Guidance: A Visual Sensing Platform For Robotic Applications

Guyue Zhou[†], Lu Fang[‡], Ketan Tang[†], Honghui Zhang[†], Kai Wang[†], Kang Yang[†]
[†] {guyue.zhou, ketan.tang, honghui.zhang, kevin.wang, kang.yang}@dji.com,
Dajiang Innovations Technology Co., Ltd., Shenzhen, P.R.China
[‡] fanglu@ustc.edu.cn, University of Science and Technology of China, P.R. China

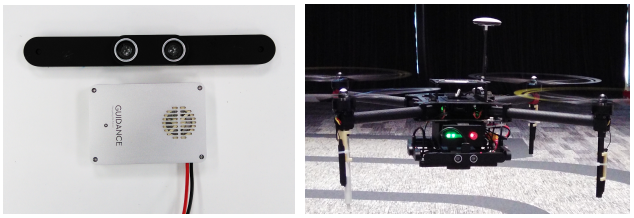
Abstract

Visual sensing, such as vision based localization, navigation, tracking, are crucial for intelligent robots, which have shown great advantage in many robotic applications. However, the market is still in lack of a powerful visual sensing platform to deal with most of the visual processing tasks. In this paper we introduce a powerful and efficient platform, `Guidance`, which is composed of one processor and multiple (up to five) stereo sensing units. Basic visual tasks including visual odometry, obstacle avoidance, depth generation, are given as built-in functions. Additionally, with the aid of a well documented SDK, `Guidance` is extremely flexible for users to develop other applications, such as autonomous navigation, SLAM, tracking.

1. Introduction

Intelligent robots possess huge potential in bringing new elements to people's daily life, making it easier and better. Drones (DJI Phantom) expands people's vision from 2-D to 3-D. Home robots (iRobot) become companion to family. Intelligent robots that have small size, low cost, high maneuverability, and high sensing abilities contribute to various applications, such as aerial photography, aerial inspection, precision agriculture.

To achieve intelligence, visual sensing is crucial due to its relatively low cost and high information throughput. There have been tremendous works discussing the usage of visual sensing in intelligent robots. Taking localization as an example. Visual odometry (VO) estimates the local motion of robots based on visual features [11, 18, 22]. Simultaneously localization and mapping (SLAM) can be treated as the extension of VO, which builds a global map during localization [7, 14]. To handle the intrinsic issue of unknown and unobservable scale of monocular SLAM, fusion of inertial measurement and SLAM results has become a trend [15, 23]. Alternatively, stereo SLAM is developped [8, 17], solving the scale problem with extrinsic stereo calibration.



(a) (b)

Figure 1: (a) Sensor unit and processor unit of `Guidance` and (b) a quadrotor equipped with `Guidance`.

Vision-based navigation is crucial to achieve collision-free flight in a complex environment. A 3-D occupancy map needs to be built to navigate robots through passable areas [9, 20]. 3-D features points are updated by Extended Kalman Filter (EKF) in [7], and improved by inverse depth parameterization in [6]. Bundle adjustment is also used for mapping in [14]. To reduce the data storage burden in mapping, multi-resolution Octomap is usually used to represent the environement map [13].

Vision-based tracking is also one of the key elements that improve the intelligence of robots. Tracking in video has been studied for decades [4, 30]. Recently researchers have studied real-time tracking on robots [21, 24, 28]. However, existing methods either require depth sensors [21] or rely on off-board computation [24, 28], which increases cost and decreases reliability.

Despite the growing demand of intelligent control systems and the rich literature in the visual sensing technologies, the public available platforms are far from powerful enough. PX4FLOW [12] is the only purchasable monocular vision system currently, with a sonar range finder for scale fusion. However its optical flow computation can only handle a small resolution of 64×64, which limits the maximum operating range and accuracy. VI-Sensor [27] is an upcoming stereo vision system with real-time feature detection and extraction conducted on relatively high resolution. However it only works in one direction. Besides, it tends to fail in certain circumstances, such as stereo blind

region, glass windows, water, *etc*. Other entertainment level stereo vision systems, such as Microsoft Kinect or ASUS Xtion, are limited to indoor applications due to their relatively large size and weight, and the infrared sensors they use.

