A Review Of Visual Inertial Odometry For Object Tracking And Measurement

Nor Azman Ismail, Tan Chun Wen, Md Sah Salam, Abdullah Mohd Nawi, Su Elya Namira Mohamed

Abstract: This paper aims to explore the use of Visual Inertial Odometry (VIO) for tracking and measurement. The evolution of VIO is first discussed, followed by the overview of monocular Visual Odometry (VO) and the Inertial Measurement Unit (IMU). Next, the related measurement approaches and the use of VIO for measurement have also been discussed. Visual Inertial Odometry is the combination of IMU in the VO system in which the visual information and inertial measurements are combined to achieve an accurate measurement. The algorithm of VO system contains four components, which are camera calibration algorithm, the feature tracker algorithm (usually the KLT algorithm), the rigid motion estimation algorithm, and the algorithm that matches a description of the features points (typically RANSAC algorithm). The IMU is the combination of accelerometer, gyroscopes and magnetometer that measures the linear and angular motion. To fuse the visual and inertial measurements data, there are two different approaches based on when and how they were fused. Tightly coupled and loosely coupled are the approaches for when the measurements are fused, while filtering and optimization based are the approaches for how they were fused. Studies on related measurement approaches can be summarized as three methods which are using the time-of-flight camera, dual cameras (stereovision), or the single camera known as monovision. This review shows that the technique that utilizes the VIO to get visual information and inertial motion has been used widely for measurement lately especially for the field related to Augmented Reality.

Index Terms: Height Measurement, Inertial Measurement Unit, Motion Tracking, Visual Odometry, Visual Inertial Odometry.

1. INTRODUCTION

Human height measurement can be achieved by using contact or non-contact techniques, where the former undeniably the traditional measuring method which requires a human resource to perform the measurement and can be considered the less desirable between the two. On the other hand, there are many non-contact methods that have been researched by researchers in this area, most of them being image-processing based. However, a new trend has proven to be very popular, especially with the advancement in technology available nowadays to the average consumer, namely Augmented Reality due to the utilization of the Visual Inertial Odometry (VIO) method in mobile devices. Visual Inertial Odometry is the fusion of Inertial Measurement Unit (IMU) in the Visual Odometry (VO) system. It estimates the ego-motion which is the 3D camera motion from a combination of images and the measurements from the IMU. The Inertial Measurement Unit is an independent system which allows the measurement of linear and angular motion by using multiple sensors such as accelerometers, gyroscopes and sometimes magnetometers. Nowadays, these inertial sensors come as a standard in smartphones which enables the measurement of linear and angular motion of the device. These inertial sensors provide 6 degrees-of-freedom which allows the detection of 3-axis translation and 3-axis rotation of the device. In the following section, the evolution of visual inertial odometry is reviewed, followed by the description of monocular visual odometry and IMU. The different approaches to measurement which include the use of VIO for measurement are also presented.

2 EVOLUTION OF VISUAL INERTIAL ODOMETRY

The introduction of Simultaneous Localization and Mapping (SLAM) in 1986 had given a great impact toward the autonomy of robots [7]. In the SLAM system model, detection of landmarks was required which was then used to be compared with a map to localize the robot while at the same time adding newly detected landmarks to that map. A variety of sensors were required for the detection of landmarks such as laser scanners and cameras. Early SLAM approaches represented the robot's state, as well as the 3D landmarks, probabilistically by using the weighted average of noisy measurements in a process called filtering. The filtering method, Extended Kalman Filter (EKF) was the first popular choice for SLAM. Therefore, a first real-time monocular SLAM (MonoSLAM) using an Extended Kalman Filter was introduced in 2003 by Davison [1]. MonoSLAM uses image features to represent and add landmarks to the map that contribute important 3D location information. Features that were no longer useful were removed in MonoSLAM. Although MonoSLAM resolved the problem of being able to be run in real-time, it was still limited to a small area because the problem of quadratic increase in complexity with the number of landmarks remained unsolved.

To accomplish real-time performance with a monocular SLAM system, a new standard framework was set by two researchers on Augmented Reality (AR) in 2007. Klein and Murray in 2007 proposed Parallel Tracking and Mapping (PTAM) for small AR Workspace, a SLAM algorithm with a feature-based method that tracks and maps many features to achieve robustness [2]. It used the keyframe-based Bundle Adjustment instead of filtering for pose and map refinement. While exploring a map, the keyframe-based method maintains a sparse set of important features, along with the landmarks detected and the camera's position. One advantage of this method is the retention of old information for explicit use instead of marginalizing out old landmarks and camera poses as in EKF-SLAM. Although PTAM was still limited to the small working area, it increased the number of usable landmarks significantly as compared to EKF-SLAM. The parallel tracking and mapping made PTAM to be widely used by most of the

