3D Res (2015)6:41
DOI 10.1007/s13319-015-0072-5

3DR EXPRESS

Fast Markerless Tracking for Augmented Reality in Planar Environment

Ahmad Hoirul Basori · Fadhil Noer Afif · Abdulaziz S. Almazyad · Hamza Ali S. AbuJabal · Amjad Rehman · Mohammed Hazim Alkawaz

Received: 27 September 2015 / Revised: 30 October 2015 / Accepted: 31 October 2015

Abstract Markerless tracking for augmented reality should not only be accurate but also fast enough to provide a seamless synchronization between real and virtual beings. Current reported methods showed that a vision-based tracking is accurate but requires high computational power. This paper proposes a real-time hybrid-based method for tracking unknown environments in markerless augmented reality. The proposed method provides collaboration of vision-based approach with accelerometers and gyroscopes sensors as camera pose predictor. To align the augmentation relative to camera motion, the tracking method is done by substituting feature-based camera estimation with combination of inertial sensors with complementary filter to provide more dynamic response. The proposed method managed to track unknown environment with faster processing time compared to available feature-based approaches. Moreover, the proposed method can sustain its estimation in a situation where feature-based tracking loses its track. The collaboration of sensor tracking managed to perform the task for about 22.97 FPS, up to five times faster than feature-based tracking method used as comparison. Therefore, the proposed method can be used to track unknown environments without depending on amount of features on scene, while requiring lower computational cost.

Keywords Markerlesstracking · Accelerometers · Unknown environment

A. H. Basori · H. A. S. AbuJabal
Faculty of Computing and Information Technology Rabigh, King Abdulaziz University, Jeddah, Kingdom of Saudi Arabia

F. N. Afif
Faculty of Computing, Universiti Teknologi Malaysia, Johor Bahru, Malaysia

A. S. Almazyad
College of Computer and Information Sciences, King Saud University, Riyadh, Kingdom of Saudi Arabia

A. S. Almazyad · A. Rehman (✉)
College of Computer and Information Systems, Al-Yamamah University, Riyadh, Kingdom of Saudi Arabia
e-mail: a_khan@yu.edu.sa

M. H. Alkawaz
Faculty of Information Sciences & Engineering, Management and Science University Shah Alam, Selangor, Malaysia

M. H. Alkawaz
Faculty of Computer Sciences and Mathematics, University of Mosul, Mosul, Iraq

Published online: 19 November 2015

1 Introduction

Augmented Reality (AR) is a new domain to mimic technology generated objects in real life. This technology superimpose virtual objects onto a scene of real world captured by a video camera. Users could see these objects with display devices such that real & virtual objects have co-existance. Augmented reality is capable of displaying information that could not attined by listening nor watching. Hence, it extends human senses and improve user performance of real-world activities. Several researchers attempted to imitate virtual and real world; accordingly in augumented reality an interesting survey is conducted by Zhou et al. [1]. In this survey, tracking methods are one of the fundamental topics for AR system development, and is the most popular sub-fields to be explored. In this paper has five important aspects required for a fast tracking method on augmented reality, namely: (1) introduction, (2) Augmented reality and markerless tracking, (3) Methodology, (4) Experimental result and (5) conclusion. This paper emphasizes the motivation and background of markerless tracking and augmented reality. The rest of this paper is organized as follows. In Sect. 2, we discuss important issues augmented reality problem and issues. Section 3, a detailed methodology are presented and discussed to explain how the research is carried out. In Sect. 4 an experimental result is obtainable and deliberated in detailways. Finally, discussions and conclusions are drawn in Sect. 5.

2 Augmented Reality and Markerless Tracking

In recent years development of interaction between humans and machines has been improved tremendously. Inevitably, connection with digital entities becomes an essential part of human regular activities. As one of the connecting bridge in human–machine interaction, Augmented Reality (AR) attempts to blend real objects with virtually-generated beings within a single dimension. Such virtual objects are typically superimposed to desired place on a scene captured by camera [2]. These real objects along with the "augmented" one are displayed to the user, so that it appears together seamlessly. Users can directly manipulate virtual beings as if they are in the real dimension. Examples of human–machine interaction using AR are shown in Fig. 1. Thus, AR can provide enhanced information that initially cannot be perceived from the environment by regular human senses.Tracking for Augmented Reality demands much more accuracy than in its sibling Virtual Reality. In VR, the whole world of the user is being replaced with the artificial one [3]. If the user's virtual hand shifts a bit than it should be in the real world, the user may not realize it because his perception is completely overridden by the content provided by the virtual environment.

However, this is not the case for Augmented Reality. A small errors can be recognized easily, because users can still see the real world where the virtual objects is augmented into. An example of correct tracking is shown in Fig. 1, where an AR system shows information of NASCAR drivers information (name, speed, position, number) according to their respective car. When these cars move to complete their race, the system will track the movement, then the driver's info balloon will follow in real-time. If the tracking system fails, the balloon will not follow the car and it will disrupt the user perception easily.