In this paper, Guidance, a brand new on-board vision platform for intelligent robots is introduced. It contains one processor and up to five Stereo Sensing Units (SSU), as shown in Fig. 1a. In particular, to compensate for the failure of vision-based algorithms, an ultrasonic range finder is coupled with each SSU. The Guidance platform provides accurate visual odometry in a very wide range, and obstacle avoidance in almost 360 degrees. A rich categories of data is output, and a well documented SDK is coupled with Guidance, such that developing other applications is extremely simplified.

The remainder of the paper is organized as follows. Section 2 explores the built-in algorithms and functionalities of Guidance. Section 3 introduces the detailed SDK interface. And Section 4 demonstrates some potential applications based on Guidance.

2. Built-in Features

A complete Guidance system consists of one core processing module and multiple sensor modules. In particular, for each sensor module, two mono-color global shutter cameras with VGA (640 × 480) resolution are mounted rigidly together with an ultrasonic sensor. And the core processing module includes a low-cost SOC FPGA (Altera Cyclone V), inertial sensors (MPU 6050), and able to connect up to five sensor modules (the recommended setup is forward, backward, leftward, rightward and downward, respectively). With careful consideration, the product definition focuses on two groups of customers: (1) traditional remote-controlled drone players and (2) robotic application developers. This section will introduce the built-in features for consumer-level drones, starting from the system overview and moving to detailed algorithms subsequently.

2.1. System Overview

Fig. 1b shows the quadrotor equipped with Guidance, in which case the players can enjoy GPS-denied hovering and anti-collision features. Technically, Guidance is in fact an upgraded version of Zhou *et al*.'s work [31] which is a visual mapping solution based on four cameras and a single processing chip - Altera's SoC FPGA. The block diagram of Guidance's built-in functionalities can be found in Fig. 2 and will be explained as follows.

The 20Hz image data will be processed by the five separated remapping kernels, generating at most 10 undistorted and rectified images with QVGA (320 × 240) resolution (although the camera is capable of VGA resolution, we only

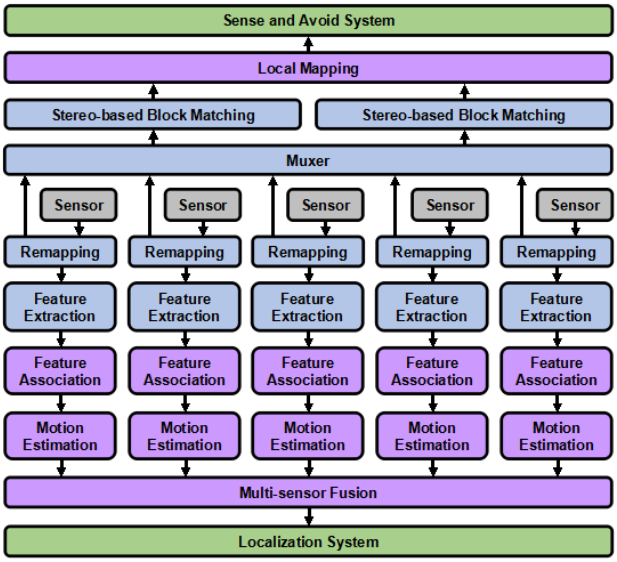


Figure 2: The block diagram of Guidance's built-in functionalities: the blocks with gray, blue, purple and green colors stand for system inputs, FPGA-based hardware kernels, ARM-based software kernels and system outputs respectively.

capture QVGA image for real-time processing). Then, there are two threads for image remapping: (1) visual odometer for the localization system and (2) visual mapping for the sense and avoid system. In the first thread, FAST [25, 26] feature detector, BRIEF [5] feature descriptor and local binary matching are employed for image feature extraction and association based on the analysis in [31]. The algorithms of motion estimation and multi-sensor fusion will be discussed in detail later. In the second thread, the dense local depth map plays the main role. Notice that the stereo-based block matching module consumes a large amount of FPGA resource, therefore the multiplexer is used to select the two most essential stereo pairs for local mapping. The selection criteria mainly relies on the moving direction of the drone.

2.2. Algorithms

The built-in algorithms can be categorized into: visual odometer, visual mapping and multi-sensor fusion. As a matter of fact, the algorithms for pixel-level processing, visual mapping and multi-sensor fusion are introduced in [31]. Here we just want to highlight that the visual odometer used in Guidance is a hybrid version of an inertial-assisted stereo visual odometer discussed in [32] and an inertial-assisted monocular visual odometer which can contribute whenever the stereo system is invalid.

