Deployment of Reliable Visual Inertial Odometry Approaches for Unmanned Aerial Vehicles in Real-world Environment

Jan Bednář[1], Matěj Petrlík[1], Kelen Cristiane Teixeira Vivaldini[1,2], Martin Saska[1]

***Abstract*— Integration of Visual Inertial Odometry (VIO) methods into a modular control system designed for deployment of Unmanned Aerial Vehicles (UAVs) and teams of cooperating UAVs in real-world conditions are presented in this paper. Reliability analysis and fair performance comparison of several methods integrated into a control pipeline for achieving full autonomy in real conditions is provided. Although most VIO algorithms achieve excellent localization precision and negligible drift on artificially created datasets, the aspects of reliability in non-ideal situations, robustness to degraded sensor data, and the effects of external disturbances and feedback control coupling are not well studied. These imperfections, which are inherently present in cases of real-world deployment of UAVs, negatively affect the ability of the most used VIO approaches to output a sensible pose estimation. We identify the conditions that are critical for a reliable flight under VIO localization and propose workarounds and compensations for situations in which such conditions cannot be achieved. The performance of the UAV system with integrated VIO methods is quantitatively analyzed w.r.t. RTK ground truth and the ability to provide reliable pose estimation for the feedback control is demonstrated onboard a UAV that is tracking dynamic trajectories under challenging illumination.**

***Index Terms*— Visual Inertial Odometry, Unmanned Aerial Vehicle, Trajectory Shaping, Feedback Control, Camera Orientation**

Multimedia Materials

The paper is supported by the multimedia materials available at mrs.felk.cvut.cz/icuas2022vio.

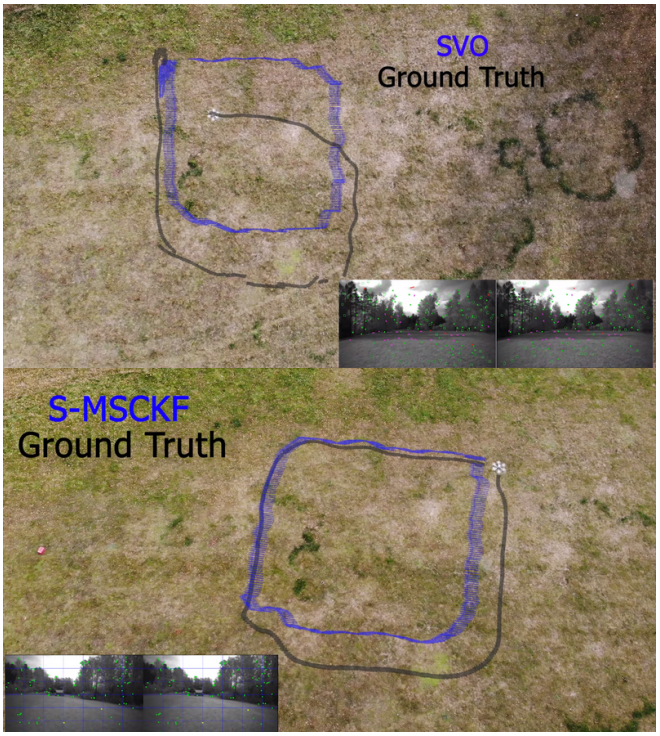

Fig. 1: Top-down overview of the experiments from Sec. V-D, in which SVO and S-MSCKF algorithms are integrated into the feedback of the control pipeline.

I. Introduction

The recent growth in the availability of Unmanned Aerial Vehicles (UAVs) increased their applicability in various applications, such as for example infrastructure inspection [1][2], search and rescue [3], and monitoring [4][5][6]. Current research has been focused on the autonomy of UAVs to allow performing the required task without a human pilot [7]. Real-time UAV localization and state estimation are essential aspects for achieving such autonomous behavior [8].