- Nor Azman Ismail: School of Computing, Faculty of Engineering, Universiti Teknologi Malaysia, PH-0183104072. E-mail: azman@utm.my
- Chun Wen Tan: School of Computing, Faculty of Engineering, Universiti Teknologi Malaysia, E-mail: shadowtan06@gmail.com
- Md Sah Salam: School of Computing, Faculty of Engineering, Universiti Teknologi Malaysia, E-mail: sah@utm.my
- Abdullah Mohd Nawi: Language Academy, Faculty of Social Sciences and Humanities, Universiti Teknologi Malaysia, E-mail: abdnawi@utm.my
- Su Elya Namira Mohamed: School of Computing, Faculty of Engineering, Universiti Teknologi Malaysia, E-mail: suelyamohamed@ymail.com

modern feature-based visual SLAM systems or visual odometry systems. Due to the almost exclusive dependency of detection and triangulation of landmarks of the previous method, dense methods, also referred to as direct methods that make use of the entire image for the frame to frame track was introduced. Kinect Fusion [3] and DTAM (Dense Tracking and Mapping) [4] were the first major real-time advances in dense tracking and mapping. However, these early methods required high levels of parallel processing, which were done on a GPU. After two years, a semi-dense visual odometry for the monocular camera in 2013 was introduced. The multiple cores real-time running capability on this system was later successfully ported to a modern smartphone [5]. The common problem with the monocular camera is the lack of scale. Furthermore, scale drift can occur when the scale is estimated during initialization only as mentioned in the study of Strasdat in 2010 [6]. This scale ambiguity problem is caused by the absence of depth information at which it limits the measurements made to be recoverable only up to a scale. Therefore, this visual system usually accumulates errors over distance, which means the further we travel, the higher the error. To mitigate this scale problem, an inertia-aided method was used which involves the use of the Inertial Measurement Unit (IMU) together with the monocular camera. The use of IMU with visual odometry, the monocular camera, led to the term Visual Inertial Odometry (VIO). In 2013, Weiss et al. used the IMU and monocular camera for odometry by using the camera as a 6 degrees of freedom sensor, which was the loosely coupled approach with Kalman Filter for state estimation [8].

3 MONOCULAR VISUAL INERTIAL ODOMETRY

Monocular visual odometry can be defined as the prediction of camera motions in sequence depending on the perceived pixels' movement in the sequence of images. Several assumptions need to be made for the implementation of monocular visual odometry. First, the camera's pointed view should be directed at a flat surface. Next, the assumption is made for the rigid movement between the ground coordinate system (C.S.) or known as plane surface and the camera C.S., where it should always be the same. There are four components in the visual odometry, which are the calibration of camera, feature tracker, rigid motion estimation algorithm, and the algorithm that matches the point of the features descriptions which is usually the RANdom Sample Consensus (RANSAC) algorithm. The first one, the camera calibration algorithm cannot be considered as a fragment of the visual odometry, due to it being only a one-time execution which is in the initialization process at which the determination of the transformation between the ground surface and the camera C.S. occur. Hence, for the real-time functionality on the mobile device, there are three main basic parts of visual odometry that must be applied, and they are the Kanade-Lucas-Tomasi (KLT) algorithm, the RANSAC algorithm, and the algorithm to determine rigid transformation. The KLT algorithm [11], [12], [13] is a feature tracker used to find out the movement between two subsequent captured images at time t and t + Δt. The RANSAC algorithm [9], [10] is an iterative scheme that uses a set of data observed including the outliers to predict the parameters of a mathematical model. By using this algorithm, it allows a robust estimation of the parameter of a model which suits the given data even when the data contain elements highly different from the precise values with the limitation of up to 50 percent elements. For the algorithm that determines the rigid motion, only a single point movement that can indicate the whole points' movement is described, instead of describing all the individual points' movement with the purpose of simplifying the computational complexity. The vectors norm and their vector product remain unchanged as the distance and the orientation of the points are maintained. The workflow of the visual odometry algorithm is as shown in Fig. 1 which is divided into two threads or parts since the modern smartphone has built-in multi-core processors and to reduce intensive computation. The left part of the diagram is the KLT algorithm or the determination of traces while the right part of the diagram is the rigid movement estimation algorithm and the RANSAC algorithm. As each component of the visual odometry algorithm runs in different threads, communication among them is required. Hence, exchanging of data in the threads is shown with a dashed line as shown in Fig.1. The first thread is in charge of finding out the traces of feature points and the transmission of the currently active traces information to the second thread. The trace information is used to identify the rigid movement and the traces that indicate the outliers corresponding to the selected model. The second thread then transmits back the trace information to the first thread. The information is then saved in the cookie of the individual trace. When a new image is captured in the KLT thread, these cookies are updated and if required, addition and deletion of traces may be involved. The rigid transformation or movement from current C.S. to the world C.S. is acquired at the end of the algorithm.