In Fig. 3, the work flow of Guidance's built-in visual odometer is illustrated. The improvements based on [32]

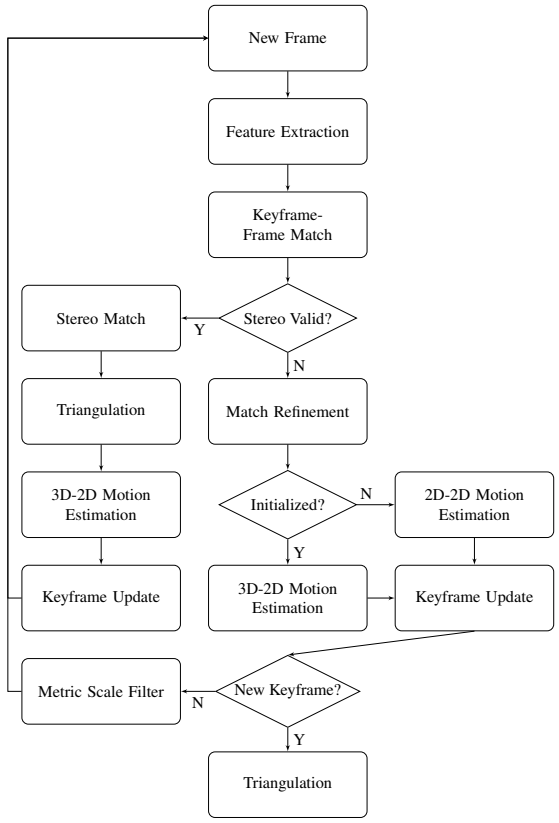


Figure 3: The flowchart of `Guidance`'s built-in visual odometer.

mainly come from the blocks of matching refinement, 2D-2D motion estimation and metric scale filtering.

2.2.1 Matching Refinement

Unlike the stereo case which can generate depth directly with the calibrated extrinsic parameters, the depth from monocular vision relies on the accurate estimated camera ego-motion, therefore more accurate feature correspondences are required. Based on the matching results using FAST feature detector and BRIEF feature descriptor, the non-pyramids Lucas-Kanade Tracker [19] is applied for matching refinement.

2.2.2 2D-2D Motion Estimation

The 2D-2D correspondences are used for motion estimation during the initialization of monocular visual odometer. Each correspondence can be represented by $c_i = \{\boldsymbol{u}_i, \boldsymbol{u}_i'\}$, where $\boldsymbol{u} = [u, v, 1]^T$ is the pixel coordinates in undistorted images. Recall the epipolar constraints $(\boldsymbol{K}^{-1}\boldsymbol{u}_i')^T[\boldsymbol{t}]_\times \boldsymbol{R}(\boldsymbol{K}^{-1}\boldsymbol{u}_i) = 0$ where $\boldsymbol{R}$ can be directly obtained from the on-board inertial sensors and $\boldsymbol{K}$ is the camera intrinsic matrix which can be calibrated in advance. Let $(\boldsymbol{K}^{-1}\boldsymbol{u}_i')^T = [x_i, y_i, 1]$ and $\boldsymbol{R}(\boldsymbol{K}^{-1}\boldsymbol{u}_i) = [a_i, b_i, c_i]^T$. Normalizing the second translational element $t_y = 1$ since the absolute scale can not be observed from monocular vision, a minimal solution (with 2 correspondences) to translation $\boldsymbol{t}$ can be obtained from

$$\begin{bmatrix} t_x \\ t_z \end{bmatrix} = \begin{bmatrix} b_1 x_1 - a_1 y_1 & c_1 y_1 - b_1 \\ b_2 x_2 - a_2 y_2 & c_2 y_2 - b_2 \end{bmatrix}^{-1} \begin{bmatrix} x_1 c_1 - a_1 \\ x_2 c_2 - a_2 \end{bmatrix}. \quad (1)$$

The case with multiple correspondences can be easily extended by solving similar linear equations.

