ROBUST 2D AUGMENTED REALITY BASED ON HOMOGRAPHY REFINEMENT AND TEMPORAL COHERENCE

Kim, Karam[1], Joongseok, Song[1], Park, Hanhoon[2], and Park, Jong-Il[1]*

[1] *Hanyang University, 222 Wangsimni-ro, Seongdong-gu, Seoul 133-791, Korea*
[2] *Pukyong National University, 45 Yongso-ro, Nam-gu, Busan 608-737, Korea*
grkim@mr.hanyang.ac.kr, jssong@mr.hanyang.ac.kr, hanhoon_park@pknu.ac.kr, jipark@hanyang.ac.kr

ABSTRACT

To augment real planes with virtual objects, the camera pose relative to the planes must be estimated, which can be done by plane-based camera tracking. In plane-based camera tracking, robustness and real-time capability are two critical factors but they are conflicting. To resolve the problem, we first employ a homography refinement method. Then, we use the temporal coherence of consecutive frames to eliminate the computational overhead caused by iterative refinements. It means that the camera pose estimated in the preceding frame is used as an initial camera pose of the current frame. Use of temporal coherence allows us to use more efficient feature tracking methods and thus we can achieve additional reduction in processing time. Experimental results demonstrate that 50% of the processing time can be saved while maintaining the robustness.

KEYWORDS

Plane-based camera tracking, feature tracking, temporal coherence, homography refinement, augmented reality

1 INTRODUCTION

By tracking a real plane in camera images, 3D motions of the camera can be estimated. Such a method is commonly said to be plane-based camera tracking. There have been a number of plane-based camera tracking methods. Most of them are based on 2D feature matching between camera images and a reference image including a real plane.

2D feature matching results in a matrix, called homography. However, in real environments, it is usually impossible to definitely define the relationship between 2D features and their correspondences with a homography. This problem causes jitter in camera poses. Thus, it would be a good solution to refine the homography in a recursive manner. This paper proposes such a method. First, an initial homography is computed by feature matching between the current frame and a reference image. Then, the current image is warped using the initial homography or the homography computed in the previous step. Next, features are matched between the warped image and the reference image. Then, the initial homography or the homography computed in the previous step is refined from the feature matching. These processes are repeated.

Unfortunately, this recursive method greatly increases the computational complexity. To resolve this problem, we use the temporal coherence between consecutive frames. That is, in our homography refinement method, the initial homography is not estimated but given by the homography of the preceding frame. This also reduces the number of iterations because the homography of the preceding frame is already almost same as that of the current frame.

To further reduce the jitter, features must be reliably matched with their true correspondences. For reliable feature matching, it must be preceded to extract features and to describe them robustly to scale, rotation, and illumination changes. A number of related researches have been carried out. SIFT[1], SURF[2], and BRISK[3] are the representative ones. However, they all suffer from high computational complexity due to the additional processes (such as dominant orientation neutralization, image pyramid construction, etc.) for achieving the invariance to scale, rotation, and illumination changes. Even BRISK cannot work in real-time in mobile environments. Therefore, for real-time capability, it would be desirable not to have the additional processes for achieving the invariances.

For the purpose, use of the temporal coherence is

*Corresponding author.

beneficial again. In our homography refinement method, the warped image and the reference image are very similar. Therefore, scale-invariant and rotation-invariant descriptors and detectors are not necessary. Therefore, the most efficient one, the combination of FAST[6] and BRIEF[4], can be utilized.

2 ROBUST PLANE TRACKING BASED ON HOMOGRAPHY REFINEMENT AND TEMPORAL COHERENCE

If a camera does not move drastically, there is little difference between consecutive frames. This is usually called temporal coherence. Based on this property, we can robustly track a 2D plane in camera images in such a way of recursively refining an infinitesimal homography between consecutive frames. The plane motion is directly related to the camera pose [5].