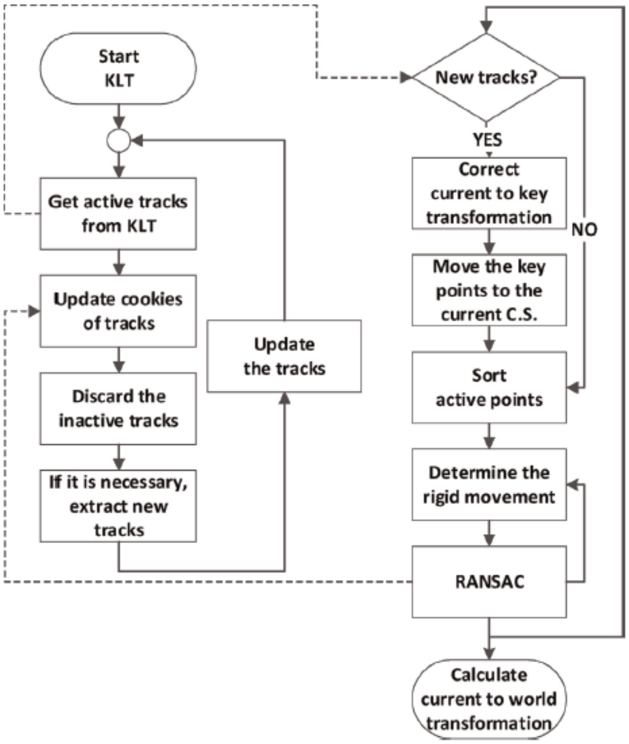
Start KLT
Get active tracks from KLT
Update cookies of tracks
Discard the inactive tracks
If it is necessary, extract new tracks
Update the tracks
New tracks?
YES
NO
Correct current to key transformation
Move the key points to the current C.S.
Sort active points
Determine the rigid movement
RANSAC
Calculate current to world transformation

***Fig 1**. The workflow of visual odometry algorithm [9], [10].*

In visual odometry, all detected movements need to be calculated in 2D space which is only being transformed into 3D space as the final result. In the coordinate systems for the rigid movement estimation in 2D space, there are three major coordinate systems that should be aligned in the beginning of the algorithm, which are the current (plane) C.S. (P), key C.S.

(K) and world C.S. (W) as shown in Fig. 2. These coordinate systems possessed a different direction with regards to the ground C.S., as well as lies on the XZ plane of ground C.S. The Y_{2D} axis of 2D current C.S. is in line with the negative X direction of the ground C.S while the X_{2D} axis is in line with the Z axis of the ground C.S. Every new frame of image in current C.S. will result in the acquisition of the new active trace positions. The key points of the first image frame defined the current position of the camera (final result of visual odometry) in the world C.S. The first frame determines the key C.S. where it only moves in the current C.S. when there is an addition of new traces. Therefore, for this example, the rigid transformation from key C.S. to the world C.S. g_{WK}^{2D} becomes equal to the transformation g_{WP}^{2D}.

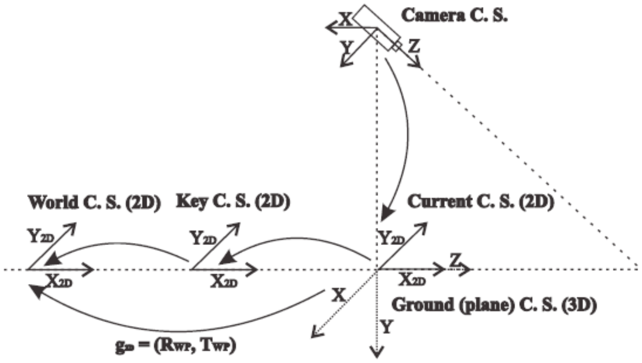

Fig 2. *2D transformation among current, key and world C.S. with rigid motion [9], [10].*

By using the defined parameters and radial distortion, the points of the normalized imaging plane can be computed from the pixels obtained from the active traces. After that, these points need to be transformed into the ground space with 2D in order to ensure that the coordinates are defined in the current C.S. The detected feature points must be sorted to ensure the alignment on the surface before proceeding to the calculation of the rigid transformation using these points. This can be achieved by pixel transformation in the ground C.S. and checking the positivity of the Y components of the points. The ray passing through the point on the plane of the image and penetrates the floor indicating that the Y components of the point is positive. Point X with value (x,y) was expressed in the form of homogeneous coordinate (x,y,1), where the point is obtained through the normalized imaging plane by the ray traveling from the focal point to the ground. It is necessary to compute the scale factor λ that derived the distance to the point of the camera C.S. in the Z direction in order to compute the corresponding ground point (λ_x,λ_y,λ) in camera C.S. To obtain this scale factor, rotation matrix, R_{pc} was used to rotate the homogeneous coordinate of the point X until the point (X,Y,Z) having the axis coincided to the ground C.S. is obtained. The Y coordinate must be equivalent to the camera height from the ground by assuming that the point is on the ground. Hence, the 2D point is defined as

$$X_P^{2D} = (Z\lambda, -X\lambda) \quad (1)$$

where $\lambda = \frac{camera\ height}{Y}$

The camera height is acquired from the computed translational vector, T_{pc} in the calibration phase.