2.2.3 Metric Scale Filter

An extended Kalman filter (EKF) (a modified version based on Weiss's work [15, 29]) is used to recover the metric scale from the loosely coupled visual and inertial measurements. Notice that there exists conditions where some of stereo odometers are valid and some are not. Therefore the stereo odometer results should be considered for scale recovery as well. Let the state vector $\boldsymbol{x}$ defined as below.

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{v}_w \\ \boldsymbol{b} \\ \lambda \end{bmatrix}, \quad (2)$$

where $\boldsymbol{v}_w$ is the metric velocity in the world coordinate frame, $\boldsymbol{b}$ is the bias of accelerometers and λ is the scale factor for the monocular odometer.

The time update which comes from the inertial sensors is written as follows.

$$\begin{aligned} \boldsymbol{x}_k = & \begin{bmatrix} \boldsymbol{I} & -\Delta t\boldsymbol{R} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{I} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & 1 \end{bmatrix} \boldsymbol{x}_{k-1} + \begin{bmatrix} -\Delta t\boldsymbol{R}\boldsymbol{a} \\ \boldsymbol{0} \\ \boldsymbol{0} \end{bmatrix} \\ & + \begin{bmatrix} \boldsymbol{R} & -\Delta t\boldsymbol{R} \\ \boldsymbol{0} & \boldsymbol{I} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{n}_a \\ \boldsymbol{n}_b \end{bmatrix}, \end{aligned} \quad (3)$$

where Δt is the time period between two samplings, $\boldsymbol{R}$ is the relative rotation during this time period, $\boldsymbol{a}$ is the reading from accelerometers with the Gaussian noise $\boldsymbol{n}_a$ and $\boldsymbol{n}_b$ is the Gaussian noise of accelerometer bias.

The measurement update which comes from the visual sensors comes as follows.

$$\boldsymbol{z} = \begin{bmatrix} \boldsymbol{v}_{mono} \\ \boldsymbol{v}_{stereo} \end{bmatrix} = \begin{bmatrix} \frac{1}{2}\lambda\boldsymbol{R}^T & \boldsymbol{0} & \frac{1}{2}\boldsymbol{R}^T\boldsymbol{v}_w \\ \boldsymbol{R}^T & \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \boldsymbol{x} + \boldsymbol{n}_z, \quad (4)$$

where $\boldsymbol{v}_{mono}, \boldsymbol{v}_{stereo}$ are the visual odometer results and $\boldsymbol{n}_z$ stands for their Gaussian noise.

3. Software Development Kit (SDK)

With the `Guidance`'s SDK, both raw and processed data from different sensors can be accessed through the USB 2.0 interface which is compatible with popular open-source software platforms: ROS [3], OpenCV [2] and

Table 1: Available data from Guidance's SDK. Note that the specifications may change in future versions.

Data	Description
Image (10 channels)	QVGA resolution 8-bit grayscale Undistorted and rectified Up to 20 Hz
Depth map (2 channels)	QVGA resolution 16-bit depth Algorithm: OpenCV BM Up to 20 Hz
Obstacle distance (5 channels)	0.1 - 20 m static range Fixed 20 Hz
Ultrasound (5 channels)	0.1 - 8 m static range Fixed 20 Hz
Inertial sensor	3-axis gyroscope 3-axis accelerometer Synchronized with images Fixed 20 Hz
Visual odometer	Body velocity Fixed 10 Hz

MAVlink [1]. Meanwhile, sample codes and Guidance's Wiki page, which can be found at http://dev.dji.com, allow developers to get started in a short time.

The detailed information about the available data with Guidance's SDK can be found in Table 1. Note that the detailed specifications may change in future specifications. The users are encouraged to visit the web page for up-to-date details. Additionally, besides such rich categories of accessible data, developers can also enjoy the following advantages:

- customize the USB data flow within limited bandwidth, implying developers can freely adjust the trade-off between frame rate and channel numbers;
- access to low-speed data (excluding image data and depth map) via UART, therefore it is convenient to communicate with off-the-shelf robotic controllers;
- calibrate the camera parameters easily with either the embedded self-calibration module or the GUI-guided calibration software on PC;
- change the exposure time of the camera sensors to either AEC (automatic exposure control) or constant.

4. Applications

In this section, several practical experiments are shown to demonstrate how drone users and robotic developers can benefit from Guidance.

