Journal of Electronic Imaging

JElectronicImaging.org

# Handheld pose tracking using vision-inertial sensors with occlusion handling

Juan Li
Maarten Slembrouck
Francis Deboeverie
Ana M. Bernardos
Juan A. Besada
Peter Veelaert
Hamid Aghajan
José R. Casar
Wilfried Philips

SPIE. IS&T imaging.org

Juan Li, Maarten Slembrouck, Francis Deboeverie, Ana M. Bernardos, Juan A. Besada, Peter Veelaert, Hamid Aghajan, José R. Casar, Wilfried Philips, "Handheld pose tracking using vision-inertial sensors with occlusion handling," *J. Electron. Imaging* **25**(4), 041012 (2016), doi: 10.1117/1.JEI.25.4.041012.

# Handheld pose tracking using vision-inertial sensors with occlusion handling

**Juan Li,[a,*] Maarten Slembrouck,[b] Francis Deboeverie,[b] Ana M. Bernardos,[a] Juan A. Besada,[a] Peter Veelaert,[b] Hamid Aghajan,[b,c] José R. Casar,[a] and Wilfried Philips[b]**
[a]Universidad Politécnica de Madrid, ETSI Telecomunicación, Avenida Complutense 30, Madrid 28040, Spain
[b]Ghent University, TELIN-IPI-iMinds, Sint-Pietersnieuwstraat 41, Ghent 9000, Belgium
[c]Stanford University, Department of Electrical Engineering, 350 Serra Mall, Stanford, California 94305, United States

**Abstract.** Tracking of a handheld device's three-dimensional (3-D) position and orientation is fundamental to various application domains, including augmented reality (AR), virtual reality, and interaction in smart spaces. Existing systems still offer limited performance in terms of accuracy, robustness, computational cost, and ease of deployment. We present a low-cost, accurate, and robust system for handheld pose tracking using fused vision and inertial data. The integration of measurements from embedded accelerometers reduces the number of unknown parameters in the six-degree-of-freedom pose calculation. The proposed system requires two light-emitting diode (LED) markers to be attached to the device, which are tracked by external cameras through a robust algorithm against illumination changes. Three data fusion methods have been proposed, including the triangulation-based stereo-vision system, constraint-based stereo-vision system with occlusion handling, and triangulation-based multivision system. Real-time demonstrations of the proposed system applied to AR and 3-D gaming are also included. The accuracy assessment of the proposed system is carried out by comparing with the data generated by the state-of-the-art commercial motion tracking system OptiTrack. Experimental results show that the proposed system has achieved high accuracy of few centimeters in position estimation and few degrees in orientation estimation. *© 2016 SPIE and IS&T* [DOI: 10.1117/1.JEI.25.4.041012]

Keywords: Pose tracking; sensor fusion; camera networks; augmented reality.

Paper 15905SS received Dec. 15, 2015; accepted for publication Jun. 20, 2016; published online Jul. 12, 2016.

## 1 Introduction

The estimation of a mobile device's pose in three-dimensional (3-D) space, i.e., the calculation of its position and orientation in respect to a reference coordinate system, is a critical process for many applications, including augmented reality (AR) (e.g., to overlay virtual content upon the reality), virtual reality (VR) (e.g., to navigate through a computer-simulated world), and interaction in smart spaces (e.g., to remotely control smart devices). Imagine a smart space (e.g., home, office, or museums) filled with controllable and interactive objects. In this space, based on the estimated position and orientation of a mobile device, context awareness of the surroundings, resource recommendations, and interaction with smart objects can be implemented. Ideally, a pose estimation system should be able to provide pose information with no error and no latency. However, despite the progress that has been made to date, current technologies still offer limited performance in terms of accuracy, cost, robustness, computational complexity, ease of deployment, and on-board power consumption.

In this context, the contribution described in this paper is to propose an accurate, fast, robust, and low-cost handheld pose tracking system that fuses data from vision (external cameras) and acceleration sensors (embedded in devices), being customizable to smart space services and scalable for both small and wide areas. Three novel data fusion methods are presented, namely, the triangulation-based method for stereo-vision system, constraint-based method for handling partial occlusion, and triangulation-based method for a multicamera network. The handheld device needs to be equipped with two light-emitting diodes (LEDs) which are used as markers. Our marker differentiates from traditional markers, such as ARToolKit[1] and ARTag,[2] in several aspects. First, in the proposed approach, markers are tracked by external stationary and calibrated vision sensors. This scheme is popularly called outside-in tracking[3] in the literature. By processing data on a server, handheld devices are freed from the pose estimation process, leaving most computing power available for applications. Traditional squared marker-based approaches adopt the inside-out tracking scheme, that is, markers are placed in the environment and the device's pose is estimated by observing the markers from its internal camera; Second, the artificial markers in the environment may cause visual discomfort. Especially, in a wide area, a large number of markers need to be placed and carefully calibrated. On the contrary, this system does not need any artificial marker in the surrounding environment; Moreover, LEDs can be used to track objects with small surfaces, such as mobile devices, glasses frames, or drones, whereas traditional markers need a big and flat surface to be placed. Compared with the state-of-the-art commercial motion tracking system, such as OptiTrack[4] and Vicon,[5] the cost is largely reduced.

The integration of accelerometers reduces the number of unknown parameters in the pose calculation and accordingly diminishes the workload of the image processing task.

*Address all correspondence to: Juan Li, E-mail: li.juan@grpss.ssr.upm.es

1017-9909/2016/$25.00 © 2016 SPIE and IS&T

Moreover, traditional inertial sensor-based approaches use linear acceleration measurements to estimate device's velocity and position. They suffer from severe drift caused by the integration of measurement errors. Instead, this system uses only current gravitational acceleration measurements to calculate pose. As no integration is involved, the solution is drift-free.

To evaluate the proposed system, user-in-motion experiments were carried out in this paper: users held a mobile device, walking in the working range, and the proposed system was compared with the reference system OptiTrack,[4] a state-of-the-art commercial motion tracking system providing highly accurate estimations. OptiTrack claims to be able to achieve submillimeter accuracy in marker location estimation with an optimal capture volume size and camera configuration. Experimental results show that our system has achieved six-degree-of-freedom (DoF) pose estimation with a mean error of few centimeters in position estimation and few degrees in orientation estimation.

This paper is an extension of our earlier work presented at ICDSC 2015,[6] where the stereo-vision system with/without occlusion handling was proposed and initially validated by experiments. In this work, each component of the system is comprehensively and thoroughly analyzed. The adaptive thresholding algorithm combined with a Kalman filter for LED tracking is improved and explained in detail. Also, in this work, the performance of the LED tracking algorithm is analyzed and compared to alternative methods by experimental data. In the pose estimation section, we provide a more detailed description of the previously proposed two fusion methods. In addition, the system is extended from a stereo-vision system to a multicamera network in this work. A novel data fusion algorithm based on multiview triangulation is presented. Furthermore, this paper provides more extensive experimental results to evaluate the system quantitatively and qualitatively. It also includes the results of a real-live evaluation in AR and gaming applications.

The remainder of the paper is structured as follows. Section 2 reviews previous research related to pose tracking. Section 3 gives an overview of the entire system. An adaptive thresholding algorithm combined with a Kalman filter for LED tracking is described in Sec. 4. In Sec. 5, three fusion methods for pose estimation are explained. Experimental results and discussions are presented in Sec. 6. Then, Sec. 7 gives two application demonstrations of the proposed approach. Finally, Sec. 8 concludes the paper with future research strategies.

## 2 Related Work

A considerable amount of research has been done on 3-D handheld pose tracking during recent decades. According to the enabling sensing technologies, most of the approaches developed so far can be grouped into four categories: inertial sensing, magnetic field sensing, visual sensing, and hybrid sensing.

### 2.1 *Inertial Sensing*

Inertial sensing approaches use accelerometers and gyroscopes to continuously calculate the position and orientation of a moving object. By integrating both the linear acceleration measured by accelerometers and the angular velocity measured by gyroscopes, the system's current position and orientation is determined. Recent advances in MEMS technologies have made it possible to manufacture small and light inertial sensors and embed them in mobile devices, which facilitate the integration with other techniques. One obvious advantage of inertial tracking is that inertial sensors are embedded and do not rely on external resources. Therefore, they have no line-of-sight requirement and are not influenced by illumination changes as vision-based methods are. Moreover, they are able to track fast and abrupt movements because their data are updated at a high rate. On the downside, they suffer from severe drift problems due to the accumulation of errors in the measurements.[7] Thus, periodic corrections from some other types of measurements are required. For example, in Ref. 8, measurements from a magnetic sensor are used to correct the heading angle.