In the 2D current C.S, the KLT algorithm obtains the active points and then maps the image plane to the ground. With the mapping of these active points, g_{KP}^{2D}, the rigid transformation's model from the current C.S. to the key C.S. can then be computed. This rigid movement denotes the kernel of whole visual odometry. All of these 2D rigid transformations were depicted as the averages of the translation vector T = $[X,Y]^T$, and an angle of rotation (yaw), that defined the rotation matrix

$$R = \begin{bmatrix} \cos(yaw) & -\sin(yaw) \\ \sin(yaw) & \cos(yaw) \end{bmatrix} \quad (2)$$

Next, computation of the rigid transformation between current and key points on the plane is performed through the use of Procrustes analysis. After performing the analysis, the appropriateness of the model parameters will be examined using the RANSAC algorithm. The current model points in the normalized image coordinates from the key points are calculated through the use of the model with the corresponding parameters and the rigid movement among the ground and camera coordinate systems. Comparison of these points are made with the actual current points by computing the Euclidean distance which is expressed in the form of pixels. With known distances between the points, points that best describe the model can be determined. The model is accepted provided that the number of points is bigger than the defined threshold. The points and their traces will be added to the list for deletion if they do not suit the resulting model. Unused trace for the last few images will be deleted, while creation of new traces is achieved by the approaches of KLT algorithm once the number of active traces drops beneath the defined threshold value. The feature points' active traces are updated while every new image frame is used to compute the final transformation from the camera C.S. to the world C.S. The movement from current C.S. to the camera C.S., g_{wp} which is the dotted line and is aligned with the ground C.S. as shown in Fig.2 need to be computed, by figuring out the 3D transformation from the world C.S. to the camera C.S. The translation and rotation equation are

$$T_{WP}^{3D} = [-T_Y, 0, T_X] \ (T_{WP}^{2D} = [Tx, Ty]) \quad (3)$$

And

$$R_{WP}^{3D} = roty(-yaw)\ r \quad (4)$$

The angle of rotation, yaw is negative in sign, as the direction of rotation axis in 3D space is opposite to the direction in 2D space.

When the 3D transformations g_{pw} = g_{wp}^{-1} and g_{cp} are determined as shown in Fig. 3, the rigid transformation from the world C.S. to the camera C.S. can be computed as

$$g_{cw} = (R_{cw}, T_{cw}) = g_{cp} g_{pw} \quad (5)$$

where g_{cp} is obtained in the calibration phase.

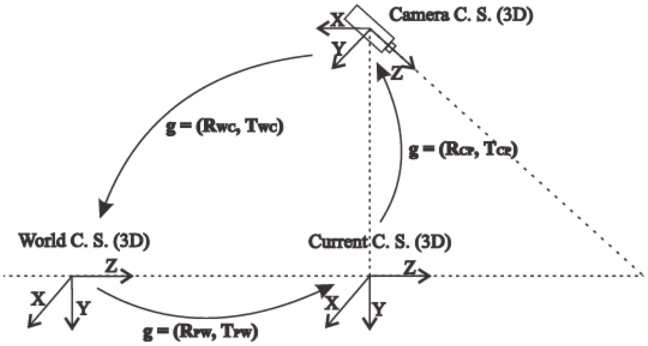

Fig 3. *The rigid transformation from the world C.S. to the camera C.S. [9], [10]*

Lastly, calculate the inverse transformation

$$g_{cw}^{-1} = g_{wc} = (R_{wc}, T_{wc}) \quad (6)$$

The translation vector T_{wc} obtained is the output of monocular visual odometry at which it defines the camera's position along the world C.S. at any instant.

