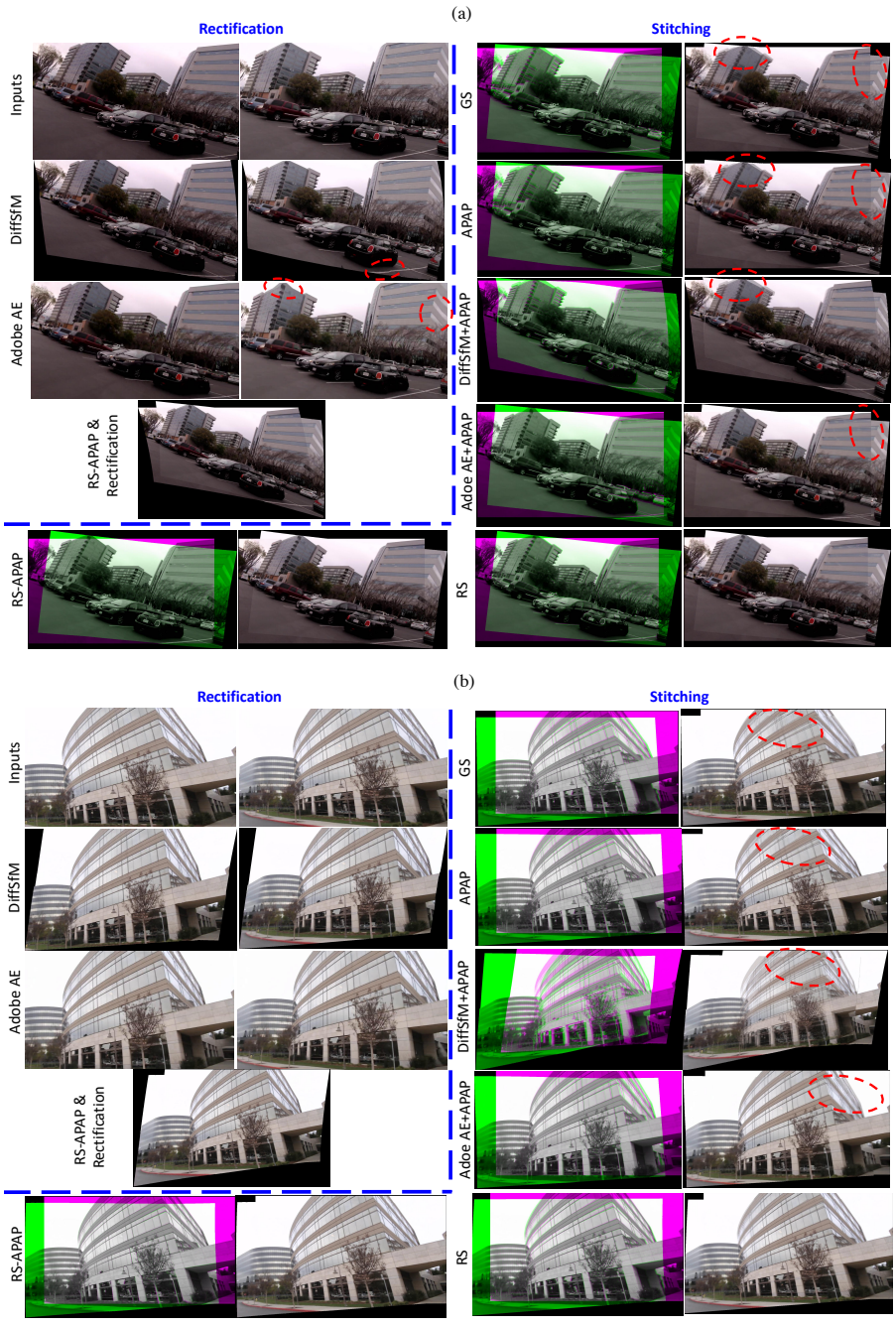

Fig. S2. Comparison of rectification/stitching on real RS images. Best viewed in screen.