### 2.2 *Magnetic Field Sensing*

Magnetic field sensing approaches are based on the principle of magnetic induction: when a coiled wire is moved through a magnetic field, an electrical current will flow in the coil. The strength of this current is a function of the distance and the orientation of the coil relative to the source of the magnetic field. A magnetic receiver in the natural magnetic field of the Earth is a 1-DoF tracker, indicating the direction relative to the "magnetic north." To achieve 6-DoF pose estimation, an artificial magnetic field is required, which is typically generated by a transmitter containing three orthogonally orientated coils. Then, the position and orientation of a receiver in this field are deduced based on the induction. Magnetic trackers are lightweight, support multiple sensors, and do not suffer from occlusion. Unfortunately, as is well-known, one problem of magnetic sensing is the sensitivity to magnetic and electrical interference caused by metallic objects within the operating volume. Additionally, it is limited in range due to the decay of the strength with the distance between the emitting source and the sensors.[9] Under laboratory conditions, the position can be estimated with several millimeter-level accuracies for a distance of a couple of centimeters.[10] Sixense Entertainment, a leading company in magnetic motion tracking for video games, has increased the optimized performance range to 8-foot radius from the base with its latest technology.[11]

### 2.3 *Visual Sensing*

Images captured by vision sensors carry a wealth of information, based on which many object pose estimation approaches have been developed. According to the system architecture, approaches can be grouped into outside-in tracking (or infrastructure-based methods), in which fixed external cameras are installed to track the target and inside-out tracking (or target-based tracking), in which the scene is observed by an on-device camera. Outside-in tracking systems are usually composed of external stationary wall- or tripod- mounted cameras directed toward the operating volume. Most commercial optical trackers use this configuration, such as OptiTrack[4] and Vicon.[5] Inside-out tracking systems determine the pose of a device relative to a reference coordinate system through observing the scene using the on-device camera. For example, the HiBall tracking system developed by Welch et al.[12] uses ceiling-mounted LEDs as references to determine the position and orientation of a moving optical sensor. However, it is not practical in use, because it requires

a dense array of LEDs deployed in the ceiling. Inside-out tracking methods are widely adopted for handheld AR. The pose of a handheld device is determined by tracking artificial markers or natural features from the internal camera. Each approach has strengths and limitations in terms of cost, accuracy, scalability, and on-device power consumption. Inside-out tracking is device-centric and is widely used in mobile AR due to its simple setup. On the other hand, the on-board processing is still a great challenge due to the fact that mobile platforms have limited processing power, memory, and battery life. Additionally, certain landmarks or natural features used for tracking have to be in the view of the on-device camera. Outside-in tracking exhibits better scalability to multiple targets than inside-out self-tracking. Additionally, the installation and configuration of outside-in tracking systems is usually faster and easier than the existing inside-out tracking systems.[13]

The objects to be tracked are typically marked with active/passive/printable markers or marker-free. Active markers are markers that emit light themselves, mostly LEDs. They are easily detected and provide visibility up to long distance. In addition, LEDs can be encoded in color and time dimension, which is friendly to multiuser applications. They are used in several systems, including the state-of-the-art commercial motion tracking system ARTrack[14] and Vicon. A similar approach to ours was proposed by Klein and Drummond,[15] which relies on six LEDs mounted on the back of a tablet to track its pose. The exposure of the full back of the tablet in the field of view of the cameras largely constrains the device's movements. In our approach, with the aid of accelerometers, the number of LEDs is reduced to two, which gains more flexibility for users. Another recent LED-based system is described in Ref. 16, in which four or more infrared LEDs are placed at known positions on the target object and are observed by an external camera.

Passive markers are retroreflective elements, i.e., they reflect the incoming infrared radiation. They are used in some of the best known commercial motion tracking systems, such as Vicon and OptiTrack. These systems use multiple stationary calibrated infrared cameras to localize objects attached with passive markers. In general, these systems can provide highly accurate marker location estimation at a high speed. To estimate a plane's orientation, at least three non-collinear markers are demanded to be in the field of view of minimum two cameras. This fact adds more challenges to the system in optimizing the capture volume size and minimizing the expense, as these commercial systems are expensive for regular end users. Moreover, the accuracy of orientation estimation is sensitive to that of marker location estimation due to the fact that orientation is deduced from the geometric configuration of markers.

Printable fiducial markers are maturely developed and widely used in AR applications, such as ARToolKit,[1] ARTag,[2] and AprilTags.[17] Typical markers used in AR are square or circle framed with a unique pattern inside for identification. They are cheap, accurate, easy to use, and easily recognized. On the other hand, they are sensitive to the illumination changes and the accuracy is affected by the distance to the marker and the viewing angle. A novel square-based fiducial system is proposed in Ref. 18, which lowers false negative rates and deals with the occlusion problem. Inside-out tracking configuration is usually adopted for handheld pose estimation, as these markers need a big and flat surface to be place, being unsuitable to be attached to a mobile device. They are widely used in simple scenarios because of the easy setup. However, in case of a wide area, their use can become complicated. First, placing markers in a wide area can be intrusive to the environment. Second, each marker's position has to be measured carefully. In addition, from time to time, they may need recalibration.

Instead of using artificial markers, tracking natural features from the environment, such as points and edges, is becoming an attractive topic of interest for researchers. Robust local descriptors, such as SIFT,[19] SURF,[20] and Ferns[21] are typically used for natural feature tracking. These descriptors are stable under different viewpoints and lighting conditions. Wagner et al.[22] proposed a modified SIFT and Ferns plus a template-matching-based method to run fast natural featuring tracking on mobile phones. However, this approach derives its speedup from tracking relatively fewer features, making it less suitable for continuous 6-DoF tracking in wide areas. Simultaneous localization and mapping (SLAM) systems have been recently developed to run on mobile devices in large-scale areas. For example, Ventura et al. present a key-frame based SLAM system that runs locally on a camera-equipped mobile device in Ref. 23. A feature-less monocular SLAM algorithm is presented in Ref. 24, which allows building large-scale and consistent maps of the environment. Generally speaking, natural feature tracking is still a complex problem and is usually expensive in terms of computational cost, and challenging for handheld devices.

### 2.4 *Hybrid Sensing*

The basic principle of hybrid sensing lies in fusing data from disparate sources, thus overcoming the limitations of each single approach and providing a robust, complete, and accurate solution. Typically, current hybrid systems use inertial sensors as a complement to aid visual sensors. For example, within the tracking methods described in Refs. 25 and 26 gyroscope was adopted to measure orientation, and in return, the vision-based system corrects the drift of the inertial system. Kalman filter and its derivatives (unscented Kalman filter and extended Kalman filter) are favorably selected to perform sensor fusion.[27]

The method proposed in this paper works on a similar concept. Rather than using measurements from gyroscope, we use gravitational acceleration, which avoids the common drift problem. The accelerometer contributes to two rotation angles and the calculation burden is largely reduced.

## 3 System Overview

### 3.1 *System Architecture*

The proposed system is composed of off-the-shelf cameras (two or more), a server and a client mobile device equipped with two LED markers and embedded accelerometers, as shown in Fig. 1. In the test, one red LED (left) and one green LED (right) are placed on the upper corners of the device. Two colors are used to distinguish between left and right. The system includes four modules: fiducial tracking, acceleration measuring, pose tracking, and applications. The flowchart is illustrated in Fig. 2. Fiducial tracking by a stationary and calibrated visual sensor network outputs the

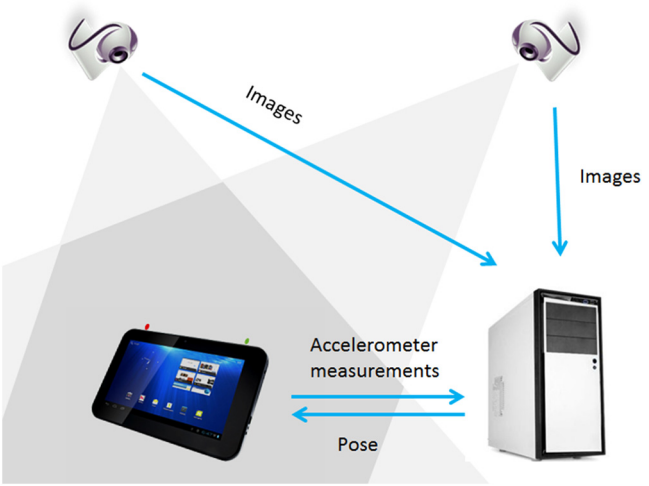


**Fig. 1** System hardware architecture.

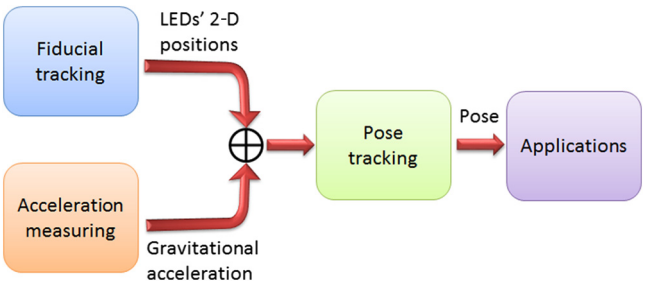


**Fig. 2** System modules and work flow.

LEDs' positions in the images. Acceleration is measured by the embedded three-axis accelerometer and transmitted to the server. Then, the fusion of visual data and inertial sensor data is carried out to estimate the current pose, which is afterward exported as to the client (device) for application usage. The implementations of each module are detailed in the following sections.

### 3.2 *Data Synchronization*

Synchronization among different sensors is required and important in hybrid tracking systems, which is difficult to achieve in practice.[28] In the proposed system, cameras are set to run at about 25 frames/s. Captured images are transmitted to the server either by wireless connection or by cable depending on the types of cameras. Accelerometers run in the client side at around 100 Hz. The communication between the server and the client is designed as wireless via TCP/IP, running in a separate thread. The adopted approximate synchronization approach for real-time operation is described as follows. The first step is to synchronize the clocks on the server and the client by measuring the clock offset using the IEEE 1588 precision time protocol.[29] Once synchronized, the data request message sent to the client is enclosed by the timestamp from the server. Then, the client calculates the local corresponding timestamp at which the message is sent and searches backward for the accelerometer measurements at that timestamp. In this approach, a buffer is necessary to save the accelerometer measurements and the timestamps.

## 4 LED Tracking

In this section, we propose an adaptive thresholding method combined with a Kalman filter for LED tracking. This method is then validated to largely improve the detection rate, reduce the computational cost, and gain robustness against illumination changes in Sec. 6.2.

### 4.1 *Adaptive Thresholding for Light-Emitting Diode Detection*

LED detection is simple in principle, but still faces several challenges. First, the appearance (shape and color) of LEDs in the image varies according to the illumination conditions, the distance between LEDs and cameras and the viewing angle. Furthermore, LEDs are small and cover only a small region in the images. Thus, they can be easily confused with other objects in the environment.