4 INERTIAL MEASUREMENT UNIT

Lately, the most common sensors in a mobile phone are the accelerometer, gyroscopes, and magnetometer. With accelerometer, the device's acceleration, tilt and vibration can be detected to identify the movement and orientation. The gyroscope allows the identification of the relative directions which are up, down, left, right, forward and backward. Rotation around the three axes is also part of the gyroscope capability. The axis orientation is not affected by tilting of the mounting. Thus, gyroscope is used to help provide consistency or to preserve a reference direction. Next is the magnetometer, which allows the detection of magnetic north and is most commonly used in GPS to determine the user's location. With the capability of determining the direction of gravity (which is in line with the Z axis of the world coordinate system) by the accelerometer and the direction of magnetic north (Y-axis) provided by a magnetometer, X-axis can be obtained by the computing the cross product of these two directions. However, a gyroscope can measure angular velocity (relative rotation) accurately and its high responsiveness (high sampling rate). It is needed for a more accurate and stable measurement. With the presence of magnetic interference, it can eradicate the false rotation returned by the magnetometer. The measurement values made by all these sensors correspond to the mobile device coordinate system as shown in Fig. 4. The occurrence of noise, drift and bias are often present in these inertial sensor's measurement, so they are seldom used individually. But these biases can be highly eliminated through the combination of all three sensors into one sensor, namely the Inertial Measurement Unit (IMU). This combination is normally done by using the Kalman filter algorithm as it is a common sensor fusion and data fusion algorithm. Based on the Super Ventures Blog, 2018 in Medium website, the common method used to measure device movement in between IMU readings is dead reckoning. It is a process of using previously derived positions to compute current position and advancing the computed position on the basis of known or estimated speeds over the time elapsed [15]. The problem is the cumulative error, where the error accumulates over time due to noises and biases in IMU data [15].

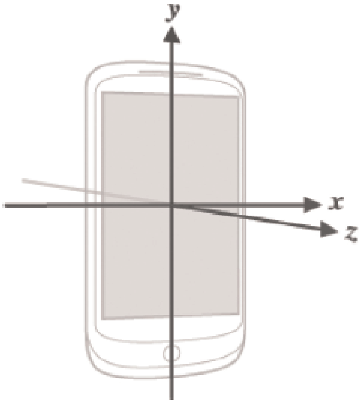

Fig 4*. Smartphone coordinate system [14].*

5 FUSION OF VISUAL DATA AND INERTIAL MEASUREMENTS

The shortage of visual data is that error accumulates with distance while the shortage in inertial measurement is that error accumulates with time. Thus, there is a need to fuse these two data for an accurate tracking and measurement, which leads to the term visual inertial odometry. The VIO approaches based on when visual and inertial measurements are fused can be classified into two ways, which are the loosely coupled approach and tightly coupled approach [9], [10]. The loosely coupled approach is the approach that separately estimates the image motions and inertial measurements, and then fuses these two estimates to obtain the final estimate [17]. The loosely coupled approach has advantages in terms of flexibility and efficiency. The tightly coupled approach is the approach that fuses the visual and inertial data directly at the measurement level to jointly estimate all IMU and camera states [17]. Generally, the tightly coupled approach is better in term of accuracy and robustness in motion estimation [17]. There were also two different approaches for VIO approaches according to how visual and inertial measurements are fused, namely the filtering-based approach and optimization-based approach [9], [10]. The Extended Kalman Filter (EKF) is typically employed in the filtering-based approach. According to the article written by Gui et. al., 2018, the framework of EKF usually comprises an estimation step and an updating step. In the estimation step, the inertial sensors provide the acceleration and the measurements of three axes rotational velocity. These measurements act as the data-driven dynamic model for a 3D rigid motion and make the motion estimation. In the updating step, the visual measurement models (cameras) update the estimation results, by providing the ranging and angular measurements among the mobile platform and features [16]. The optimization-based approach mainly depends on the feature extraction and image alignment optimization, where both are image processing techniques. The inertial measurement acts as prior terms. According to the same article written by Gui et .al, 2018, there are two stages in most cases namely mapping and tracking. For the mapping stage, the image features in 3D space like edges, corners or other landmarks are retrieved through various features detectors. After that, all the features detected are used to determine a reprojection error between two images, which is then optimized to find the landmarks or features coordinates. For the tracking stage, the reprojection error between two images are determined by using the landmarks or features coordinates in the map. Optimization is then applied to get the modification in the orientation and position of the mobile platform [16].

6 RELATED MEASUREMENT APPROACHES

There have been several approaches performed by other researchers in terms of measurement starting from time-of-flight method moving toward stereovision and lastly monovision. The timeline of techniques that have been used is shown below in Table 1. In 1982, a time of flight (TOF) distance measurement method using ultrasonic waves was introduced by Claudio et al. [18]. The time of the ultrasonic wave from and to the object was computed to obtain the distance. However, the problem of this was that the accuracy of the distance calculation was not enough, and the ultrasonic wave was shorter.