Handy AR attempted to replace conventional fiducial markers with user's outstretched hand, by using skin-color detection and contour extraction [4]. Area of user's hand was detected and segmented by comparing it with previously learned histogram of hand's skin colour. Then, hand's boundary was generated by calculating distance between centre blob and the farthest point from the hand. Within this boundary, the system narrowed its target to detect user's fingertips. This was done by implementing

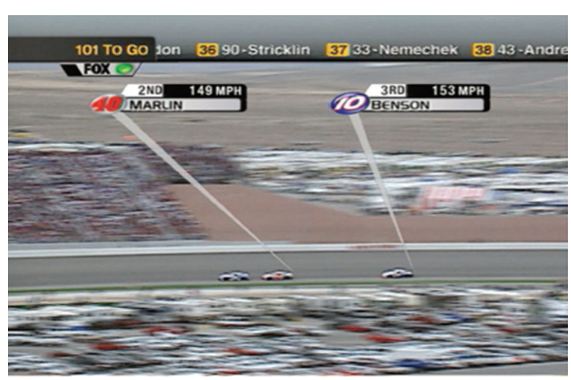

Fig. 1 AR on professional car racing showing race information. Image courtesy of NASCAR

contour detection to generate ellipses surrounding curvature of the fingertips' contour. Each of the five user's fingertips can be indexed by detecting the thumb as a point with farthest mean distance. This model, along with calculated camera parameters is applied to estimate the camera's pose. According to the results, the method is comparably in par with marker-based tracking techniques in terms of accuracy, with average RMS error of 5.86 pixels. However, fingertips detection could fail when hand is exposed to large change of illumination, such as in outdoor scenes. It also showed an error when the hand was occluded by object with similar skin-color [5,6].

In this research, we are particularly interested in planar, unknown environments, which can be commonly found in any place, e.g. walls, floors or workspaces. When the scene is unknown, i.e. there is no prior information to do the tracking (such as by training system to detect particular markers or CAD models on the scene), it is considered a very difficult task. Therefore, researchers introduced a constrain for tracking planar scenes only [7–9]. Some AR tracking methods focus more on tracking planar, unknown environments. It attempts to deduce the tracking without initial knowledge of the scene and put the information in some form of map. Commonly called as SLAM, when new features were discovered, it expands the map so the knowledge grows incrementally. Neubert et al. [8] used SLAM to construct models from the captured scene.

One way to achieve markerless Augmented Reality is done by making the system to calculate the camera pose based on what it observes on screen. The system tries to detect any regions that have abundant information about the scene as features. These features will then be calculated and tracked in every frame, so that the camera pose can be calculated continuously. These kind of techniques are often called as feature-based tracking, and it is widely used in more recent Augmented Reality system [1, 10].

However, In order to achieve robust and optmimum tracking performance, kinds of features tracked needs to be specified. The chosen feature must be robust to motion and environmental changes such as illumination. It should also be easily detectable and deduceable. Some features that has been used by researchers are point features, edges and feature descriptors [11].

Point features are sub-areas in the captured image that can be seen clearly. Point feature's position is capable to be deduced correctly. Localization of point features can also be obtained easily throughout sequence of video frame and furthermore their correspondences can be determined [12]. Thus, point feature are reliable as target to track unpredictable motion. Another special features that can be extracted from image sequences is corners, which are the special point features lying in the crossing of edges. Because of the uniqueness of these features, corners' attributes were used to determine matching quality of the tracking [13, 14].

Often, tracking point features will be complemented with another feature known as edges. Edges have been widely used as initial stage in image processing and also as main feature in many computer vision applications. They can be defined as areas corresponding with discontinuations occuring in an image [15]. According to Lindeberg [16], in the process of image formation, these discontinuities can be potentially related to depth or orientation. A physical properties of captured objects are strongly interrelated with appearance of edges [16]. In the early days of edge detection technique, an edge, characerized by sudden intensity change can be detected by solving zero-crossing changes of the convolution equation [17]:

$$\nabla^2 G(x, y) * I(x, y) \tag{1}$$

where I is the image, G(x,y) is two dimensional Gaussian kernel distribution, and is the Laplacian of them. Edges are another choice for tracking because it provides feature connectivity that does not owned by normal point features. Furthermore, they are also resistant to change in various lighting conditions and aspect changes [12]. An example of tracking edges and corners is shown in Fig. 2, where the system tracks available corners and edges of the objects captured in a scene. Harris detector and FAST [12] are famous feature detection techniques used by researchers around the world.

In Augmented Reality, most sensor type used are position-measuring devices. For example, one can use ultrasonics to measure distance by calculating travelling time of ultrasonic waves emitted. Some kind of sensors that have been used are GPS, wave sensors and inertial sensors. A position reside in three-dimensional (3-D) space can be described with its correspondence

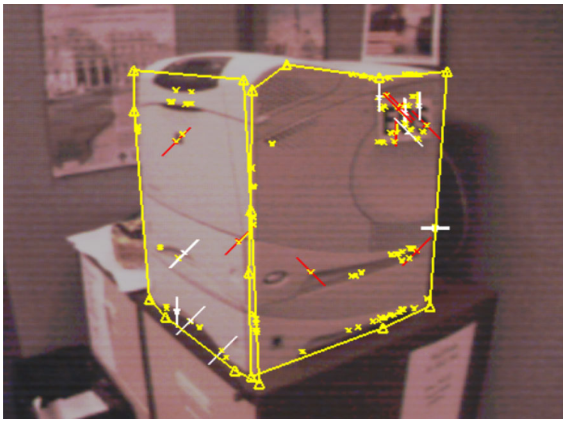

Fig. 2 Tracked edges and corners from a scene captured by camera [8]

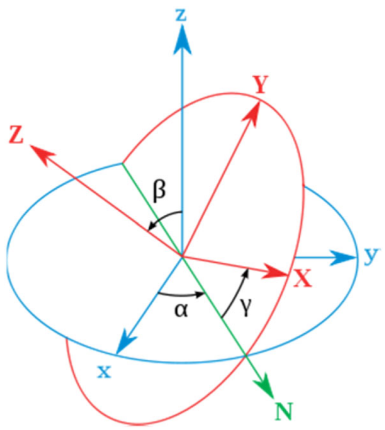

Fig. 3 The position and orientation of object. The xyz system is the reference fixed coordinate system, while XYZ are the position and αβγ are the orientation. Image courtesy of Wikimedia

inside Cartesian coordinate system. This position is determined by six values: three for linear coordinates and three for orientation value, illustrated in Fig. 3. Linear coordinates is the measure (in length) of difference to move the object's coordinate with respect to the reference. On the other hand, orientation is a measure (in degree) of rotation of the object with respect to the reference. Frequently, the measurements are called length, width, height for position, and pitch, yaw, roll for orientation, usually utilized for ship movement [18]. The orientation may be also called as Tait-Bryan angles or Euler angles [19, 20].