Our method uses the color property to detect LED markers. The first step is to segment the foreground and background. The basic approach using frame difference of the current frame with the previous frame is adopted due to its simplicity and cheapness in computational cost. In this work, a fixed low threshold 10 is selected to make sure LEDs are detected as foreground. Alternatively, other background subtraction approaches may be used.[30] Then, hue, saturation, and value (HSV) color space is adopted for color thresholding since it separates out the intensity (luminance) from the color information (chromaticity).[31] In order to understand the color perception of LEDs under different illumination conditions and different environments, and therefore set proper threshold values, we captured images under different controlled lightings. Then, we manually selected the LED regions in the image and analyze their color properties in HSV space. The observations are represented in the form of histograms in Fig. 3. Note that the $H$ values of both colors are concentrated in a quite narrow range, which is expected because the $H$ (hue) component represents the color tone (e.g., red or blue). On the other hand, $S$ and $V$ values cover a relative wide range. From experimental observations, we found that wrong detections are mostly caused by the lower limits of $S$ and $V$ channels, which lead to confusion by similar colors from the surrounding environments. This observation indicates that color thresholding with fixed values is not suitable for LED detection. Therefore, in the following, we propose an adaptive thresholding method to set lower limits on $S$ and $V$ channels dynamically.

In the following description of the approach, we will not specify colors, as it is the same process for both types of LEDs. The minimum values of $S$ and $V$ in the histogram shown in Fig. 3 are indicated as $S_{\min}$ and $V_{\min}$ (red LED: $S_{\min} = 0.13$, $V_{\min} = 0.30$; green LED: $S_{\min} = 0.11$, $V_{\min} = 0.11$). Let $MS_{k-1}$ and $MV_{k-1}$ be the minimum $S$ and $V$ values of the detected LED marker in the frame $k-1$. At the beginning of the process of the frame $k$, our method first calculates the values of $TS_k$ and $TV_k$, which are the lower threshold values applied to the frame $k$ in $S$ and $V$ channels, through taking the previous detected LED marker into consideration

$$TS_k = \alpha MS_{k-1} + (1-\alpha)S_{\min},$$
$$TV_k = \alpha MV_{k-1} + (1-\alpha)V_{\min}, \quad (1)$$

where $\alpha$ is a weighting value between 0 and 1.

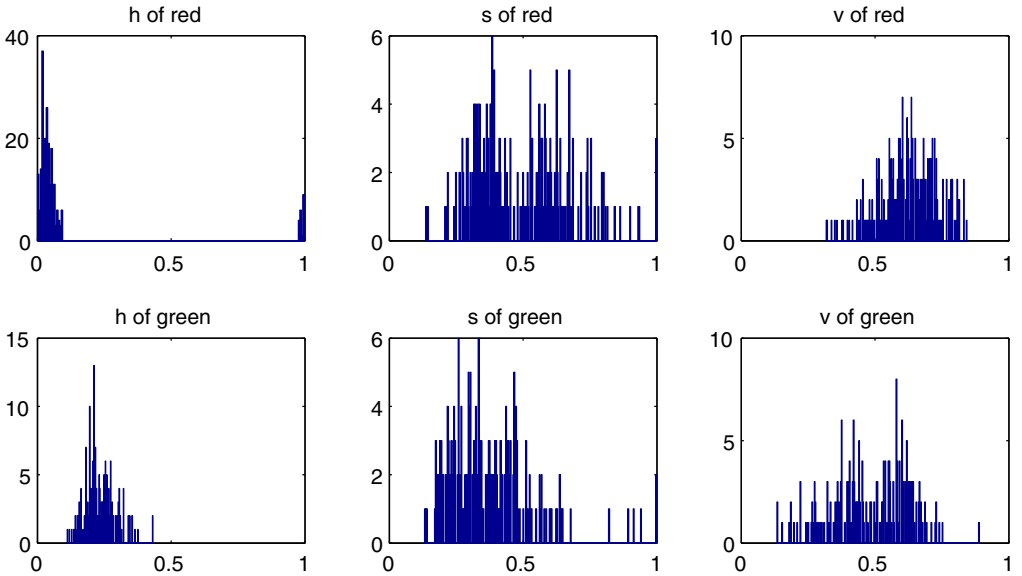


**Fig. 3** *H*, *S*, and *V* histograms of a red LED and a green LED.

The selection of the value of $\alpha$ depends on the actual experimental scenario, especially the lighting conditions, as it indicates to which extent we should believe that the lighting condition does not change during the frame update interval. However, it is difficult to provide an optimal value of $\alpha$ due to the sophisticated lighting condition and its effect on the images. A rough guideline for the selection of $\alpha$ is to set a high value in an almost constant lighting condition, whereas it should be a small value in a varying lighting condition. In our test, by sweeping over different values, we have found that $\alpha = 0.5$ gives a good weight to the previous detection result. After color thresholding, if more than one segment is detected as possible LED region, we then rank them based on the segment centroid's brightness ($V$ value) considering the fact that LEDs emit light. The brightest centroid is considered as the final detection result of the LED marker position in the image.

### 4.2 *Kalman Filter for Light-Emitting Diode Tracking*

A Kalman filter is applied to track LED movements in the image sequences. The purposes of this process are twofold. First, the Kalman filter gives a prediction of the positions where LEDs are supposed to appear in the next frame. Instead of searching the entire image to locate LEDs, only a small area around the predicted position is processed. In this way, the computational cost is reduced, which is one of the crucial criteria for real-time services. Additionally, confusion caused by noise from the surrounding environment is alleviated because the searching region is narrowed.

The state vector of the filter $\mathbf{x}$ is a $8 \times 1$ vector containing the two-dimensional (2-D) position and 2-D velocity of two LEDs in the image, which can be expressed as

$$\mathbf{x} = (\mathbf{p}_1, \dot{\mathbf{p}}_1, \mathbf{p}_2, \dot{\mathbf{p}}_2), \tag{2}$$

where $\mathbf{p}_1$ and $\mathbf{p}_2$ are the positions of the two LEDs in the image.

Therefore, the state transition model relates the state at a previous time instance $k-1$ with the current state at time $k$ as

$$\begin{aligned}\mathbf{x}_k &= F_k \mathbf{x}_{k-1} + w_{k-1} \\ &= \begin{pmatrix} 1 & 0 & \Delta t & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & \Delta t & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & \Delta t & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & \Delta t \end{pmatrix} \mathbf{x}_{k-1} + w_{k-1},\end{aligned} \tag{3}$$

where $\Delta t$ is the time interval between updates and $w_{k-1}$ is the process noise.

The measurement model relates the state vector to the measurement vector $\mathbf{z}_k$ through a transformation matrix $H_k$ as follows:

$$\begin{aligned}\mathbf{z}_k &= \begin{pmatrix} \mathbf{p}_1 \\ \mathbf{p}_2 \end{pmatrix} = H_k \mathbf{x}_k + v_k \\ &= \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} \mathbf{x}_k + v_k,\end{aligned} \tag{4}$$

where $v_k$ is the measurement noise.

The error covariance matrices of the filter are determined following the method proposed in the paper,[32] where the reader can find more detailed explanations. A region of interest (ROI) can be generated based on the predicted positions where the tracked features are expected to appear in the next frame. Therefore, the computational cost is reduced as there is no need to process the entire image. The rectangular ROI is centered at the predicted position of the LED, however, the size of the ROI may be determined in various ways. For example, a fixed size of $32 \times 32$ pixels is chosen in Ref. 33. In Ref. 34, the size of the window is set as three or four times of the size of the previously detected object. In this work, the size of the ROI is determined by the predicted error covariance matrix provided by the filter, as it indicates the uncertainty of the prediction. Taking the tracking error into account, three times of the predicted uncertainty is applied

**Algorithm 1** LED tracking.

1. Get a sub-image from the captured image $I$ based on the specified rectangular ROI
2. Segment the foreground and background based on frame difference
3. Convert the image from RGB color space to HSV color space
4. Adjust the lower limit of the $S$ and the $V$ values
5. Filter the image by color thresholding and save the centroids of the detected contours as candidates
6. Determine the centroid with the highest $V$ value as the final result
7. Update the Kalman filter and generate a rectangular ROI around the predicted position
8. Go back to step 1 for the next frame

to generate a sufficiently large window to make sure the tracked object is inside the selected window.

The complete LED tracking process is summarized in Algorithm 1.

## 5 Pose Estimation Algorithms

In this section, we will present three fusion algorithms for six-DoF pose calculation combining visual and inertial measurements. We will first describe the coordinate systems related to the system in Sec. 5.1. Next, the contribution of the accelerometers to pitch and roll rotation angles estimation is explained in Sec. 5.2. Then, the stereoscopic solution based on triangulation is explained in Sec. 5.3. A constraint-based algorithm for occlusion handling is described in Sec. 5.4. Finally, a multiview solution by triangulation in a multicamera network is presented in Sec. 5.5.