***Table 1**. Timeline of literatures chosen*

Year	Title
1982	“A temperature compensated ultrasonic sensor operating in air for distance and proximity measurements” [18]
1997	“Integrated Time-of-Flight Laser Radar” [19]
1998	“Noncontact measurement of vibration using airborne ultrasound” [20]
2000	“Laser Measuring System For Flexible Micromanipulation Station” [21]
2002	“A twofold modulation frequency laser range finder” [22]
2006	“Calibration for increased accuracy of the range imaging camera SwissRanger” [23]
2012	i. “Accuracy and resolution of Kinect depth data for indoor mapping applications” [25] ii. “Distance Measurement Using Dual Laser Source and Image Processing Techniques” [26]
2013	“Happy Measure: Augmented Reality for Mobile Virtual Furnishing” [27]
2015	“Object distance measurement using a single camera for robotic applications” [28]
2017	Implementation of Visual Inertial Odometry in ARKit [29], [30], [31]

In 1997, Palojarvi et .al [20] introduced a laser pulse for distance measurement which emits a signal and computes the time of departure and arrival of that emitted signal to get the distance. This method resulted in worse accuracy even though it made the measurement easily. In 1998, an enhancement of the ultrasonic wave method was performed [20]. A phase difference calculating technique by using the means of the ultrasonic wave was proposed. This method had disadvantages in long-distance measurement as the accuracy changes depending on the distance of the object. In 2000, a triangular measurement principle was introduced into the distance measurement [21]. The distance was determined by using the emittance of a laser beam and the laser point variation of the object. The mechanism they used involved the image processing technique which was simple to use, but was rather imprecise. In 2002, a laser distance measurement with phase shift method was established [22]. To obtain the distance, the phase shift of the modulated laser waves received and transmitted was calculated. The advantage was that it had a high precision measurement, but it was undesirable due to the complexity of the driven circuit. In 2006, A calibration scheme was proposed to increase the range imaging camera's accuracy, SwissRanger which was also known as the time of flight camera [23]. Another range camera known as the Kinect arose for depth measurement in 2012. The Kinect is a triangulation sensor that is different from the SwissRanger (time of flight measurement principle). In 2012, an experiment was performed for the calibration of Kinect sensors for depth measurement [25]. The outcome was that the increase in the distance to the sensor will result in an increase in error. In the same year, Abdulqadir and Abduladheem proposed a stereovision approach for distance measurement in 2012. They used a dual laser source with image processing technique for measurement of distance or displacement of an object. The laser spot was detected using image processing and the distance based on the laser spot position on the image and the relationship between pixel number and distance was calculated. However, accuracy was a problem as the system focused more on cost reduction [26]. In 2013, the marker-based approach to perform measurement was used in the Augmented Reality environment [27]. A 2D marker was used to define the world coordinate frame and to provide the system with all geometric information about the camera’s pose by the acquired image. Next, a new approach to measurement which was the single camera approach was done by Alizadeh for robotic application in 2015. His study focused on the object distance measurement using the image processing technique. The study involved a single fixed camera applied with feature extraction algorithms and a single camera with variable pitch angle. This approach leaves a significant impact for further studies on the use of the single camera in related applications [28]. In late 2017, visual inertial odometry was implemented in Apple’s ARKit for measuring tasks in an Augmented Reality environment [29], [30], [31]. Apple’s ARKit explored a new and more realistic experience of performing measurement in an Augmented Reality environment as it was the first markerless based approach till now.In general, there have been three types of approaches for measurement which are time-of-flight, stereovision and monovision.