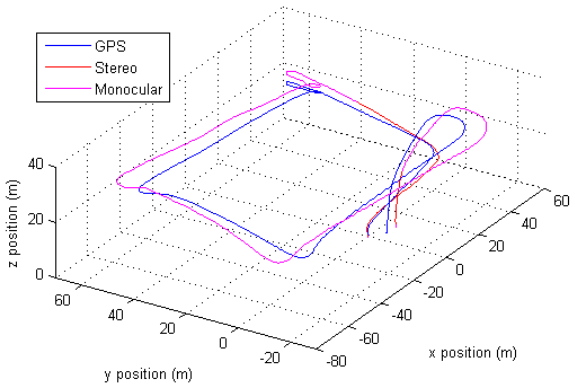


Figure 4: The large-scale performance of Guidance's built-in visual odometer: the trajectories with blue, red and pink colors stand for the positional observation from GPS, stereo visual odometer and monocular visual odometer respectively. Notice that monocular method is only used at extremely low and extremely high altitude, therefore the pink color is only seen in part of the trajectory.

4.1. Autonomous Navigation

With the built-in hybrid stereo/monocular vision odometer and multi-directional observations, Guidance enables the reliable positional control and anti-collision functionalities which are the fundamentals of autonomous navigation.

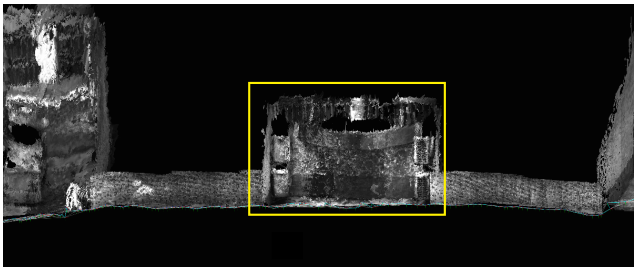
Fig. 4 illustrates the performance of Guidance's odometer in large-scale outdoor environment. The flight duration is in total 109s and the flight distance is around 300 m with more than 20 m height. The monocular method is used in both extremely low and extremely high altitude, in which case stereo method does not have appropriate baseline, while in the middle stereo method is used. Compared to the ground truth provided by GPS, the expectation and standard deviation of the velocity estimation errors using our algorithm are $\boldsymbol{m}_e = [0.0785, 0.0767, 0.0822]^T$ m/s and $\boldsymbol{\sigma}_e = [0.0722, 0.0699, 0.0773]^T$ m/s, respectively. The detailed rum-time of algorithm blocks are listed in Table 2.

4.2. Visual SLAM

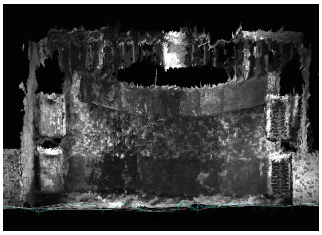
Regarding to SLAM, one of the most important tasks for intelligent robots [7, 10, 14], Guidance provides accurate VO results, which is one of the central parts of SLAM, as well as rectified images of all cameras. The local maps of VO can be easily accumulated to construct a global map using either Kalman filter [7] or bundle adjustment [14], leading to practically visual SLAM results. Alternatively, a monocular SLAM method can be directly implemented

Table 2: Runtime of algorithm blocks.

Block	Runtime
Re-mapping	N/A
FAST	N/A
BRIEF	N/A
Pixel-wise Pipeline	2.56ms
Block-based Stereo Match	11.14ms
Feature Matching	2.73ms
Matching Refinement	4.44ms
3D-2D Motion Estimation	1.61ms
2D-2D Motion Estimation	1.09ms
Triangulation	2.11ms
Multi-sensor Fusion	1.27ms
Visual Mapping	8.38ms
Total Runtime	47.34ms