### 5.1 *Notation*

Pose can be mathematically represented as a $3 \times 4$ transformation matrix from a certain coordinate system to the final coordinate system; it is composed of a rotation matrix and a translation vector. The coordinate systems used in the remaining of this paper are defined as follows:

- World coordinate system $\{w\}$: the $Z$-axis points toward the sky and is perpendicular to the ground plane. The $X$ axis and $Y$ axis are in the horizontal plane and defined following the right-hand rule. A point in $\{w\}$ is denoted by $\mathbf{P} = (X, Y, Z)^T$.
- Accelerometer coordinate system $\{a\}$: when the device is held in its default orientation, the $X$ axis is horizontal and points to the right, the $Y$ axis is vertical and points up, and the $Z$ axis points toward the outside of the screen face. A measurement of gravitational acceleration is denoted as $\mathbf{g}_a = (g_x, g_y, g_z)^T$.
- Camera coordinate system $\{c\}$: we use a standard camera coordinate system of which the origin is at the center of projection and the $Z$ axis is along the optical axis. The $X$ and $Y$ axes are parallel to the image plane.
- Camera frame coordinate system $\{f\}$: a point in the camera plane is expressed in pixels as $p = (u, v)$.

$$R = R_x(\psi)R_y(\theta)R_z(\phi) = \begin{bmatrix} \cos\theta\cos\phi & \cos\theta\sin\phi & -\sin\theta \\ \sin\psi\sin\theta\cos\phi - \cos\psi\sin\phi & \sin\psi\sin\theta\sin\phi + \cos\psi\cos\phi & \sin\psi\cos\theta \\ \cos\psi\sin\theta\cos\phi + \sin\psi\sin\phi & \cos\psi\sin\theta\sin\phi - \sin\psi\cos\phi & \cos\psi\cos\theta \end{bmatrix}, \tag{5}$$

where $R_i(\alpha)$ is a basic 3-D rotation matrix around the $i$ axis ($i = x, y$ or $z$; $\alpha = \psi, \theta$ or $\phi$).

The rotation matrix $R$ converts a vector expressed in $\{w\}$ $\mathbf{v}_w$ to the expression in $\{a\}$ $\mathbf{v}_a$ as

$$\mathbf{v}_a = R\mathbf{v}_w. \tag{6}$$

The gravitational vector is expressed in $\{w\}$ as $\mathbf{g}_w = [0,0,-g]^T$ (where $g$ is the magnitude of gravity) and is expressed in $\{a\}$ as $\mathbf{g}_a = [g_x, g_y, g_z]^T$. Applying the rotation matrix to the gravitational vector gives:

$$\mathbf{g}_a = \begin{bmatrix} g_x \\ g_y \\ g_z \end{bmatrix} = R\mathbf{g}_w = R\begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix} = \begin{bmatrix} g\sin\theta \\ -g\sin\psi\cos\theta \\ -g\cos\psi\cos\theta \end{bmatrix}. \tag{7}$$

Then, from Eq. (7), the roll and pitch rotation angles are calculated as

### 5.2 *Pitch and Roll Estimation Using Accelerometers*

As is known that accelerometers sense both gravitational and dynamic (movement induced) acceleration forces, in this paper, only gravitational accelerations are isolated to contribute to the orientation estimation. The isolation can be achieved by a low-pass filter. This topic is further studied in literature.[35] In case of Android mobile devices, a "gravity sensor" is embedded since API Level 9 (Android 2.3) was released. Thus, for this work we will not go further into details about the isolation of gravitational accelerations.

The orientation of a handheld device can be defined by a sequence of three elemental rotations, i.e., pitch (rotation about the $X$ axis), roll (rotation about the $Y$ axis), and yaw (rotation about the $Z$ axis) from an initial status. The composite rotation matrix $R$ depends on the order in which the pitch, roll, and yaw rotations are applied. There are six possible orderings of the three angles. In principle, all these are equally valid. However, four of these rotation sequences are rejected as they are unsuitable for determining the device's inclination from accelerometers.[36] It is conventional therefore to select either the rotation sequence yaw-roll-pitch or yaw-pitch-roll to allow solution for the roll and pitch angles from acceleration measurements. In this work, we choose the yaw-roll-pitch order. Therefore, the rotation matrix $R$ can be determined as

$$\theta = \arcsin\frac{g_x}{g}, \tag{8}$$

$$\psi = \arctan\frac{g_y}{g_z}. \tag{9}$$

It is worth noting that accelerometers are insensitive to rotation about the earth's gravitational field vector. Therefore, accelerometers are not sufficient to estimate the yaw rotation angle.

### 5.3 *Triangulation-Based Stereo-Vision System*

#### 5.3.1 *Position estimation*

The LEDs' positions in the images are obtained following the steps described in Sec. 4. In case that both LEDs are detected in both camera views, the linear triangulation method[37] is applied for the 3-D reconstruction of LEDs' positions in $\{w\}$. Let $\mathbf{P}_l = (X_l, Y_l, Z_l)^T$ denote the 3-D position of the left LED and let $\mathbf{P}_r = (X_r, Y_r, Z_r)^T$ denote the 3-D position of the right LED in $\{w\}$. The position of the device $\mathbf{P}$ is considered as the center of the two LEDs.

Since the 3-D distance between two LEDs is fixed as $D$ once they are deployed, a further check can be carried out by testing whether the distance between two estimated LEDs $\hat{D}$ meets the condition $\|\hat{D} - D\| < T$, where $T$ is the threshold. In this work, the value of $T$ is selected as $3\sigma$, where $\sigma$ is the standard deviation of the calculated distances in a 5-min running test (in our test, $D = 18.3$ cm, $\sigma = 1.23$ cm). This test provides a high degree of reliability to the final results.

#### 5.3.2 *Yaw estimation*

As is aforementioned, accelerometers are insufficient to estimate the yaw rotation angle. In this work, it is determined by the estimated 3-D positions of the LEDs. Let $\Delta\mathbf{P}_w$ and $\Delta\mathbf{P}_a$ denote the vector from the left LED to the right LED expressed in $\{w\}$ and in $\{a\}$, respectively. Applying the rotation matrix to the two vectors following Eq. (6) gives

$$\Delta\mathbf{P}_a = R\Delta\mathbf{P}_w. \tag{10}$$

According to the definition of $\{a\}$, $\Delta\mathbf{P}_a$ is known as

$$\Delta\mathbf{P}_a = [D \quad 0 \quad 0]^T. \tag{11}$$

The property of the rotation matrix determines that its transpose is identical to its inverse. Therefore, Eq. (10) can be rewritten as

$$\Delta\mathbf{P}_w = \mathbf{P}_r - \mathbf{P}_l = R^T\Delta\mathbf{P}_a = \begin{bmatrix} D\cos\theta\cos\phi \\ D\cos\theta\sin\phi \\ -D\sin\theta \end{bmatrix}. \tag{12}$$

Then, the roll and yaw rotation angles are obtained as

$$\theta = -\arcsin\frac{Z_r - Z_l}{D}, \tag{13}$$

$$\phi = \arctan\frac{Y_r - Y_l}{X_r - X_l}. \tag{14}$$

The roll angle can be calculated through both vision data and accelerometer data. Considering there may be wrong detections in the vision task, the roll angle is calculated by Eq. (13).

### 5.4 *Constraint-Based Stereo-Vision System with Occlusion Handling*

In this section, we propose a new fusion algorithm to determine the 6-DoF pose in case of partial occlusions, which refers to the situations when one LED marker is located in the view of one single camera. Let us start with the case when the right LED marker is seen by one camera and the left LED is seen by both cameras. The 3-D position of the left LED marker $\mathbf{P}_l$ is obtained by triangulation as is explained in the previous section. We denote by $p_r = (u_r, v_r)$ the position of the right LED marker in the detected camera view. The geometry of the occlusion handling system is depicted in Fig. 4.

We mentioned in the previous subsection that the roll rotation angle can be calculated either by Eq. (8) or by Eq. (13). Therefore, combining the two equations, a constraint on the gravitational acceleration and the height difference between two LEDs can be obtained as

$$\frac{g_x}{g} = -\frac{Z_r - Z_l}{D}. \tag{15}$$

Thus, the element $Z_r$ of $\mathbf{P}_r$ can be calculated as

$$Z_r = Z_l - Dg_x/g. \tag{16}$$

This constraint is essential in the proposed occlusion handling approach to recover the position of the partially occluded LED. As is well-known, in a pin-hole camera model, a 2-D point in the image is corresponded to a ray of 3-D points that connects the camera projection center and the 2-D point, as depicted in Fig. 4 (the blue line). According to the camera projection model, the relationship between the LED's 3-D position $\mathbf{P}_r$ in $\{w\}$ and its 2-D position $p_r$ in the image can be expressed as

$$\lambda\begin{bmatrix} u_r \\ v_r \\ 1 \end{bmatrix} = M\begin{bmatrix} \mathbf{P}_r \\ 1 \end{bmatrix} = M\begin{bmatrix} X_r \\ Y_r \\ Z_r \\ 1 \end{bmatrix}, \tag{17}$$

where $\lambda$ is a scale factor and $M$ is a $3\times4$ transformation matrix.

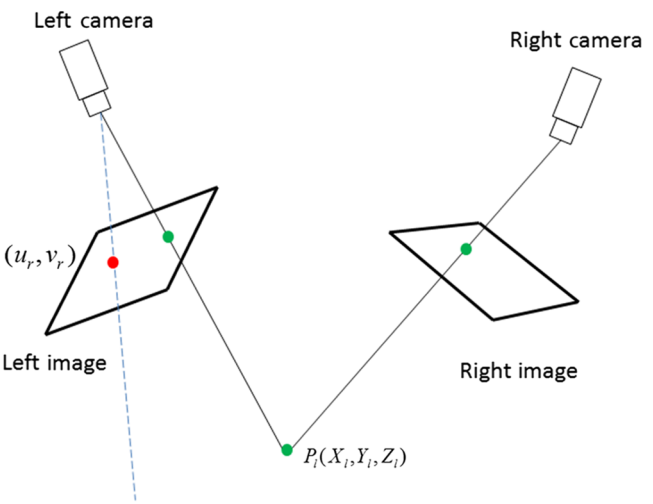


**Fig. 4** The geometry of the occlusion handling system.

The matrix $M$ indicates the transformation between $\{w\}$ and $\{c\}$, which is obtained by camera calibration. Therefore, in Eq. (17), there are three unknown parameters ($X_r$, $Y_r$, and $\lambda$) and three equations. Thus, the 3-D position of the right LED can be calculated by this determined system. Consequently, the position and the orientation of the device are calculated in the same way as explained in the previous section. The same process is done in case that the left LED marker is partially occluded.