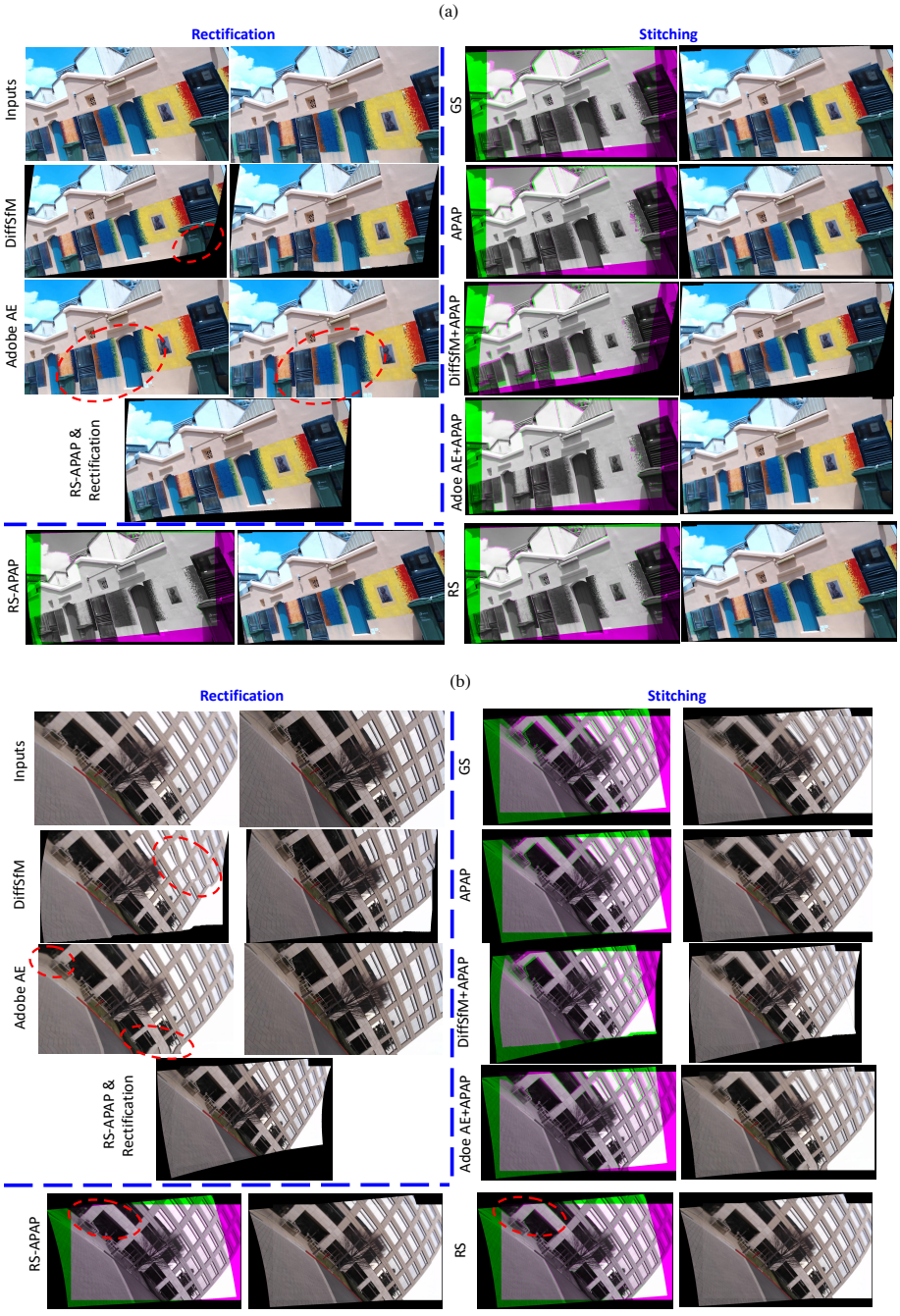

Fig. S3. Comparison of rectification/stitching on real RS images. Best viewed in screen.

Fig. S4. Qualitative comparison to DiffSfM [9], the method of Lao and Ait-Aider [2], and the method of Vasu et al. [7]. Stitched images with rectification are shown for [2] and ours. [7]'s results for the last two images are not provided by the authors.

S2.4 Demo Videos

We include two videos in this supplementary material to demonstrate the effectiveness of our image rectification method. The videos are captured by a hand-held RS camera while the holder is running, and hence contain large camera shaking, leading to significant image distortions. We compare our method with 'DiffSfM' and 'Adobe AE'. We refer the readers to the videos for the full results, and extract two frames from the videos as examples to show in Fig. S6. As can be seen in Fig. S6(a), 'DiffSfM' rectifies some parts of the scene while inducing evident distortions due to its sensitivity to errors in optical flow estimates. As visualized in Fig. S7, it is obvious that the circled region contains unsmooth and wrong flow estimates, which inevitably affect the image rectification in 'DiffSfM'. We can also see that 'Adobe AE' strongly bends the straight lines near the boundary. In the example of Fig. S6(b), 'DiffSfM' again ruins some parts of the image, while 'Adobe AE' fails to rectify the building as compared to our method.

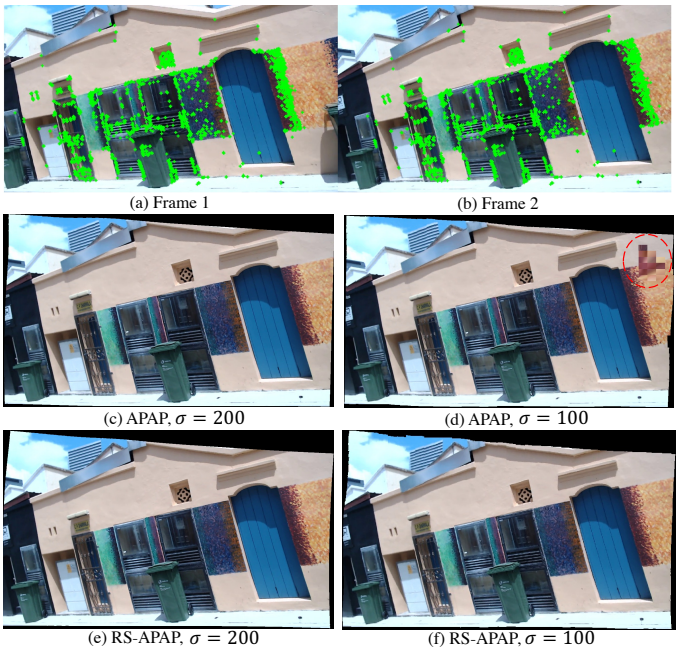

Fig. S5. An example to demonstrate the sensitivity to keypoints distribution. (a)(b) show the original image pair with keypoints. (c)-(f) show the results of warping frame 2 to the canvas of frame 1 using APAP or RS-APAP with different values of σ.