Because AR tracking works in real-world space, the reference coordinate used is the world coordinate

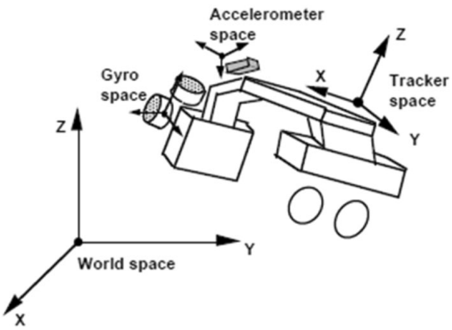

Fig. 4 Example of sensor configuration. The sensor will measure the position and orientation with respect to the world coordinate [25, 26]

system. The sensor-installed device translates its position according to this reference. The system exploits these sensors to obtain the motion of the user with respect to the reference centre of world coordinate, as illustrated in Fig. 4. When these measurements are obtained, the rendering module will draw the objects accordingly. The sensors will do the measurements continuously to update the change of user's viewing position.

In Augmented Reality, most sensor type used are position-measuring devices. For example, one can use ultrasonics to measure distance by calculating travelling time of ultrasonic waves emitted. Some kind of sensors that have been used are GPS, wave sensors and inertial sensors.

3 Proposed Methodology

Markerless tracking is beneficial to fully integrate virtual into real-world, thus enhancing realism of Augmented Reality technology. Such method enables a natural, robust feature-tracking technique, but there are issues occurred when tracking minimum-feature scene. The markerless tracking estimates the tracking by firstly generating "feature map" containing the features observed from scene in their respective location. Accordingly when the system runs, it will track any changes happened due to camera motion, and estimate the new camera motion based on the change and the feature map.

In the proposed method, the sensor-based estimationis highlighted, as shown in Fig. 5. Feature-based approach is chosen as a ground truth estimation so that the method is capable to track any planar structure

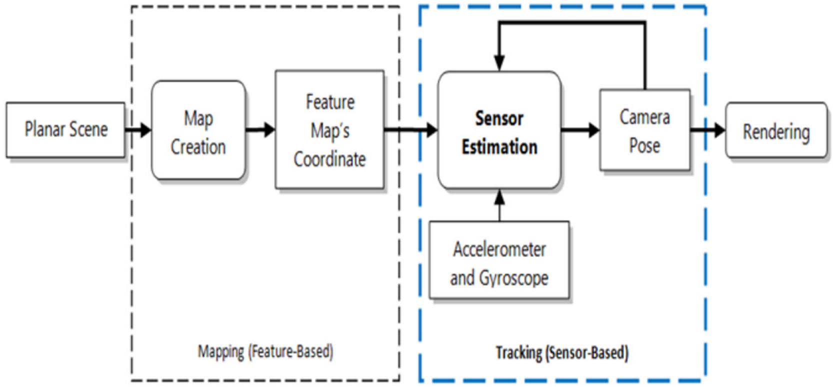

Fig. 5 Overview of the enhanced method

available on the scene. The same map making approach is used to generate the base of estimation, and then sensor-based approach is used as a complement for feature tracking. Main part of this method is to estimate the orientation of the camera according to features that are detected by map creation process (blue highlighted). The process will done continously according to the motion of the camera.

This sensor will be used to complement the tracking process done by the camera by giving extra information to the system. Because camera pose consists of 3-DOF translation and 3-DOF rotation, accelerometer and gyroscope can be used used as position tracker. In this research, we will focus on estimating the orientation component of the camera pose. These sensors will be attached to the camera and used to measure orientation value relative to the ground truth. The workflow is shown in Fig. 6. In order for sensor to deduct correct orientation of the camera, calibration will be done as the first step. Calibration is done to determine the origin of coordinate for the system, so that each position value will be measured relative to this center. In this research, the calibration point will be determined as the position of the dominant plane obtained in the map creation process in the previous step. This position will be treated as ground truth and any measurement will be done relative to it.

Here, the estimation of camera pose, particularly the orientation will be done by digital accelerometers and gyroscopes. The accelerometer provides measurement of acceleration acting on itself. Hence, when there is no external force act in the device, it will measure acceleration due to earth's gravity. By determining gravity vectors, accelerometer can act as 2-DOF orientation measurement for pitch and roll angle [21, 22]. For example the estimation of