### 5.5 *Triangulation-Based Multivision System*

In a multicamera system, a marker may be viewed by $N$ ($N > 2$) cameras, several approaches can be used for multiview triangulation. In this work, the triangulation based on the least squares minimization similar to the solution to the stereo-vision triangulation is adopted. Applying Eq. (17) to a general pair of 3-D–2-D corresponding point leads to

$$\lambda\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = M\begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \begin{bmatrix} M(1,1) & M(1,2) & M(1,3) & M(1,4) \\ M(2,1) & M(2,2) & M(2,3) & M(2,4) \\ M(3,1) & M(3,2) & M(3,3) & M(3,4) \end{bmatrix}\begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}, \tag{18}$$

where $\lambda$ is the scale factor, $(u, v)$ is the 2-D point in the image, and $(X, Y, Z)$ is the corresponding 3-D point.

Eliminating $\lambda$ leads to two linear equations

$$\begin{aligned} &[M(3,1)X + M(3,2)Y + M(3,3)Z + M(3,4)]u \\ &\quad = M(1,1)X + M(1,2)Y + M(1,3)Z + M(1,4) \\ &[M(3,1)X + M(3,2)Y + M(3,3)Z + M(3,4)]v \\ &\quad = M(2,1)X + M(2,2)Y + M(2,3)Z + M(2,4). \end{aligned} \tag{19}$$

When the marker is viewed by $N$ cameras, the following over-determined linear equations system can be obtained

$$A\mathbf{P} = A[X, Y, Z]^T = b, \tag{20}$$

where $A$ is a $2N \times 3$ matrix and $b$ is a $2N \times 1$ matrix, described next:

$$A = \begin{bmatrix} M_1(1,1) - u_1M_1(3,1) & M_1(1,2) - u_1M_1(3,2) & M_1(1,3) - u_1M_1(3,3) \\ M_1(2,1) - v_1M_1(3,1) & M_1(2,2) - v_1M_1(3,2) & M_1(2,3) - v_1M_1(3,3) \\ \vdots & \vdots & \vdots \\ M_N(1,1) - u_NM_N(3,1) & M_N(1,2) - u_NM_N(3,2) & M_N(1,3) - u_NM_N(3,3) \\ M_N(2,1) - v_NM_N(3,1) & M_N(2,2) - v_NM_N(3,2) & M_N(2,3) - v_NM_N(3,3) \end{bmatrix}, \quad b = \begin{bmatrix} u_1M_1(3,4) - M_1(1,4) \\ v_1M_1(3,4) - M_1(2,4) \\ \vdots \\ u_NM_N(3,4) - M_N(1,4) \\ v_NM_N(3,4) - M_N(2,4) \end{bmatrix}, \tag{21}$$

where $M_i$ is the $3 \times 4$ transformation matrix of the $i$'th camera, $(u_i, v_i)$ is the detected LED in the image from the $i$'th camera, $i = 1,2 \ldots N$.

Afterward, we do the same as in the stereo-vision system to solve the equations using a least squares approach minimizing $\|A\mathbf{P}_w - b\|$ and get

$$\mathbf{P}_w = A^+b, \tag{22}$$

where $A^+$ is the pseudoinverse matrix of $A$.

Once the 3-D positions of both LEDs are obtained, the position and rotation of the device can be calculated as explained in the previous section.

## 6 Evaluation

In this section, the LED tracking module and the pose estimation module of the proposed system are quantitatively evaluated. First, the prototype and the reference system setup as well as the offline data synchronization are explained in Sec. 6.1. Then, the aforementioned adaptive thresholding approach combined with a Kalman filter for LED tracking is compared with three other methods in Sec. 6.2. The performance of the stereo-vision system with/without occlusion handling is assessed in Sec. 6.3. Finally, the overall performance of the proposed system integrated with several fusion algorithms is analyzed in Sec. 6.4.

### 6.1 *Experimental Setup*

#### 6.1.1 *Prototype setup*

The proposed system involves four RGB cameras deployed in the four corners of the ceiling, as shown in Fig. 5(a). They are calibrated beforehand following the calibration method based on the POSIT algorithm.[38] The image resolution is set at $780 \times 580$ pixels, running at 25 frames/s. A tablet PC Samsung Galaxy Tab S 10.4 is used as the hand-held device. Two battery-powered LEDs with diameter of 5 mm are placed on the border of the tablet with an interval of 18.3 cm, as illustrated in Fig. 5(b). Gravity measurements are directly obtained from the gravity sensor.

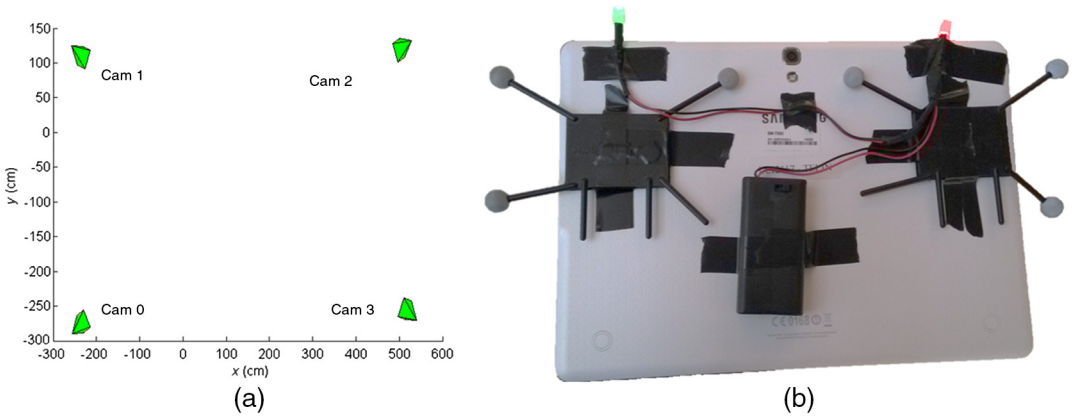


**Fig. 5** Experimental setup. (a) The deployment of four cameras. (b) The deployment of battery-powered LEDs and OptiTrack markers. Six reflective markers are used in the OptiTrack system.

### 6.1.2 *Reference system setup*

The reference system OptiTrack is composed of eleven infrared cameras, covering a scene of 7.5 m × 4.0 m. The cameras are calibrated using their own calibration tools. Six reflective markers with diameter of 1.1 cm are placed on the back of the tablet, as shown in Fig. 5(b). These markers are selected as a rigid body. The pose estimation results are relative to the rigid body coordinate system. It has the same axes as the accelerometer coordinate system, but with an offset from the center of the two LEDs, which is corrected during comparison. The OptiTrack system works at 120 Hz.

### 6.1.3 *Experimental data synchronization*

For a fair comparison of the algorithms, we make sure they work on comparable and synchronized data. Therefore, the evaluation is done offline with image sequences recorded at 25 frames/s and the accelerometer data recorded at 100 Hz.

Data synchronization among the gravity sensor, image sequences, and OptiTrack results are necessary to compare our results to the OptiTrack results frame by frame. To enable synchronization, at the beginning of the test, the user holds the tablet horizontally and moves it up and down several times. Then, the $Y$ element of gravity sensor measurements and the roll angle estimations from OptiTrack are plotted in Fig. 6. As we can see, the shaking movements are represented in the figure as rise and fall. We select the last local minimum as the starting point for the gravity sensor and the OptiTrack. The corresponding physical meaning is that the tablet is placed at the local lowest position which could be easily searched from the video. As the data is recorded at a constant frequency, correspondences can be easily found by sampling with a known starting point.

## 6.2 *Light-Emitting Diode Tracking Performance Analysis*

The performance of the proposed adaptive thresholding approach combined with a Kalman filter is analyzed in terms of correct LED detection rate and average processing time. Our method is compared with three alternative approaches, which are fixed thresholding, fixed thresholding with a Kalman filter, and adaptive thresholding. In the fixed thresholding method, the minimum and maximum values of $H$, $S$, and $V$ from the histogram in Fig. 3 are used as threshold values.

In the test, a user held the tablet and walked around in the scene from one corner to another corner. In total, 430 frames were recorded by the camera 0 and camera 1 shown in Fig. 5(a). Each algorithm is tested on two video recordings. The detected red and green LEDs' positions are saved in a file. The ground truth of LEDs' positions in the images is manually generated in MATLAB®. If the difference between

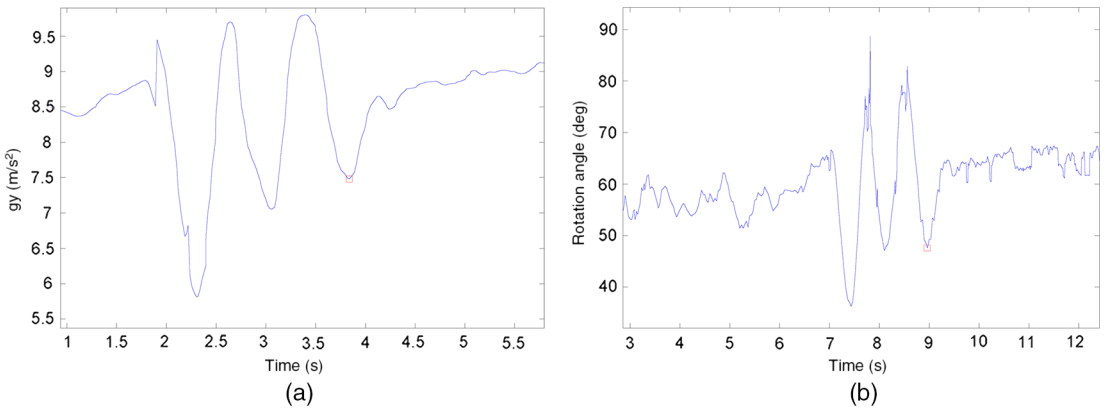


**Fig. 6** Measurements from a gravity sensor and OptiTrack. The red squares in the figures indicate the starting point. (a) $Y$ elements of measurements from a gravity sensor. (b) Roll angle estimations from OptiTrack.