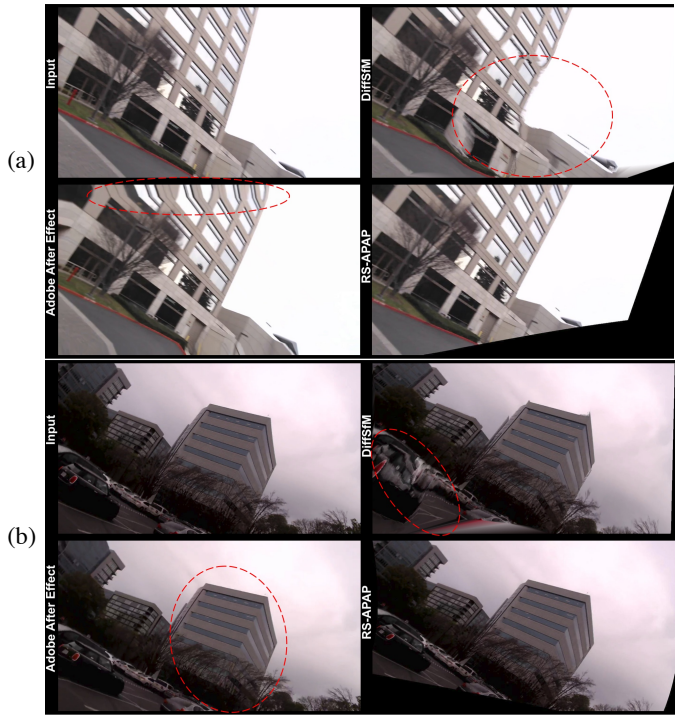

Fig. S6. Two examples extracted from the demo videos for image rectification.

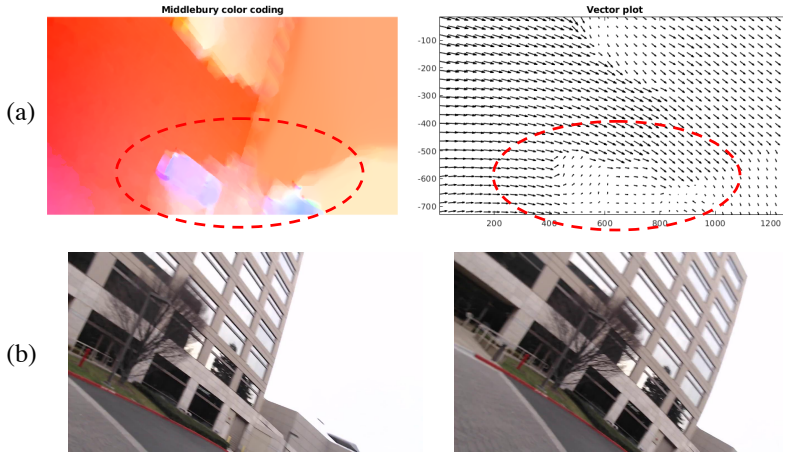

Fig. S7. (a) The optical flow results (visualized using [6]) for the example in Fig. S6(a). (b) The input image in Fig. S6(a) and its next frame used to compute optical flow.

References

1. Horn, B.K.: Motion fields are hardly ever ambiguous. International Journal of Computer Vision **1**(3), 259–274 (1988)
2. Lao, Y., Aider, O.A.: Rolling shutter homography and its applications. In: IEEE Transactions on Pattern Analysis and Machine Intelligence (2020)
3. Ma, Y., Košecká, J., Sastry, S.: Linear differential algorithm for motion recovery: A geometric approach. International Journal of Computer Vision **36**(1), 71–89 (2000)
4. Ma, Y., Soatto, S., Kosecka, J., Sastry, S.S.: An invitation to 3-d vision: from images to geometric models, vol. 26. Springer Science & Business Media (2012)
5. Maybank, S.: Theory of reconstruction from image motion, vol. 28. Springer Science & Business Media (2012)
6. Sun, D., Roth, S., Black, M.J.: Secrets of optical flow estimation and their principles. In: CVPR. IEEE (2010)
7. Vasu, S., Mohan, M.M., Rajagopalan, A.: Occlusion-aware rolling shutter rectification of 3d scenes. In: CVPR (2018)
8. Zaragoza, J., Chin, T.J., Brown, M.S., Suter, D.: As-projective-as-possible image stitching with moving dlt. In: CVPR (2013)
9. Zhuang, B., Cheong, L.F., Hee Lee, G.: Rolling-shutter-aware differential sfm and image rectification. In: ICCV (2017)