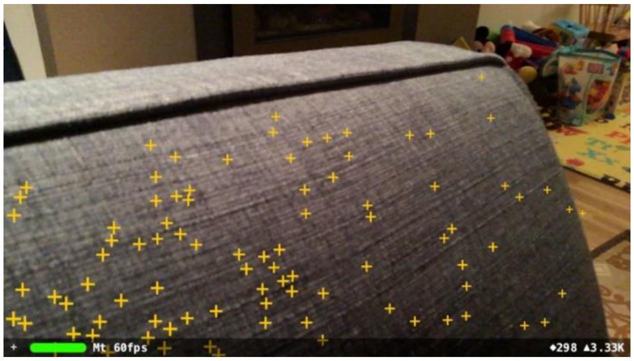

***Fig. 6**. Feature point extraction using ARKit [31]*

7 AUGMENTED REALITY MEASUREMENT USING VIO

The combination of visual information and inertial measurements has been greatly used as motion tracking technique in an Augmented Reality environment which can be used for obtaining an accurate measurement. Both the ARKit platform and ARCore platform use this technique to enable accurate motion tracking in real time. With accurate motion tracking, different measuring tasks can be achieved. The basic concept of motion tracking is that it starts in one spot, in which the system recognizes it as starting location. Then with the movement of the camera, the accelerometer and gyroscopes calculate the distance travelled from the starting location and in what direction. After that, combined with the information gathered from the camera, a clear image of the world around it is formed. The motion tracking technique in ARKit, known as world tracking, utilises the VIO technique [30], has led to a new way of tracking in Augmented Reality and has created vast measurement applications. ARKit detects a bunch of feature points in the environment by using the iPhone or iPad built-in camera and motion sensors. After that, it tracks them with the phone movement. A 3D space model is not created, but it can "pin" objects to one point, changing the scale and perspective of that object. Flat surfaces are also able to be detected. The typical process involved in computing measurement are image acquisition, feature extraction and feature comparison. The first step involved is the camera on the mobile phone continuously reads video frames and the scene images are obtained. The notable features in the scene images are identified and feature points are then extracted to identify the surface in the scene. The crosses are displayed as unique features found as shown in Fig. 6. With identified features, the features are then compared between two images. The changes in the relationship between features are determined. The known features can then be tracked as the camera continues to produce a sequence of images. Visual information is obtained with the tracking of these known features, while at the same time the inertial data is obtained from the tracking of device poses by the inertial system. These two outputs data are then joined via a Kalman Filter that determines which of the two systems provide the best approximation of the real-world position.

8 CONCLUSIONS

This paper described the historical event that lead to the evolution of visual inertial odometry. The details of the algorithm involved in monocular visual odometry system had also been summarized in this paper. It was concluded that the components in visual odometry were the camera calibration, feature tracker, algorithm for rigid motion estimation, and the RANSAC algorithm. This paper also explained the components in IMU, its function as well as the dependency of each components for obtaining better inertial measurements. The approaches for when and how the fusion of visual and inertial measurements made were also explained. For when these data were fused, it was identified that tightly coupled approach provided accurate and robust measurement while loosely coupled approach had better efficiency and flexibility. The overview process for filtering and optimization-based approaches used for how these data fused were discussed. In addition, various approaches that have been done for carrying out measurement are presented, which include the VIO measurement approach. The main objective of this paper was to determine what is visual inertial odometry and how it works in achieving accurate tracking which in turn obtain an accurate measurement. The grown in Augmented Reality related field had made the grown in using VIO technique for motion tracking as a key element for better accuracy, robustness and efficiency in tracking and measurement.

ACKNOWLEDGMENT

This research is supported by School of Computing, Faculty of Engineering, Universiti Teknologi Malaysia (UTM). We would like to thank members of VicubeLab research group who has provided insights and expertise to assist this research.

REFERENCES

[1] A. Davison. Real-time simultaneous localisation and mapping with a single camera. In Proc. of the International Conference on Computer Vision (ICCV), 2003.

[2] G. Klein and D. Murray. Parallel tracking and mapping for small AR workspaces. In Proc. of the International Symposium on Mixed and Augmented Reality (ISMAR), 2007.

[3] S. Izadi, D. Kim, O. Hilliges, D. Molyneaux, R. Newcombe, P. Kohli, J. Shotton, S. Hodges, D. Freeman, A. Davison, and A. Fitzgibbon. KinectFusion: real-time 3d reconstruction and interaction using a moving depth camera. In Proc. of the 24th annual ACM symposium on User interface software and technology (UIST), 2011.

[4] R.A. Newcombe, S. Lovegrove, and A.J. Davison. DTAM: Dense tracking and mapping in real-time. In Proc. of the International Conference on Computer Vision (ICCV), 2011).

[5] J. Engel, J. Sturm, and D. Cremers. Semi-dense visual odometry for a monocular camera. In Computer Vision (ICCV), 2013 IEEE International Conference on, pages 1449–1456, Dec 2013.

[6] H. Strasdat, J. Montiel, and A. Davison. Scale-drift aware large scale monocular SLAM. In Proc. of Robotics: Science and Sytems, 2010

[7] Shelley, M. A. (2014). Monocular visual inertial odometry on a mobile device. Master's thesis, Institut für Informatik, TU München, Germany.

[8] Stephan Weiss, Markus W. Achtelik, Simon Lynen, Michael C. Achtelik, Laurent Kneip, Margarita Chli, and Roland Siegwart. Monocular vision for long-term micro aerial vehicle state estimation: A compendium. Journal of Field Robotics, 30(5):803–831, 2013.

[9] Tomažič, Simon, and Igor Škrjanc. "Monocular Visual Odometry On A Smartphone." IFAC-PapersOnLine 48.10 (2015): 227-232. Web.

[10] Tomažič, Simon, and Igor Škrjanc. "Fusion of Visual Odometry And Inertial Navigation System On A Smartphone." Computers in Industry 74 (2015): 119-134. Web

[11] B.D. Lucas, T. Kanade, An Iterative Image Registration Technique with an Application to Stereo Vision. In: Proceedings of the 7th International Joint Conference on Artificial Intelligence – Volume 2, (1981) pp. 674–679. Morgan Kaufmann Publishers Inc. Canada.