**Table 1** Comparison of different LED tracking approaches.

| | LED correct detection rate | | | | Average processing time (s) | |
|---|---|---|---|---|---|---|
| | Left cam | | Right cam | | | |
| | Red | Green | Red | Green | Left cam | Right cam |
| Fixed thresholding | 54.88% | 79.30% | 87.8% | 89.6% | 0.20 | 0.19 |
| Fixed thresholding + tracking | 70.23% | 92.56% | 88.0% | 90.6% | 0.04 | 0.04 |
| Adaptive thresholding | 94.19% | 94.19% | 96.9% | 96.3% | 0.19 | 0.20 |
| Adaptive thresholding + tracking | 94.88% | 96.05% | 96.9% | 96.3% | 0.04 | 0.05 |


**Fig. 7** Camera views: (a) left camera view and (b) right camera view.

the detected pixel position and the ground truth position is within three pixels, we consider it as a correct detection. The results are listed in Table 1.

The results validate that the proposed adaptive thresholding method combined with a Kalman filter largely improves the correct detection rate and reduces the processing time, compared with the fixed thresholding method. It is also noticeable that the LED detection rates are improved by 40% and 16.75% through using the proposed method compared to the fixed thresholding method for the process of the recording from the left camera. Let us check the image sources from the cameras. Figure 7 demonstrates two examples of camera views from both cameras in the test. It can be seen that the view from the right camera is towards a relative closed space, which means the illumination change is small. The capture area of the left camera is exposed to artificial light sources, natural light sources and a strobe light from one OptiTrack camera located around the top center of the image. All these light sources introduce challenges to LED detection, which can explain the low LED detection rate in the left camera recording by using the fixed thresholding method. However, our proposed approach is still able to achieve a high detection rate in the varying illumination conditions.

### 6.3 *Stereo-Vision System Performance Analysis*

This section is focused on the assessment of the proposed stereo-vision system with/without occlusion handling. The same experimental data from the LED tracking performance analysis is exploited to evaluate the performance of the stereo-vision system in terms of estimation accuracy and LED detection rate. The accuracy assessment is carried out by comparing with the reference system OptiTrack in terms of 3-D position and orientation estimation. The detection rate is defined as the percentage of the frames with good pose estimation among the entire recording sequences. The position and orientation errors are defined as the square root of the error in each single axis.

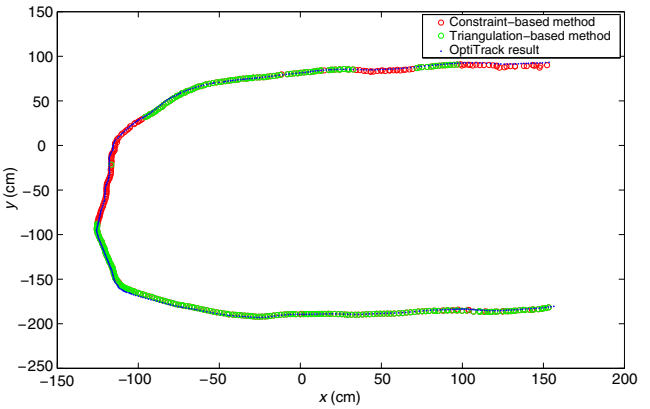


**Fig. 8** Estimated 2-D trajectories from the triangulation-based method (green dots), constraint-based method (pink dots), and OptiTrack (blue dots).

**Table 2** Comparison of pose estimation algorithms.

| | Triangulation-based | Constraint-based | Overall stereo-vision solution |
|---|---|---|---|
| Mean position error (cm) | 1.91 | 1.93 | 1.91 |
| Standard deviation (cm) | 0.80 | 0.90 | 0.83 |
| Min position error (cm) | 0.23 | 0.40 | 0.23 |
| Max position error (cm) | 3.97 | 6.75 | 6.75 |
| Mean orientation error (deg) | 3.26 | 4.41 | 3.60 |
| Standard deviation (deg) | 1.55 | 3.53 | 2.41 |
| Min orientation error (deg) | 0.34 | 0.35 | 0.34 |
| Max orientation error (deg) | 8.49 | 17.18 | 17.18 |
| Detection rate | 67.21% | 30.23% | 97.44% |

The resulting 2-D trajectories from the two methods and OptiTrack are depicted in Fig. 8, where it can be observed intuitively that trajectories from both single methods are close to the reference.

The quantitative comparison results of the two methods and the overall stereo-vision solution from the 430-frame test sequence are listed in Table 2. It shows that both single methods have small errors. The overall stereo-vision solution has achieved an accuracy of 1.91 cm in position estimation and 3.60 deg in orientation estimation with a high-detection rate of 97.44%. It is also observed that the constraint-based method for occlusion handling has a bit larger errors and standard deviation in the estimations, compared with the triangulation-based method. This is expected because the occlusion handling method is based on the equivalent relationship between the inertial measurements and LEDs' positions, as expressed in Eq. (15), leading to a high sensitivity of LEDs' 3-D reconstruction to inertial measurements. However, in the triangulation-based method, LEDs' 3-D positions are obtained from pure visual data, independent from inertial data.

The errors of the proposed overall stereo-vision system compared with OptiTrack in terms of position and rotation angles in three dimensions are depicted in Fig. 9. It is obvious that yaw rotation angle estimations are noisier than pitch and roll angle estimations. This is because yaw angles are calculated from markers' 3-D positions. A small error in the marker position estimation will lead to a big error in the angle estimation, if the interval between markers is small. This is a common problem existing in most of the marker-based tracking systems. Therefore, it is recommended

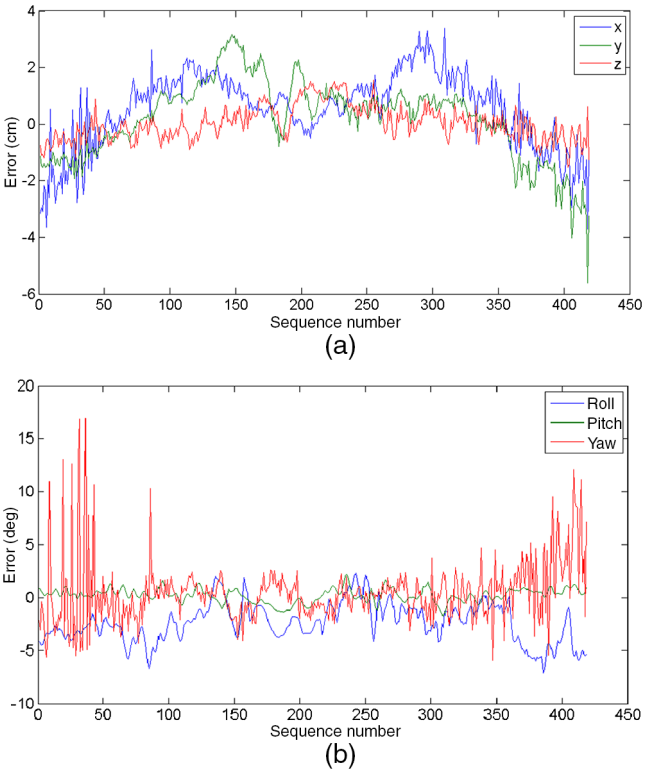


**Fig. 9** Pose estimation error: (a) position estimation error and (b) rotation estimation error.

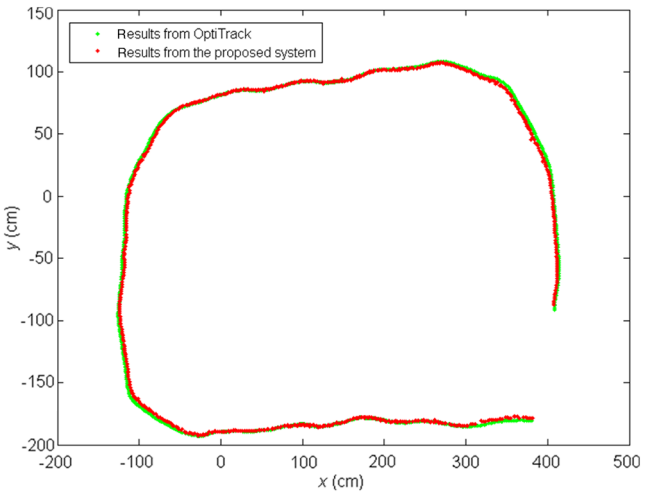


**Fig. 10** Estimated 2-D trajectories from OptiTrack (green dots) and the proposed overall system (red dots) from the first sequence.

to place the markers at as large as possible interval in the running device.