(a) global map



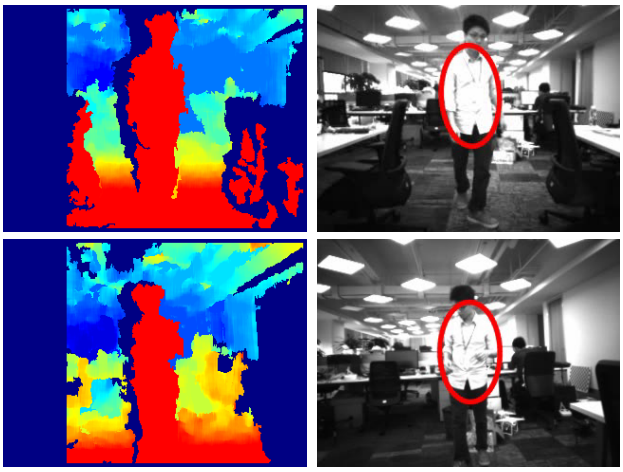
(b) a local map

(c) corresponding left view image

Figure 5: RGBD SLAM result generated via `Guidance` on a representative scene in outdoor environment.

based on rectified images. Followed by global pose graph optimization such as g^{2}o [16], a globally consistent map can be constructed.

Additionally, `Guidance` is able to provide more reliable and precise RGBD SLAM [8, 9], due to the availability of depth map from block matching procedure. As illustrated in Fig. 5, where RGBD SLAM is performed on a representative scene in outdoor environment, the SLAM result (Fig. 5b) is visually pleasant in reflecting the structure of real scene in Fig. 5c.



depth map grayscale image

Figure 6: Results of depth based tracking.

4.3. Depth Based Tracking

Recall that `Guidance` can provide dense depth map generated by stereo matching, which offers extra useful information for many implementations. Without loss of generality, we explore the possibility of depth based tracking by `Guidance` in the following contents.

For ease of implementation, a sample tracking algorithm, CAMshift [4], is adopted, which uses color histogram to model the object, and continuously searches for the mode of the distribution with respect to the color histogram. Due to the lack of color information in the camera output, a two-channel pseudo color image is artificially generated using the grayscale image and its corresponding depth map. Then a 2-D histogram of the object over the pseudo color image is computed, followed by the standard CAMshift algorithm.

We evaluate our depth based tracking via `Guidance` on a challenging indoor sequence, *i.e.*, the saturated lights in ceiling have exactly the same pixel intensity as the object, which tends to induce the failure of traditional pure grayscale image based tracking. As illustrated in Fig. 6, with the help of dense depth map, an accurate and robust tracking becomes possible under the tough case that pixel intensities are hardly distinguished with each other.

5. Conclusion

`Guidance`, a brand new powerful visual sensing platform for robotic applications, is introduced in this paper. In specific, it owns up to five stereo sensing units and one central processor. The ARM+FPGA architecture ensures real-time precessing of built-in functions, including visual odometry, obstacle avoidance, and depth map generation. In addition, a coupled SDK is developed, providing a flexible development platform for users exploring various applica-

tions, such as autonomous navigation, SLAM, tracking, *etc*. Our implementations show that the complicated vision based tasks of robots are significantly simplified and eased with `Guidance`.

References

[1] MAVLink: Micro Air Vehicle Communication Protocol. http://qgroundcontrol.org/mavlink/start. 4

[2] OpenCV: Open Source Computer Vision Library. http://opencv.org/. 4

[3] ROS: Robot Operating System. http://www.ros.org/. 4

[4] G. Bradsky. Computer vision face tracking for use in a perceptual user interface. *Intel Technology Journal*, (Q2), 1998. 1, 5

[5] M. Calonder, V. Lepetit, M. Ozuysal, T. Trzcinski, C. Strecha, and P. Fua. BRIEF: Computing a Local Binary Descriptor Very Fast. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(7):1281–1298, 2012. 2

[6] J. Civera, A. Davison, and J. Montiel. Inverse depth parametrization for monocular slam. *Robotics, IEEE Transactions on*, 24(5):932–945, Oct 2008. 1

[7] A. Davison. Real-time simultaneous localisation and mapping with a single camera. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pages 1403–1410 vol.2, Oct 2003. 1, 4, 5

[8] F. Endres, J. Hess, N. Engelhard, J. Sturm, D. Cremers, and W. Burgard. An evaluation of the RGB-D SLAM system. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 1691–1696, May 2012. 1, 5

[9] F. Endres, J. Hess, J. Sturm, D. Cremers, and W. Burgard. 3-D mapping with an RGB-D camera. *Robotics, IEEE Transactions on*, 30(1):177–187, Feb 2014. 1, 5

[10] J. Engel, T. Schops, and D. Cremers. LSD-SLAM: Large-scale direct monocular SLAM. In *ECCV*, 2014. 4