[12] C. Tomasi, T. Kanade, Detection and Tracking of Point Features. Int. J. Comput. Vision. Technical Report, Carnegie Mellon University (1991).

[13] J. Shi, C. Tomasi, Good features to track, Computer Vision and Pattern Recognition, 1994. Proceedings CVPR '94., 1994 IEEE Computer Society Conference on (1994) 593–600

[14] SensorEvent | Android Developers. (2018). Developer.android.com. Retrieved 31 March 2018, from http://developer.android.com/reference/android/hardware/SensorEvent.html#values.

[15] Why is ARKit better than the alternatives? – Super Ventures Blog – Medium. (2018). Medium. Retrieved 30 March 2018, from https://medium.com/super-ventures-blog/why-is-arkit-better-than-the-alternatives-af8871889d6a

[16] J. Gui, D. Gu, S. Wang and H. Hu, "A review of visual inertial odometry from filtering and optimisation perspectives", Advanced Robotics, vol. 29, no. 20, pp.

1289-1301, 2015. Available: 10.1080/01691864.2015.1057616.

[17] Y. He, J. Zhao, Y. Guo, W. He and K. Yuan, "PL-VIO: Tightly-Coupled Monocular Visual–Inertial Odometry Using Point and Line Features", Sensors, vol. 18, no. 4, p. 1159, 2018. Available: 10.3390/s18041159.

[18] C. Canali, G. De Cicco, B. Morten, M. Prudenziati and A. Taroni, "A Temperature Compensated Ultrasonic Sensor Operating in Air for Distance and Proximity Measurements", IEEE Transactions on Industrial Electronics, vol. -29, no. 4, pp. 336-341, 1982. Available: 10.1109/tie.1982.356688.

[19] P. Palojarvi, K. Maatta and J. Kostamovaara, "Integrated time-of-flight laser radar", IEEE Transactions on Instrumentation and Measurement, vol. 46, no. 4, pp. 996-999, 1997. Available: 10.1109/19.650815.

[20] O. Mater, J. Remenieras, C. Bruneel, A. Roncin and F. Patat, "Noncontact measurement of vibration using airborne ultrasound", IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control, vol. 45, no. 3, pp. 626-633, 1998. Available: 10.1109/58.677607.

[21] A. Buerkle and S. Fatikow, “Laser Measuring System For Flexible Micromanipulation Station”, Proceedings of 2000 IEEE/RSJ International Conference On Intelligent Robots And Systems, pp. 799-804, 2000.

[22] S. Poujouly and B. Journet, "A twofold modulation frequency laser range finder", Journal of Optics A: Pure and Applied Optics, vol. 4, no. 6, pp. S356-S363, 2002. Available: 10.1088/1464-4258/4/6/380.

[23] Kahlmann, T., Remondino, F., & Ingensand, H. (2006). Calibration for increased accuracy of the range imaging camera swissrangertm. Image Engineering and Vision Metrology (IEVM), 36(3), 136-141.

[24] Wang, C. M., & Chen, W. Y. (2012, August). The human-height measurement scheme by using image processing techniques. In Information Security and Intelligence Control (ISIC), 2012 International Conference on (pp. 186-189). IEEE.

[25] K. Khoshelham and S. Elberink, "Accuracy and Resolution of Kinect Depth Data for Indoor Mapping Applications", Sensors, vol. 12, no. 2, pp. 1437-1454, 2012. Available: 10.3390/s120201437.

[26] Yasir, Omar & Riyadh, Wameedh & Y Abdulqadir, Omar & Abdul-Adheem, Wameedh. (2012). Distance Measurement Using Dual Laser Source and Image Processing Techniques

[27] R. Swaminathan, R. Schleicher, S. Burkard, R. Agurto and S. Koleczko, "Happy Measure", International Journal of Mobile Human Computer Interaction, vol. 5, no. 1, pp. 16-44, 2013. Available: 10.4018/jmhci.2013010102

[28] Alizadeh, P. (2015). Object distance measurement using a single camera for robotic applications (Doctoral dissertation, Laurentian University of Sudbury).

[29] ARKit - Apple Developer. (2018). Developer.apple.com. Available: https://developer.apple.com/arkit/

[30] Breaking down Apple’s new augmented reality platform. (2018). The Verge. Available : https://www.theverge.com/2017/6/6/15742736/apple-arkit-augmented-reality-platform-wwdc-breakdown

[31] ARKit By Example — Part 2: Plane Detection + Visualization. (2018). Mark Dawson. Available : https://blog.markdaws.net/arkit-by-example-part-2-plane-detection-visualization-10f05876d53