### 6.4 *Overall Performance Analysis*

Our overall handheld pose estimation solution in a multicamera network is the integration of the three proposed data fusion approaches. The performance of the overall solution is assessed in terms of estimation accuracy by comparing with the results generated by OptiTrack. In the experiment, four cameras are involved and the recording procedure is the same as previously mentioned. Testers held the tablet and

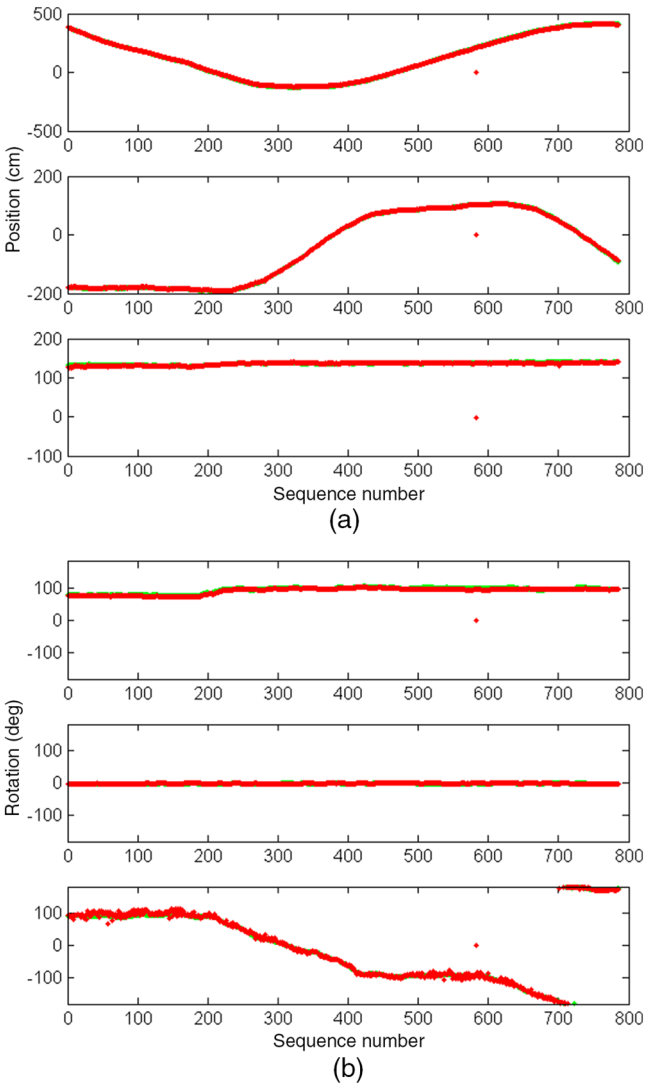


**Fig. 11** Estimation results from OptiTrack (green dots) and the proposed overall system (red dots). (a) Position estimation in $X$, $Y$, and $Z$ axes. (b) Rotation estimation about $X$, $Y$, and $Z$ axes.

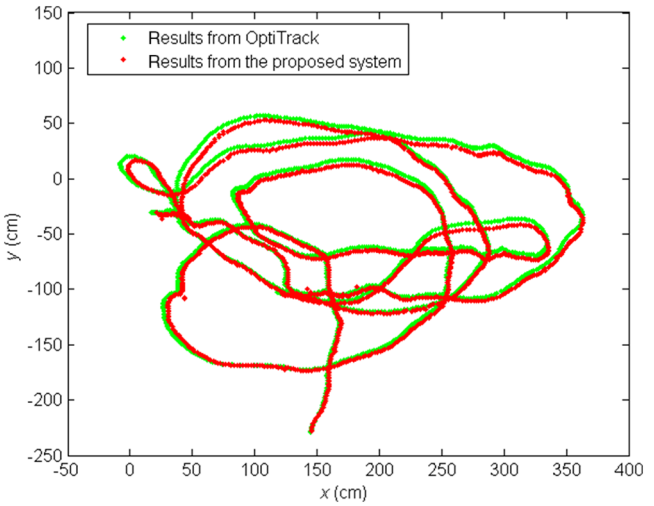


**Fig. 12** Estimated 2-D trajectories from OptiTrack (green dots) and the proposed overall system (red dots) from the second sequence.

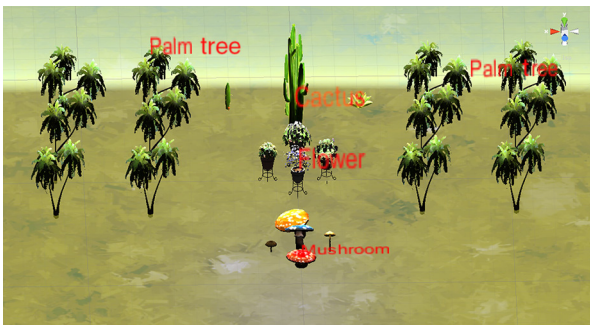


**Fig. 13** A screen shot of an AR application for education.

walked around in a 7.5 m × 4.0 m room. Two sequences were recorded.

In the first sequence, the tester walked in a rectangular trajectory, and, in total, 785 frames were recorded. The 2-D trajectories generated by OptiTrack and the proposed system are plotted in Fig. 10. The position and rotation estimation in each dimension are illustrated in Fig. 11. It can be seen that it has achieved a high detection rate (99.8%). The mean errors of the position and rotation estimation from this experiment are $3.08 \pm 1.40$ cm and $4.82 \pm 3.52$ deg, respectively.

In the second sequence, the tester walked in an irregular pattern with varying lighting condition (manually controlled). In total, 1800 frames were recorded. The 2-D trajectories generated by OptiTrack and the proposed system are plotted in Fig. 12. A high-detection rate of 99.8% has been achieved (three losses of detection). The mean errors of the position and rotation estimation from this sequence are $4.49 \pm 1.69$ cm and $5.51 \pm 3.31$ deg, respectively.

## 7 Applications

The proposed six-DoF pose tracking approach for hand-held devices is a general solution which can be applied to various application fields, including AR, VR, and interactions in smart spaces. In this section, we include two demonstrations of the proposed approach. One is to apply it to an AR application for education in a slightly textured scene, which is quite challenging for markerless approaches based on scene texture. The other demonstration is to use the hand-held device as a 3-D game controller which was presented in ICDSC 2015 Demo Session.[39]

### 7.1 *Augmented Reality*

Handheld AR has become increasingly attractive in recent years. It is applied in various fields, such as smart spaces, tourism, entertainment, training and education.[40] Computer generated content (e.g., text, videos, images) is overlaid on the real image to provide context aware data, enable the manipulation of the displayed content, or facilitate the interaction with the physical world. One of the crucial challenges is the registration of the virtual space with the physical space, referred to the need of accurate tracking of the hand-held device pose with respect to a certain coordinate system. We implement an application for immersive learning based on the proposed system. A virtual 3-D botanical garden with various plants is created on top of a slightly-textured table. By moving a hand-held device, users are allowed to navigate in the garden, observing the plants' features from different views. Also, each plant is labeled with its name for education

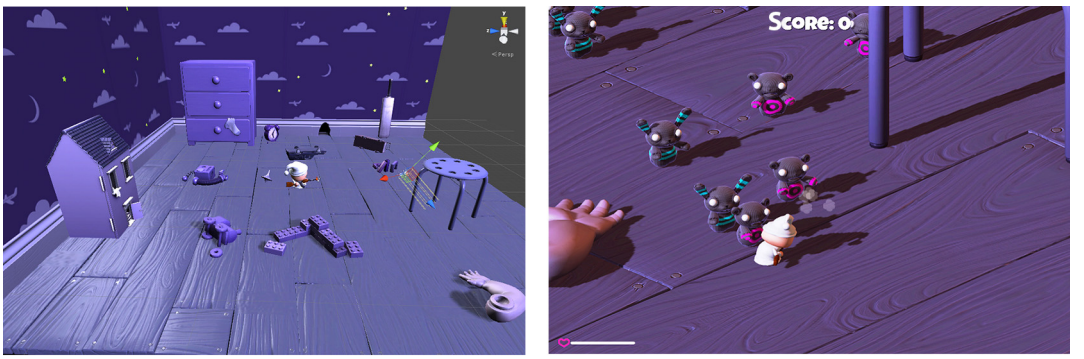


**Fig. 14** Game scenario: (a) the environment setup and (b) a screen shot of the game.

usage. Figure 13 shows a capture of the view from the mobile device.

### 7.2 *Game Controller*

A mobile device is allowed to work as a six-DoF game controller with the proposed approach, providing an immersive and 360-deg gaming experience. The 3-D game is designed using Unity (Ref. 41), working with two or more cameras. The estimated position of the hand-held device relative to a reference coordinate system is used to control the movements of the player in the game. The yaw rotation angle is applied to control the gun direction, while the pitch angle is designed as a shooting trigger. Compared with conventional game controllers sensing button-presses, the proposed approach allows a more immersive gaming experience. Additionally, the display screen of a hand-held device can be used to display the game scenario and provide more possibilities for multimedia elements integration. The game environment and a screen shot of the game are shown in Fig. 14.

## 8 Conclusions

This paper describes a low-cost, accurate, fast, and robust handheld pose estimation system by combining LED marker tracking from two or more cameras and inertial measurements. The experimental results validate its high accuracy and robustness against the illumination changes and partial occlusions. Moreover, two application examples of the proposed solution are implemented to validate its feasibility for application usage.

As we can see from the experimental results, the raw estimation results from the proposed system are noisy. Therefore, future work will be done on data filtering. Additionally, the adaptation of the proposed system to work for multiple users will be taken into consideration.

### *Acknowledgments*

This work has been supported by the Spanish Ministry of Economy and Competitiveness under grant TEC2014-55146-R, the Technical University of Madrid under grant RP150955017, the China Scholarship Council and Ghent University/iMinds.