First, an initial homography is obtained from the homography estimated in the preceding frame. Then, the current image is warped using the initial homography or the homography computed in the previous step. Next, features are detected/matched in/between the warped image and the reference image using BRIEF. Then, the initial homography or the homography computed in the previous step is refined from the feature matching. These processes are repeated as

$$\mathbf{H(t) = \Delta H H(t\text{-}1).} \quad (1)$$

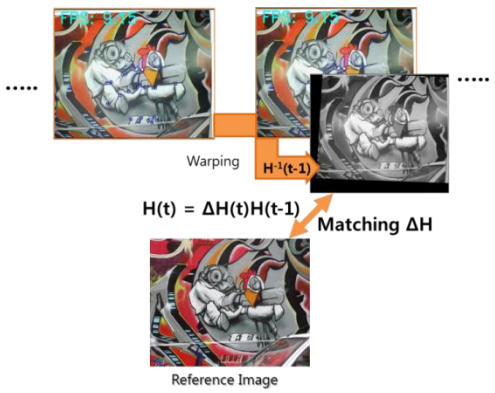

Figure 1. Recursive homography refinement. The current frame is warped using the previously-computed homography **H**(t-1). Then the differential homography Δ**H**(t) is computed between the warped image and the reference image through feature matching. Finally, it is multiplied by **H**(t-1).

From the finally refined homography in each frame, the camera pose can be estimated as follows. The homography can be decomposed as

$$\mathbf{H} \alpha \mathbf{A}[\mathbf{R}_1\ \mathbf{R}_2\ \mathbf{T}]. \quad (2)$$

Here, $\mathbf{R}_1$, $\mathbf{R}_2$, and $\mathbf{T}$ are rotation vectors and a translation vector. The matrix A is a camera internal matrix, which is computed from camera calibration. Assuming $\mathbf{G} = \mathbf{A}^{-1}\mathbf{H}$ in Eq. (2), the camera pose matrix is derived as

$$l = \sqrt{||\mathbf{G_1}||\,||\mathbf{G_2}||}, \mathbf{R_1} = \frac{\mathbf{G_1}}{l}, \mathbf{R_2} = \frac{\mathbf{G_2}}{l}, \mathbf{T} = \frac{\mathbf{G_3}}{l} \quad (3)$$

and when $\mathbf{c} = \mathbf{R_1} + \mathbf{R_2}, \mathbf{p} = \mathbf{R_1} \times \mathbf{R_2}$, $\mathbf{d} = \mathbf{c} \times \mathbf{p}$,

$$\mathbf{R_3} = \mathbf{R'_1} \times \mathbf{R'_2} \quad (4)$$

where $\mathbf{R_1}' = \frac{1}{\sqrt{2}}\left(\frac{\mathbf{c}}{||\mathbf{c}||} + \frac{\mathbf{d}}{||\mathbf{d}||}\right), \mathbf{R'_2} = \frac{1}{\sqrt{2}}\left(\frac{\mathbf{c}}{||\mathbf{c}||} - \frac{\mathbf{d}}{||\mathbf{d}||}\right)$.

3 EXPERIMENTAL RESULTS

For experiments, a known plane (having a variety of patterns) was fixed on a given location or moved along predefined displacements and angles using a pulley. Then, image sequences were taken from a web camera (Logitech HD Pro C910) at 640 × 480 resolution. Therefore, the ground truth for camera motions is given. On the image sequences, feature points were detected by the FAST detector and were described by the BRIEF descriptor.

To show that general plane based camera tracking methods are not robust to jitter, while fixing the camera and the planar pattern, the amount of jitter were measured and compared between an existing method (using SURF[2] and having neither homography refinement nor use of temporal coherence) and the proposed one.