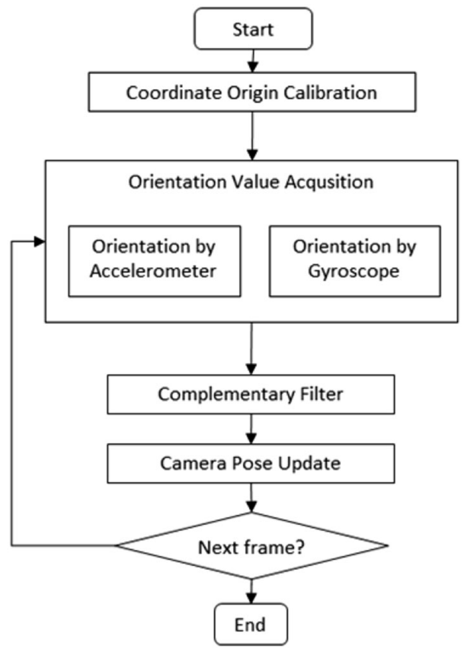

Fig. 6 Workflow for pose estimation by gyroscope

inclination angle $\widehat{Z}_A$ with respect to earth's horizontal is formulated as [23, 24]:

$$\hat{Z}_A = \frac{y_t - \hat{a}_t^-}{\left|y_t - \hat{a}_t^-\right|} \tag{2}$$

where y_t is the accelerometer signal, and $\hat{a}_t^-$ is the estimated acceleration signal without the effect of gravity. Signal from a gyroscope's direct output measures the angular speed $\boldsymbol{\omega}$. The angular orientation itself can be approximated numerically by integrating the signal over a time-span Δt:

$$\theta_k = \theta_{k-1} + \omega_k \cdot \Delta t \tag{3}$$

where k represents the current time/iteration. Thus, orientation of a device can be measured by two kinds of such inertial sensor.

A complementary filter enables fusing of multiple signals intended for the same measurement. It works by taking the noisy measurements and complementing their special characteristics into a single output signal. A complementary signal works by taking two measurements $y_1 = x + \mu_1$ and $y_2 = x + \mu_2$ of the measured x, where $\boldsymbol{\mu_1}$ and $\boldsymbol{\mu_2}$ is the high and low frequency noise. The estimation of filter output $\hat{\boldsymbol{X}}(\boldsymbol{s})$ is determined in frequency domain as:

$$\hat{X}(s) = F_1(s)Y_1 + F_2(s)Y_2 \tag{4}$$

where Y_1 and Y_2 is signal from both measurement, $F_1(s)$ and $F_2(s)$ are the associated low pass and high pass filter with total gain of $F_1(s) + F_2(s) = 1$.

As stated previously, due to the nature of its technology an accelerometer is not adequate for dynamic measurement. While these can be handled by replacing it with gyroscopes, it has drift error caused by integration of the angular speed signal. In short, in orientation measuring one can trust gyroscope in dynamic short-term, while in long-term accelerometers will become more dependable. In this paper, the benefit of signals from both accelerometers and gyroscopes will be combined by a specific method called complementary filtering. The complementary filtered output θ_c is obtained by filtering gyroscope signal θ_g to the high-pass and orientation from accelerometer θ_a so that Eq. 4 becomes:

$$\theta_{k,c} = f_1 \cdot \theta_{k,a} + f_2 \cdot \left(\theta_{k-1,c} + \omega_{k,g} \cdot \Delta t\right) \tag{5}$$

From Eq. (5), an estimation of the orientation based on combination of accelerometer and gyroscope measurement can be obtained. It is noted, however, that the complementary filter cannot process the yaw orientation, because accelerometers are unable to measure such signal (due to no change in gravity vector). Finally, orientation component of the camera pose, represented by pitch α, yaw β and roll γ, is estimated as:

$$\begin{aligned}
\hat{\alpha}_{k,c} &= \alpha_0 + \left[F_1 \cdot \alpha_{k,a} + F_2 \cdot \left(\alpha_{k-1,c} + \dot{\alpha}_{k,g} \cdot \Delta t\right)\right] \\
\hat{\beta}_{k,c} &= \beta_0 + \beta_{k-1,c} + \dot{\beta}_{k,g} \cdot \Delta t \\
\hat{\gamma}_{k,c} &= \gamma_0 + \left[F_1 \cdot \gamma_{k,a} + F_2 \cdot \left(\gamma_{k-1,c} + \dot{\gamma}_{k,g} \cdot \Delta t\right)\right]
\end{aligned} \tag{6}$$

where origin $(\alpha_0, \beta_0, \gamma_0)$ is the orientation of the feature map's dominant plane. These calculations are done for each incoming signals so that the camera pose will be updated dynamically. Finally, the values obtained from sensors will be used to estimate the orientation of the camera pose relative to the dominant plane. When the camera moves in rotating manner, accelerometer and gyroscope will estimate how far it deviates from the original position, and then they are combined. In other words, the position of camera is obtained from both accelerometer and gyroscope measurement collaborated by the complementary filter. The resulting orientation is regarded as the estimation of the camera pose relative to the initial position.

In this Paper, a sensor-based pose estimation algorithm is developed. The pose estimation is then integrated to an existing mapping algorithms and then combined to construct a hybird markerless tracking method for augmented reality. The proposed algorithm is shown by Fig. 7. The algorithm consists of two steps of mapping and tracking that provides estimation of camera orientation as tracking output. It starts by creating a map based on features on scene, and then tracking such map with the combined accelerometer and gyroscope.

Algorithm for Fast Markerless Tracking

```
1.  Start
2.  Capture stereo pair from the scene
3.  Mapping
       a. Initialize map from stereo pair
       b. Bundle Adjustment
       c. Feature Estimation
       d.    Obtain map's dominant plane
             coordinate
4.     Map's coordinate = Tracking
       coordinate origin
5.  Tracking
       a. Obtain accelerometer orientation
          value
       b. Obtain gyroscope orientation
          value
       c.   Combine by complementary filter
       d. Camera pose = filtered
       orientation
6.  Render virtual objects
7.  Go back to 5 through system run
8.  End
```

Fig. 7 Algorithm of the proposed method

4 Experimental Result

In this section, the inertial sensors output is analyzed and discussed. This research, two kinds of sensors are used: digital accelerometers and gyroscope. While both of these may give inertial measurement, in particular orientation (rotation). Hence, some controlled experiments were conducted to show the output of orientation given. In the following cases, both accelerometers and gyroscopes will act as an orientation measurement device.