In most cases of outdoor deployment, the Global Navigation Satellite System (GNSS) is used for localization due to its full availability and easy-to-use approach. However, it limits the UAV deployment to large open areas with direct satellite visibility. Also, its precision is not sufficient for a precise localization required by some applications. The GNSS precision can be improved with Differential GNSS (DGNSS) or Real-Time Kinematic (RTK), which gives a centimeter precision. Nonetheless, these solutions require specific equipment, for example, a base station for sending GNSS position corrections to the UAV, which further reduces the applicability of such a system.

Another less restricting approach is gathering knowledge of UAV surroundings suitable for localization by onboard sensors, such as LiDAR-based (Hector SLAM [9] for single-plane LiDARs, and LiDAR Odometry And Mapping - LOAM [10] for multiple-plane LiDARs), and vision-based methods [11][12].

Although the vision-based approaches are dependent on lighting conditions, using cameras as a localization sensor has several advantages, such as low weight, small size, low cost, and mainly rich data flow. Among many possible combinations of sensors, monocular cameras and IMUs (Inertial Measurement Unit) provide the least expensive and lightweight, but still sufficiently robust and precise state

[1]Jan Bednář, Matěj Petrlík, Kelen Cristiane Teixeira Vivaldini and Martin Saska are with the Faculty of Electrical Engineering, Czech Technical University in Prague, Czech Republic, {jan.bednar14|matej.petrlik|martin.saska}@fel.cvut.cz.

[2]Kelen Cristiane Teixeira Vivaldini is with the Federal University of São Carlos, Department of Computer, Brazil, vivaldini@ufscar.br.

estimation, using Visual-Inertial Odometry (VIO) methods. VIO algorithms increase the robustness of a Visual Odometry (VO), relying on a camera only, by incorporating information from an IMU to improve motion tracking performance. VIO approaches exploit the observation that the visual data obtained by the camera can reduce the drift of the IMU data, whereas the IMU data can retrieve the metric scale and roll and pitch angles of the visual data. Methods that employ stereo cameras instead of monocular cameras estimate the depth of features more accurately, which consequently results in lower metric scale drift. This increase in performance comes at the cost of a slightly higher computational load.

A. Related works

Several benchmark studies for visual estimation of UAV state can be found in the literature. In [13], the benchmark for vision-based odometry is presented and VO methods are compared by evaluating tracking accuracy in terms of the accumulated error (drift) over the full sequence of images, with different resolutions and two lenses with different fields of view. The dataset does not contain a ground truth and the error is obtained by aligning the beginning and end of trajectory segments from the same part of the environment.

A benchmark of visual-inertial methods regarding computational requirements, translation errors, and absolute translation error on the EuRoC dataset [14] is proposed by [15]. The authors conclude that the accuracy and robustness of VIO techniques need to be improved. Other visual odometry comparisons, such as [16], are related only to non-inertial methods or non-6DoF variants [17].

KAIST VIO dataset [8] serves as a benchmark of Visual-Inertial Odometry methods. The authors show a benchmark test of various visual-inertial odometry algorithms on NVIDIA Jetson platforms showing an analysis of resource usage and RMSE (Root Mean Squared Error) of absolute trajectory error considering image resolution.

In these works, the localization methods are tested on publicly available benchmarks, usually based on a custom sensor set. We can observe that these studies do not consider reliability aspects in non-ideal real-world situations, robustness to degraded sensor data, and the effects of external disturbances and feedback control coupling. These imperfections inherently present in any UAV flight in real-world conditions negatively affect the ability of the state-of-the-art VIO approaches to output a sensible pose estimation.

Outlier rejection techniques of the VIO methods that are based on the RPE metric often cannot identify correspondences that cause large orientation errors. A method reformulating the reprojection process is proposed by [18] to reject these keypoints. This method classifies correspondences as inliers and outliers by separating the orientation and translation error components into different dimensions.