Table 1. Processing time and jitter reduction rate of the proposed camera tracking method

Processing time (ms/frame)		Jittering reduction rate by the proposed method (%)
Existing method	Proposed method	
114.30	29.86	32.43

As shown in Table 1, the proposed method could reduce about 32% of the jitter caused by SURF. In addition, Since SURF performs scale invariant and rotation invariant feature points tracking, the processing time was about 4 times longer than the proposed method.

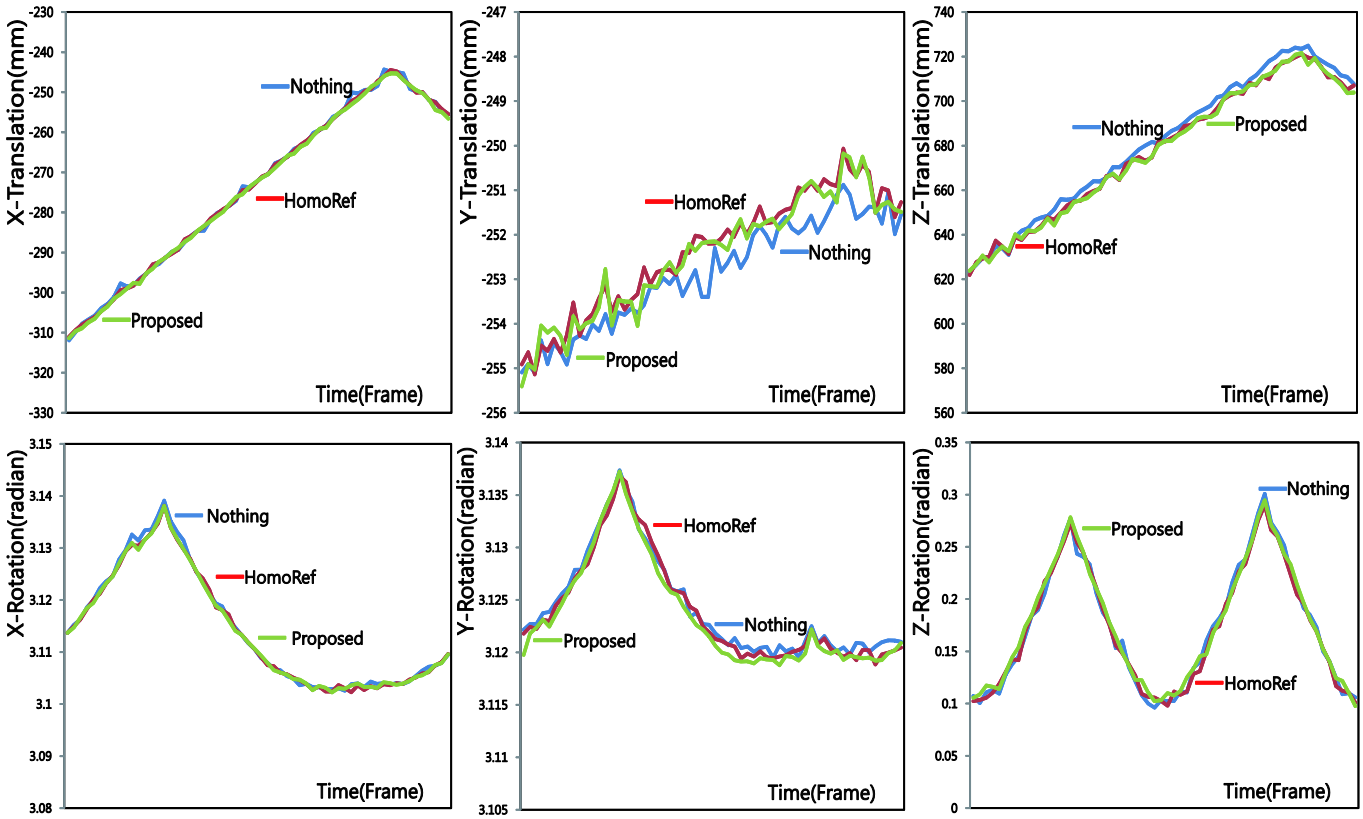

Figure 2. Comparison of camera pose results by three methods (*Nothing*, *HomoRef*, and *Proposed*). (a), (b), and (c): translations along the a x-axis, y-axis and z-axis, (d), (e), and (f): rotations for the x-axis, y-axis, and z-axis.