4.1 Sensors Output

The sensors is mounted on same rigid platform so that it will move at the same displacement value. The resulted output will be recorded for a fixed length of iteration and the signal response will be analysed. Firstly, the measurement is done individually. Then, a complementary filter is introduced to the sensors so that output from accelerometer and gyroscope is combined together. This filtered output will then analysed to be compared with the preceding one.

The experiment conducted is to measure the output when the remote is standing still, i.e. there is no movement acted on the device. Intuitively it can be said that the ground truth measurement is be zero for all time. Figure 8 shows the example result of angular speed from calibrated gyroscope output measured for about 18 s, with the measurement interval of 10 ms. According to the result, the three orientation component (pitch, yaw, roll) showed averagely zero value along measurement time. It can be noticed, however, that noise occurs on the measurement so that the value oscillated. The first experiment showed mean pitch, yaw, roll value of −0.8391, −0.9258, −0.7046 deg/s respectively, and with the uncertainty expressed as standard deviation value of 0.4529, 0.2931, 0.6909 deg/s, respectively.

In order to obtain the angular position (orientation) the previous gyroscope output is numerically integrated once over a time-step along the measurement time. Fig. 9 shown plotting result. According to the result, the four experiments showed that the resulted orientation did not stay still in the zero value, in fact drifts occur in the measurement. Result showed that the pitch, yaw, roll value drifts for −8.732°, −9.311°, and −7.257°, respectively, after the system runs for 10 s. The drift came larger when the system runs longer, reaching −62.33°, −78.67°, and −53.34°, respectively, after more than 90 s running. Thus, when the remote is standing still without any movement, the measurement from gyrosensor yields integrated angular position with drift.

The following result display the response output from the accelerometers, shown by Fig. 10. The gravity force measured by accelerometer yields an estimation for pitch and roll angle. However, accelerometers are unable to measure yaw due to no change of gravity happened in the yaw angle. Here it can be seen that either pitch or the roll angle stays on relatively stable output. In some point of measurement, there are variances occurred in form of spikes.

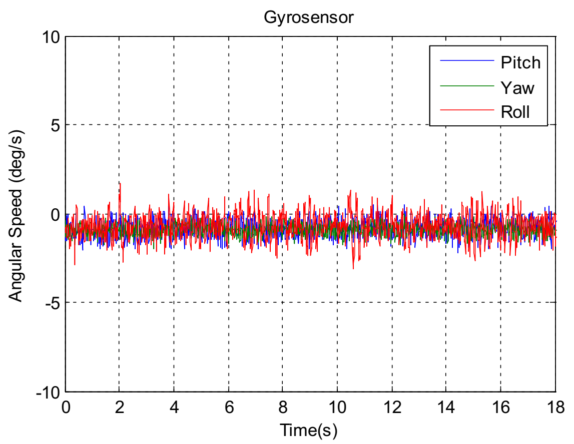

Fig. 8 Calibrated angular speed from gyrosensors output with respect to time. The angular speed consist of pitch (*blue*), yaw (*green*) and roll (*red*). (Color figure online)

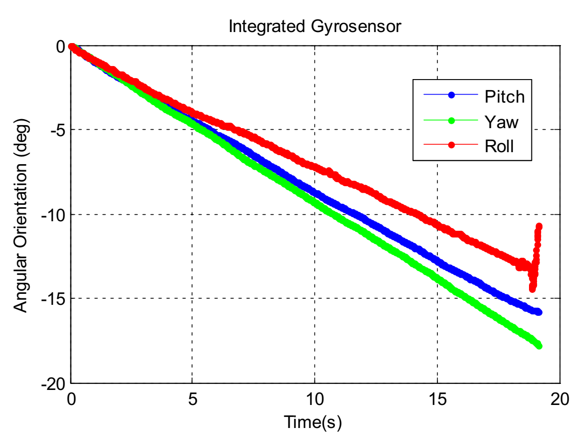

Fig. 9 Obtained angular orientation from integrating the angular speed over a time span. Each sub-figure displays consisting of pitch (*blue*), yaw (*green*) and roll (*red*). (Color figure online)

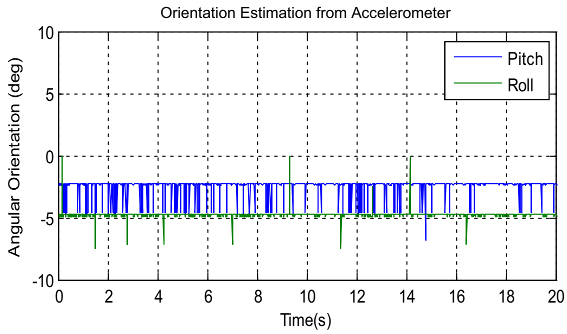

Fig. 10 Estimated angular orientation from accelerometer

In this experiment, the mean of orientation measurement for pitch and roll is $-2.497°$ and $-4.823°$ with deviation of $0.635°$ and $0.241°$, respectively.

The result also showed that the accelerometers estimation yielded an offset to the measurement, hence, the measurement will be corrected with the average value. Figure 11 shows the corrected value of such measurement, resulting in mean value for pitch and roll of $0.361°$ and $0.031°$, respectively. According to these result, measurement from accelerometer is more stable without drift as was happened in the gyroscope.