[11] C. Forster, M. Pizzoli, and D. Scaramuzza. SVO: Fast semi-direct monocular visual odometry. In *ICRA*, 2014. 1

[12] D. Honegger, L. Meier, P. Tanskanen, and M. Pollefeys. An open source and open hardware embedded metric optical flow cmos camera for indoor and outdoor applications. In *Robotics and Automation (ICRA), IEEE International Conference on*, pages 1736–1741, May 2013. 1

[13] A. Hornung, K. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard. Octomap: an efficient probabilistic 3D mapping framework based on octrees. *Autonomous Robots*, 34(3):189–206, 2013. 1

[14] G. Klein and D. Murray. Parallel tracking and mapping for small AR workspaces. In *Sixth IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR)*, 2007. 1, 4, 5

[15] L. Kneip, S. Weiss, and R. Siegwart. Deterministic initialization of metric state estimation filters for loosely-coupled monocular vision-inertial systems. In *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, pages 2235–2241, Sept 2011. 1, 3

[16] R. Kummerle, G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard. g2o: A general framework for graph optimization. In *Robotics and Automation (ICRA), IEEE International Conference on*, pages 3607–3613, 2011. 5

[17] T. Lemaire. Vision-based SLAM: Stereo and monocular approaches. *International Journal of Computer Vision*, 74(3):343 – 364, 2007. 1

[18] A. Levin and R. Szeliski. Visual odometry and map correlation. In *Computer Vision and Pattern Recognition, CVPR*, volume 1, pages I–611–I–618 Vol.1, 2004. 1

[19] B. D. Lucas and 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*, pages 674–679, 1981. 3

[20] D. Magree, J. Mooney, and E. Johnson. Monocular visual mapping for obstacle avoidance on uavs. *Journal of Intelligent & Robotic Systems*, 74(1-2):17–26, 2014. 1

[21] T. Naseer, J. Sturm, and D. Cremers. Followme: Person following and gesture recognition with a quadrocopter. In *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, pages 624–630, Nov 2013. 1

[22] D. Nister, O. Naroditsky, and J. Bergen. Visual odometry. In *Computer Vision and Pattern Recognition, CVPR*, volume 1, pages I–652–I–659 Vol.1, 2004. 1

[23] G. Nützi, S. Weiss, D. Scaramuzza, and R. Siegwart. Fusion of imu and vision for absolute scale estimation in monocular slam. *Journal of Intelligent & Robotic Systems*, 61(1-4):287–299, 2011. 1

[24] J. Pestana, J. L. Sanchez-Lopez, P. Campoy, and S. Saripalli. Vision based gps-denied object tracking and following for unmanned aerial vehicles. In *Safety, Security, and Rescue Robotics (SSRR), IEEE International Symposium on*, pages 1–6, 2013. 1

[25] E. Rosten and T. Drummond. Fusing points and lines for high performance tracking. In *IEEE International Conference on Computer Vision*, volume 2, pages 1508–1511, 2005. 2

[26] E. Rosten and T. Drummond. Machine learning for high-speed corner detection. In *European Conference on Computer Vision*, volume 1, pages 430–443, May 2006. 2

[27] Skybotix. VI-Sensor. http://www.skybotix.com/. 1

[28] C. Teuliere, L. Eck, and E. Marchand. Chasing a moving target from a flying uav. In *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, pages 4929–4934, Sept 2011. 1

[29] S. Weiss, M. W. Achtelik, S. Lynen, M. Chli, and R. Siegwart. Real-time onboard visual-inertial state estimation and self-calibration of mavs in unknown environments. In *Robotics and Automation (ICRA), IEEE International Conference on*, pages 957–964, 2012. 3

[30] Y. Wu, J. Lim, and M.-H. Yang. Online object tracking: A benchmark. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 2411–2418, June 2013. 1

[31] G. Zhou, A. Liu, K. Yang, T. Wang, and Z. Li. An embedded solution to visual mapping for consumer drones. In *Computer Vision and Pattern Recognition Workshops (CVPRW), IEEE Conference on*, pages 670–675, 2014. 2

[32] G. Zhou, J. Ye, W. Ren, T. Wang, and Z. Li. On-board inertial-assisted visual odometer on an embedded system. In *ICRA*, 2014. 2