For the image sequences where the camera was fixed and the planar patterns were moved along each axis, three methods of Table 2 were compared. Figure 2 and Table 3 show the results.

Table 2. Three different methods for plane-based camera tracking

Name	Description
Nothing	General one. No homography refinement. No use of temporal coherence Feature detection/description: SURF
HomoRef	Only with homography refinement. No use of temporal coherence Feature detection/description: SURF
Proposed	The proposed method. With both homography refinement and use of temporal coherence Feature detection/description: FAST/ BRIEF

Table 3. Processing time and jitter reduction rate of *HomoRef* and *Proposed* in the case where only camera translations exist

Processing time (ms/frame)		Jitter reduction rate (%)	
HomoRef	*Proposed*	*HomoRef*	*Proposed*
157.006	28.43	11.21	31.30

Both *HomoRef* and *Proposed* had the effect on reducing jitter (thus improving stability). However, *Proposed* was better than *HomoRef*. It may be because BRIEF is more powerful for describing/matching feature points in/between consecutive frames than SURF. In terms of processing time, *Proposed* was much better than *HomoRef* because it has no the step for estimating the initial homography and BRIEF is much faster than SURF.

For the image sequences where the camera was fixed and the planar patterns were rotated by given angles around each axis, three methods of Table 2 were compared. Figure 2 and Table 4 show the results.

Table 4. Processing time and jitter reduction rate of *HomoRef* and *Proposed* in the case where only camera rotations exist

Processing time (ms/frame)		Jitter reduction rate (%)	
HomoRef	*Proposed*	*HomoRef*	*Proposed*
174.46	29.86	17.64	26.81

As with the translation case, *Proposed* further

improved the processing speed and the stability than *HomoRef* although both are better than *Nothing*. Actually, although BRIEF was not designed to be rotationally invariant, the weakness could be overcome by using the temporal coherence, i.e., making the current frame very similar to reference image by image warping.

Finally, using *Proposed*, we implemented a simple 2D augmented reality system. A result is shown in Fig. 3 where a virtual object was reliably overlaid on the real plane.

Figure 3. A result of 2D augmented reality system

4 CONCLUSION

In this paper, we proposed a robust and fast plane-based camera tracking method which was based on recursive homography refinement and use of temporal coherence between consecutive frames. With the proposed method, we could implement reliable 2D augmented reality systems.

5 Acknowledgement

This work was supported by the IT R&D program of KEIT.

[KI002058, Signal Processing Elements and their Hardware IP Developments to Realize the Integrated Service System for Interactive Digital Holograms]

6 REFERENCE

[1] D.G. Lowe, "Distinctive image features from scaleinvariant keypoints," International Journal of Computer Vision, vol. 60, pp. 91–110, 2004.

[2] H. Bay, A. Ess, T. Tuytelaars, and L. V. Gool, "SURF: Speeded up robust features," Computer Vision and Image Understanding, vol. 110,pp.346-359, 2008.

[3] S. Leutenegger, M. Chli, and R. Y. Siegwart, "BRISK: Binary robust invariant scalable keypoints," in Proc. of ICCV, pp. 2548-2555, 2011.

[4] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, "BRIEF: Binary robust independent elementary features," in Proc. of ECCV, pp. 778-792, 2010.

[5] R. Hartley and A. Zisserman, Multiple Viw Geometry, Cambridge University Press, 2004.

[6] E. Rosten and T. Drummond, "Fusing points and lines for high performance tracking," in Proc. of ICCV, pp. 1508 - 1515 ,2005.