In the Fig. 12, a complementary filter is used to estimate orientation from both accelerometers and gyroscopes. According to the result, the pitch and roll value resulted in a more steady manner, as compared to its preceding sensor output. It can be seen that the complementary filter eliminates accelerometer's spike and gyroscope's error drift so that the output become near to the estimated truth. In this experiment, the filter constant value used was 0.75 s, resulted in mean orientation output for pitch and roll of $-0.603°$ and $-0.503°$ with deviation of $0.1°$ and $0.187°$, respectively.

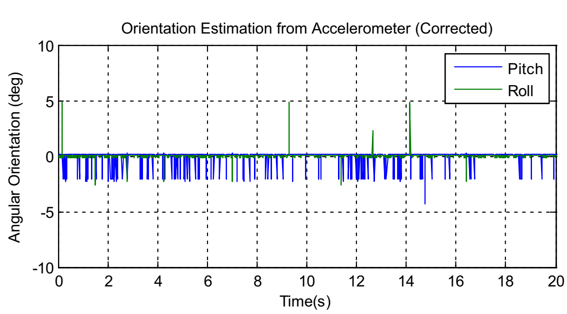

Fig. 11 Orientation result after correction from average value

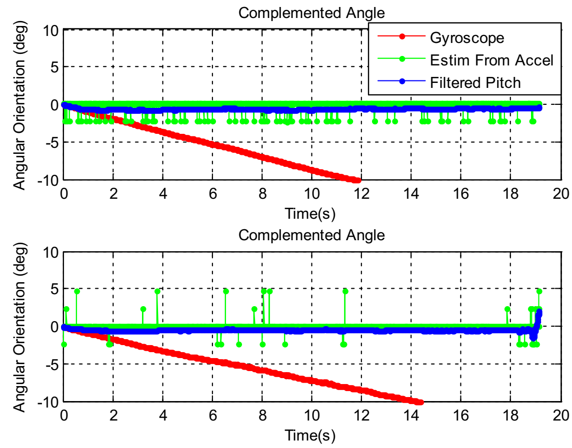

Fig. 12 Orientation yielded from sensors and from filtering process

According to these result, it can be deduced that usage of complementary filter can correct some errors from both the accelerometers and gyroscope. When the system is near to static, the complementary filter will pass the accelerometer value, and conversely the gyroscope signal will be passed when the system shifts to more dynamic behavior, i.e. camera is moving. Next, some experiment to measure the performance of the system are conducted and analyzed.

4.2 Orientation

As described in the literature review, tracking in Augmented Reality means to estimate the position and movement of camera according to the observation on the scene, while the system run iteratively. In the next experiment, an analysis will be done to the result of tracking itself, which is the orientation of the camera (camera pose). The resulted orientation, which is the combination of estimation from accelerometer and gyroscope will be evaluated. Also, the capability of proposed method to aid feature-based tracking method when it fails in tracking the scene will be analyzed.

In this experiment, some scenes were used for evaluation is shown in Fig. 13. For each of these environment, orientation measurement were conducted. The measurement is done for each succession of iteration, i.e. time will be measured when the tracking runs recurrently.

Furthermore,variety of scenes were used for evaluation is shown in Fig. 14. For each of these environment, orientation measurement were conducted. The measurement is done for each succession of iteration, i.e. time will be measured when the tracking runs recurrently. To evaluate such issues, a comparison for the proposed method is made. PTAM, as the base method used to develop the proposed method, will be used again as the benchmark. Both method will be run altogether in the same time. Each iteration, the system will estimate both PTAM and our method estimation, resulting in two different estimations of measurements. The result is then compared and analyzed to look at behaviour of the proposed method.

The example scenario of measuring orientation is by moving the camera so that the orientation estimation changes, as illustrated in Fig. 14.

Figure 15 shows orientations obtained from a measurement, which is done for 300 iterations. The measurement is done at the same time. As can be seen from the graph, the proposed method follows the estimation of PTAM. When the camera pose is changed by means of rotating, the orientation changes and this yields a change in the output. As was shown by the figure, The proposed method managed to measure the same behavior with PTAM. It is notable, however, that the errors occurred in the measurement, as can be seen starting from about 120th iteration where the pitch and yaw signal drifts from PTAM