### *References*

1. H. Kato and and M. Billinghurst, "Marker tracking and HMD calibration for a video-based augmented reality conferencing system," in *Augmented Reality, 1999 Proc. 2nd IEEE and ACM Int. Workshop on IEEE (IWAR'99)*, (1999).
2. M. Fiala, "ARTag, a fiducial marker system using digital techniques," in *IEEE Computer Society Conf. on Computer Vision and Pattern Recognition (CVPR '05)*, Vol. 2, pp. 590–596, IEEE (2005).
3. J.-F. Wang et al., "Tracking a head-mounted display in a room-sized environment with head-mounted cameras," *Proc. SPIE* **1290**, 47 (1990).
4. N. Point, "Optitrack," https://www.optitrack.com/ (15 May 2016).
5. Vicon, "Vicon motion capture system," http://www.vicon.com/ (15 May 2016).
6. J. Li et al., "A hybrid pose tracking approach for handheld augmented reality," in *Proc. 9th Int. Conf. on Distributed Smart Camera*, pp. 7–12, ACM (2015).
7. O. J. Woodman, "An introduction to inertial navigation," Technical Report UCAMCL-TR-696, Vol. **14**, p. 15, University of Cambridge, Computer Laboratory, Cambridge, UK (2007).
8. E. Foxlin, "Pedestrian tracking with shoe-mounted inertial sensors," *IEEE Comput. Graphics Appl.* **25**(6), 38–46 (2005).
9. A. M. Franz et al., "Electromagnetic tracking in medicine—a review of technology, validation, and applications," *IEEE Trans. Med. Imaging* **33**(8), 1702–1725 (2014).
10. E. Wilson et al., "A hardware and software protocol for the evaluation of electromagnetic tracker accuracy in the clinical environment: a multicenter study," *Proc. SPIE* **6509**, 65092T (2007).
11. I. Khalfin and A. Rubin, "Electromagnetic tracker (ac) with extended range and distortion compensation capabilities employing multiple transmitters," US Patent App. 13/796, 210 (2013).
12. G. Welch et al., "High-performance wide-area optical tracking: the hiball tracking system," *Presence Teleoperators Virtual Environ.* **10**(1), 1–21 (2001).
13. T. Pintaric and H. Kaufmann, "Affordable infrared-optical pose-tracking for virtual and augmented reality," in *Proc. Trends and Issues in Tracking for Virtual Environments Workshop, IEEE VR*, pp. 44–51 (2007).
14. ART, "Art," http://www.ar-tracking.com/home/ (15 May 2016).
15. G. Klein and T. Drummond, "Sensor fusion and occlusion refinement for tablet-based AR," in *Third IEEE and ACM Int. Symp. on Mixed and Augmented Reality (ISMAR '04)*, pp. 38–47, IEEE (2004).
16. M. Faessler et al., "A monocular pose estimation system based on infrared LEDs," in *IEEE Int. Conf. on Robotics and Automation (ICRA '14)*, pp. 907–913, IEEE (2014).
17. E. Olson, "Apriltag: a robust and flexible visual fiducial system," in *IEEE Int. Conf. on Robotics and Automation (ICRA '11)*, pp. 3400–3407, IEEE (2011).
18. S. Garrido-Jurado et al., "Automatic generation and detection of highly reliable fiducial markers under occlusion," *Pattern Recognit.* **47**(6), 2280–2292 (2014).
19. D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vision* **60**(2), 91–110 (2004).
20. H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: speeded up robust features," in *Computer Vision–ECCV 2006*, pp. 404–417, Springer (2006).
21. M. Ozuysal et al., "Fast keypoint recognition using random ferns," *IEEE Trans. Pattern Anal. Mach. Intell.* **32**(3), 448–461 (2010).
22. D. Wagner et al., "Real-time detection and tracking for augmented reality on mobile phones," *IEEE Trans. Visual Comput. Graphics* **16**(3), 355–368 (2010).
23. J. Ventura et al., "Global localization from monocular slam on a mobile phone," *IEEE Trans. Visual Comput. Graphics* **20**(4), 531–539 (2014).
24. J. Engel, T. Schöps, and D. Cremers, "LSD-SLAM: large-scale direct monocular slam," in *Computer Vision–ECCV 2014*, pp. 834–849, Springer (2014).
25. K. Satoh, S. Uchiyama, and H. Yamamoto, "A head tracking method using bird's-eye view camera and gyroscope," in *Proc. 3rd IEEE/ACM Int. Symp. on Mixed and Augmented Reality*, pp. 202–211, IEEE Computer Society (2004).

26. G. Reitmayr and T. W. Drummond, "Going out: robust model-based tracking for outdoor augmented reality," in *IEEE/ACM Int. Symp. on Mixed and Augmented Reality (ISMAR '06)*, pp. 109–118, IEEE (2006).
27. G. Ligorio and A. M. Sabatini, "Extended Kalman filter-based methods for pose estimation using visual, inertial and magnetic sensors: comparative analysis and performance evaluation," *Sensors* **13**(2), 1919–1941 (2013).
28. B. Jiang, U. Neumann, and S. You, "A robust hybrid tracking system for outdoor augmented reality," in *IEEE Virtual Reality 2004*, p. 3, IEEE (2004).
29. J. Kannisto et al., "Software and hardware prototypes of the IEEE 1588 precision time protocol on wireless LAN," in *14th IEEE Workshop on Local and Metropolitan Area Networks (LANMAN '05)*, p. 6, IEEE (2005).
30. M. Piccardi, "Background subtraction techniques: a review," in *IEEE Int. Conf. on Systems, Man and Cybernetics*, Vol. 4, pp. 3099–3104, IEEE (2004).
31. S. Sural, G. Qian, and S. Pramanik, "Segmentation and histogram generation using the HSV color space for image retrieval," in *Proc. 2002 Int. Conf. on Image Processing*, Vol. 2, pp. II–589, IEEE (2002).
32. M. Kohler, "Using the Kalman filter to track human interactive motion modelling and initialization of the Kalman filter for translational motion," Technical Report, University of Dortmund, Germany (2007).
33. O. Kwon, J. Shin, and J. Paik, "Edge based adaptive Kalman filtering for real-time video stabilization," in *IEEE Int. Conf. on Consumer Electronics (ICCE)*, pp. 75–76 (2006).
34. S. Kim and J.-S. Kang, "A new outdoor object tracking approach in video surveillance," in *Advanced Intelligent Systems*, pp. 167–177, Springer International Publishing (2014).
35. V. T. van Hees et al., "Separating movement and gravity components in an acceleration signal and implications for the assessment of human daily physical activity," *PLoS One* **8**(4), e61691 (2013).
36. K. Tuck, "Tilt sensing using linear accelerometers," Freescale Semiconductor Application Note, Freescale Semiconductor, pp. 1–8 (2007).
37. R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, Cambridge, UK (2003).
38. D. F. DeMenthon and L. S. Davis, "Model-based object pose in 25 lines of code," in *Computer Vision ECCV'92*, pp. 335–343, Springer (1992).
39. J. Li et al., "A new 360-degree immersive game controller," in *Proc. 9th Int. Conf. on Distributed Smart Camera*, pp. 201–202, ACM (2015).
40. Z. Huang et al., "Mobile augmented reality survey: a bottom-up approach," arXiv preprint arXiv:1309.4413v2 [cs.gr] (2013).
41. Unity, Unity3d, www.unity3d.com (01 June 2016).



**Juan Li** is pursuing her PhD in the Department of Signals, Systems, and Radio-Communications at Universidad Politécnica de Madrid. She received her BS degree in electrical engineering from Beihang University, China, in 2010. Her current research interests include data fusion, object pose estimation, and interactions in smart spaces.

**Maarten Slembrouck** is a researcher at Ghent University in the Telin Department, research group IPI, which stands for image processing and interpretation. He has been pursuing his PhD since 2011. His work mainly focuses on multicamera 3-D reconstruction in regions covered by multiple cameras. His extended 3-D reconstruction algorithm is also able to recover from severe occlusion in one or multiple camera views.

**Francis Deboeverie** received his Master of Science in electronics and ICT Engineering Technology from the University of Ghent, Belgium, in 2007. In 2014, he received his PhD in engineering from Ghent University, Belgium. Currently, he is a postdoctoral researcher at the Image Processing and Interpretation Group (IPI-TELIN-iMinds) at Ghent University. His research interests include image interpretation with polynomial feature models for real-time vision systems.

**Ana M. Bernardos** is an associate professor in the Universidad Politécnica de Madrid, where she leads research activities in technologies for smart spaces. Her current research interests lie at the intersection of ubiquitous computing, human–computer interaction and data fusion. In this area, she has coauthored more than 80 articles and coordinated the participation in numerous cooperative research projects.

**Juan A. Besada** received his degree in telecommunication engineering from the Universidad Politécnica de Madrid in 1996 and his PhD from the same university in 2001. Currently, he is an associate professor at Universidad Politecnica de Madrid. He has published more than 100 papers in international conferences and journals. His main interests are air traffic control, sensor networks, and data fusion applied to both environments.

**Peter Veelaert** is a full professor in the Department of Telecommunications and Information Processing and the head of the vision systems research group at Ghent University. His recent research has focused on discrete and combinatorial geometry, geometrical models that take into account uncertainty, and motion analysis with smart multicamera systems.

**Hamid Aghajan** is with the Department of Telecommunication and Informatics at Ghent University and has also been director of the Ambient Intelligence Research Lab at Stanford University. He received his BS degree in 1989 from Sharif University of Technology, Tehran, Iran, and his MS and PhD degrees from Stanford University in 1991 and 1995, respectively, all in electrical engineering. The focus of research in his group is on ambient intelligence, behavior modeling based on activity monitoring and multicamera networks.

**José R. Casar** is a full-time professor at the Universidad Politécnica de Madrid and is heading the Data Processing and Smart Spaces Group. He graduated in telecommunication engineering in 1981 and received his PhD in 1983 from the Department of Signals, Systems and Radio Communications of Universidad Politécnica de Madrid. His current interests include the Internet of things, advanced interaction methods, user experience design, and smart spaces. His group is running an experience laboratory on the spaces of the future.

**Wilfried Philips** is a full-time professor at Ghent University and is heading the research group Image Processing and Interpretation, which is also part of the Flemish ICT Research Institute iMinds. In 1989, he received his diploma degree in electrical engineering and in 1993, his PhD in applied sciences, both from Ghent University, Belgium. The recent research activities in the group include image and video restoration and analysis, image and video quality assessment and computer vision.