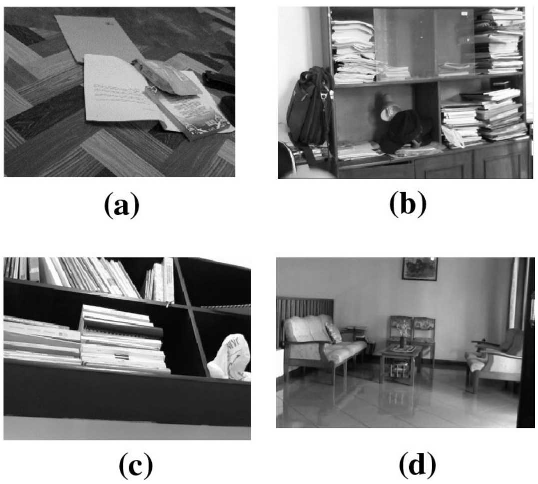

Fig. 13 Scenes used for measuring processing speed

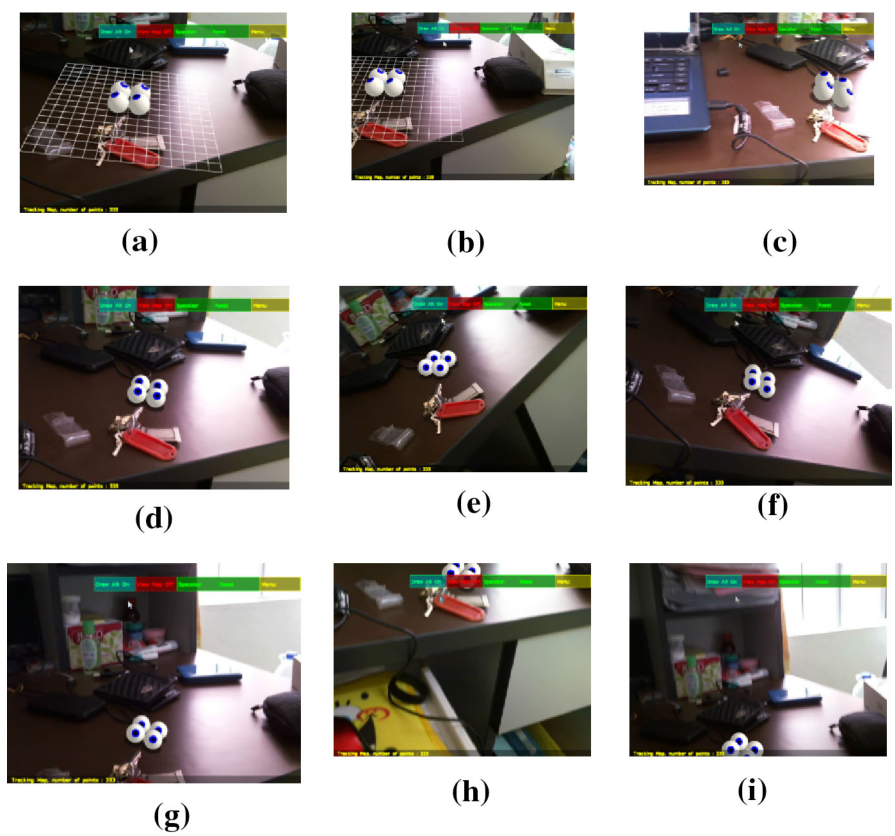

Fig. 14 Example scenario to measuring orientation value. **a–c** Adjusting yaw angle, **d–f** roll angle, and **g–i** pitch angle

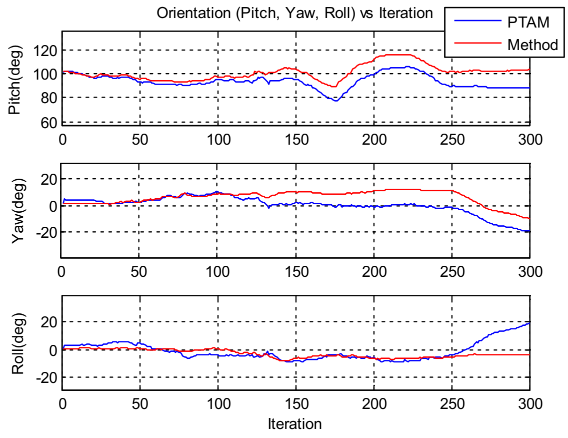

Fig. 15 Output orientation of proposed method and PTAM

measurement. Also, while the roll signal estimates the angle closely, at 250th iteration onwards the drifts also occurred.

In this paper, some information about the resulting processing speed will be shown. The experiment is conducted by measuring processing speed of each iteration. As a benchmark, the underlying method PTAM [1] will be run with the same treatment as the proposed method. According to workflow of the proposed method, the tracking module runs after a map creation has been generated by map-making module.

Figure 16 showed the measurement resulted from the fourth scene. As depicted by the plot, in this experiment the proposed method also runs on faster processing time on averagely 57.95 ms (17.25 frame per second), while PTAM runs on 210.53 ms (4.75 frame per second) for tracking 435 feature points. It is also noted that there are spikes for a number of times in the PTAM measurement, due to the method's attempt

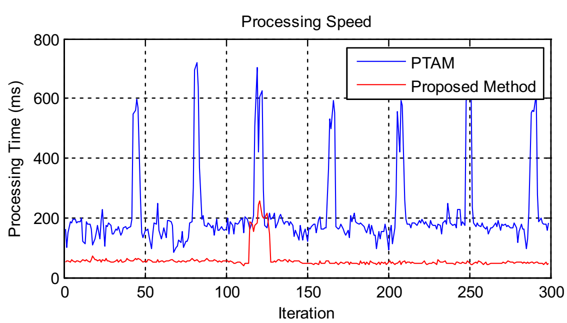

Fig. 16 Processing speed from fourth scene. Feature points in the final map on the end of iteration (proposed method—PTAM): 311 and 435

to relocalise feature previously lost. In this experiment the proposed method managed to have lower processing time throughout the entire iteration, so that it run on 363 % faster than PTAM.

According to these results, it can be seen that the proposed method is able to run on shorter processing time comparing to PTAM. Using inertial sensors with complementary filter result in direct input to the system, while a feature-based tracking attempts to calculate as many features in the scene as possible. Consequently, the method requires lower computational power to obtain its camera pose. The proposed method can sustain its computational costs with respect to amount of feature points stored in its map.

5 Conclusion

The proposed method.consists of a feature-based map creation and sensor-based estimation. The base method, PTAM [7] utilized a fully vision-based method to track any planar scene in the real environment. In this research, instead of using feature points to detect the scene, a collaboration of accelerometer and gyroscope with a complementary filter replaced the existing camera pose estimation approach. The result if quite promising since the speed of tracking reach triple times (363 %) than the based method. This markerless tracking has big potential, it can be implemented not only for Augmented Reality but it also can be used for tracking moving object or moving subject in robotic or unmanned vehicle.

References

1. Zhou, F., Duh, H. B. L. & Billinghurst, M. (2008). Trends in augmented reality tracking, interaction and display: A review of ten years of Ismar. In *Proceedings of the 2008 7th international symposium on mixed and augmented reality (ISMAR 2008): ACM & IEEE* (pp. 193–202).
2. Saba, T., & Altameem, A. (2013). Analysis of vision based systems to detect real time goal events in soccer videos. *Applied Artificial Intelligence, 27*(7), 656–667.
3. Saba, T., & Rehman, A. (2012). *Machine learning and script recognition* (pp. 29–34). Saarbrücken: Lambert Academic Publisher.
4. Lee, T. and Hollerer, T. (2007). Handy Ar: Markerless inspection of augmented reality objects using fingertip tracking. In *IEEE proceedings of the 2007 IEEE 11th IEEE international symposium on wearable computers* (pp. 1–8).

5. Soleimanizadeh, S., Mohamad, D., Saba, T., & Rehman, A. (2015). Recognition of partially occluded objects based on the three different color spaces (RGB, YCbCr, HSV). *3D Research,*. doi:10.1007/s13319-015-0052-9.
6. Haron, H., Rehman, A., Adi, D. I. S., Lim, S. P., & Saba, T. (2012). Parameterization method on B-spline curve. *Mathematical Problems in Engineering,*. doi:10.1155/2012/640472.
7. Klein, G., & Murray, D. (2007). Parallel tracking and mapping for small Ar workspaces. In *Proceedings of the 2007 IEEE 6th international symposium on mixed and augmented reality (ISMAR 2007): IEEE & ACM* (pp. 225–234). Orlando.
8. Neubert, J., Pretlove, J., & Drummond, T. (2007). Semi-autonomous generation of appearance-based edge models from image sequences. In *ISMAR '07 Proceedings of the 2007 6th IEEE and ACM international symposium on mixed and augmented reality* (pp. 1–9). *Washington DC*.
9. Uematsu, Y., & Saito, H. (2009). Multiple planes based registration using 3d projective space for augmented reality. *Image and Vision Computing, 27*, 1484–1496.
10. Mooser, J., You, S., Neumann, U., & Wang, Q. (2009). Applying robust structure from motion to markerless augmented reality. In *Proceedings of the 2009 IEEE 2009 workshop on applications of computer vision (WACV)* (pp. 1–8).
11. Siltanen, S. (2012). *Theory and applications of marker-based augmented reality*. Finland: VTT.
12. Rosten, E., & Drummond, T. (2005). Fusing points and lines for high performance tracking. In *Proceedings of the 2005 IEEE international conference on computer vision* (pp. 1508–1511).
13. Charnley, D., & Blisett, R. (1988). Surface reconstruction from outdoor image sequences. In *Proceedings of the 1988 Fourth Alvey Vision Club*.
14. Rehman, A., & Saba, T. (2014). Neural network for document image preprocessing. *Artificial Intelligence Review, 42*(2), 253–273. doi:10.1007/s10462-012-9337-z.
15. Mundher, M., Muhamad, D., Rehman, A., Saba, T., & Kausar, F. (2014). Digital watermarking for images security using discrete slantlet transform. *Applied Mathematics and Information Sciences, 8*(6), 2823–2830. doi:10.12785/amis/080618.
16. Lindeberg, T. (2011). Edge detection. Accessed November 22, 2012, from http://www.encyclopediaofmath.org/index.php?title=Edge_detection&oldid=17883.
17. Marr, D., & Hildreth, E. (1980). Theory of Edge Detection. *Proceedings of the Royal Society of London, 207*, 187–217.
18. Die Deutschen Versicherer (2012). Summary of Ship Movement. Accessed October 17, 2012, from http://www.containerhandbuch.de/chb_e/stra/index.html?/chb_e/stra/stra_02_03_03.html.
19. Weisstein, E. W. (2012). Euler angles. Accessed October 17, 2012, from http://mathworld.wolfram.com/EulerAngles.html.
20. Kato, H., & Billinghurst, M. (1999). Marker tracking and hmd calibration for a video-based augmented reality conferencing system. In *Proceedings of the 1999 IEEE computer society and 2nd IEEE and ACM international workshop on augmented reality (IWAR'99)* (pp. 85–94). Washington DC.
21. Mizell, D. (2003). Using gravity to estimate accelerometer orientation. In *Proceedings of the 2003 seventh IEEE international symposium of wearable computers* (pp. 252–253).
22. Muhsin, Z. F., Rehman, A., Altameem, A., Saba, T., & Uddin, M. (2014). Improved quadtree image segmentation approach to region information. *The Imaging Science Journal, 62*(1), 56–62.
23. Luinge, H. J., & Veltink, P. H. (2005). Measuring orientation of human body segments using miniature gyroscopes and accelerometers. *Medical & Biological Engineering & Computing, 43*(2), 273–282.
24. Weisstein, E. W. (2013). Plane. MathWorld—A Wolfram Web Resource. Retrieved June 22, 2013, from http://mathworld.wolfram.com/Plane.html.
25. Azuma, R. (1995). Predictive tracking for augmented reality. PhD Thesis, UNC-Chapel Hill.
26. Neamah, K., Mohamad, D., Saba, T., & Rehman, A. (2014). Discriminative features mining for offline handwritten signature verification. *3D Research,*. doi:10.1007/s13319-013-0002-3.