Because of the issues raised, this paper identifies critical conditions for a reliable flight under VIO localization and proposes workarounds and compensations for situations in which such conditions cannot be achieved. Also, we propose a modification of multirotor UAV control and estimation system to integrate the VIO. For the UAV control, we rely on the well-documented and open-source MRS System [19] that has been actively used by many researchers in the aerial robotics field. This system allowed the experimental verification of methods published in [3] [20] [21] [22] [23] that could benefit from the proposed VIO-MRS complex.

B. Contributions

The main contributions are as follows:

- integration of VIO approaches into the UAV control system to enable applications without GNSS access in real-world conditions;
- trajectory shaping technique that modifies the dynamics of an input trajectory to reduce motion blur and aggressive tilting motion during which a large part of image features is lost;
- evaluation of pose estimation precision w.r.t. RTK ground truth under challenging conditions in an outdoor field with distant image features during the dark evening or direct sun;
- reliability analysis of the VIO methods in the feedback of the SE(3) controller and identification of the failure cases;
- optimal placement of the camera considering the feature distribution in the camera image that is often degraded by the high contrast of direct sunlight.

II. Visual-Inertial Odometry Algorithms

The state-of-the-art VIO methods can be divided into two principal categories — filter-based and optimization-based approaches. The filter-based methods generally rely on a variant of the Extended Kalman Filter (EKF) for estimating the camera pose. The IMU measurements of accelerations and angular rates are used for the state propagation of the filter based on a non-linear model, while the update step fuses poses obtained by image feature matching. An example of filter-based VIO algorithms is S-MSCKF (Stereo Multi-State Constraint Kalman Filter) algorithm [24] and ROVIO (Robust Visual-Inertial Odometry) algorithm [25], [26].

Optimization-based methods jointly optimize the residuals of image and IMU data measurements to obtain the state estimates. There are several optimization-based approaches such as SVO (Semi-Direct Visual Odometry) [27], VINS-Fusion (Visual-Inertial System Fusion) [28] or OKVIS (Open Keyframe-Based Visual-Inertial SLAM) [29]. Judging by the results published in [28], VINS-Fusion provides a more precise pose estimation than OKVIS.

According to the prior survey, three VIO algorithms have been chosen for the detailed analysis in real-world conditions in this work:

- The filter-based S-MSCKF algorithm.
- The semi-direct SVO method, which combines feature-based and direct methods with robust probabilistic depth estimation aided by motion priors from the IMU.
- The optimization-based VINS-Fusion algorithm, which is a stereo and multi-sensor extension of VINS-Mono.

A. S-MSCKF—Stereo multi-state constraint kalman filter

The S-MSCKF [24] is a stereo variant of the MSCKF [30] [31] [32] approach designed explicitly for UAVs. The authors show that the S-MSCKF achieves robustness with a modest computational budget for aggressive, three-dimensional maneuvering, and fast flights with speeds reaching up to $17.5\,\mathrm{m\,s^{-1}}$. A UAV with S-MSCKF localization can be deployed in both indoor and outdoor environments and also allows indoor/outdoor transitions. This method is not suitable for long-distance flights as the uncertainty of the position estimate grows as the UAV travels, which is caused by not observable global position and yaw. However, the authors demonstrated during an agile flight of over $700\,\mathrm{m}$ that the error is only $3\,\mathrm{m}$, which is sufficient for most typical applications.

B. SVO—Semi-direct visual odometry

SVO [33] is an optimization-based visual odometry algorithm that uses a direct method to track features. The method minimizes the photometric error between pixels corresponding to the projected zone of the same 3D point using feature-based methods for joint structure optimization from motion (SfM). Furthermore, the robust probabilistic depth estimation algorithm enables tracking pixels on weak corners and edges. The authors emphasize the high computation speed of the direct tracking approach, which is possible thanks to the absence of feature extraction and matching steps. The extension [27] of the original algorithm includes generalization for multiple-camera systems, motion prior term added to the optimized cost function, and edge features use. These additions further improve the accuracy and especially the robustness to aggressive motion.

C. VINS-Fusion—Visual inertial navigation system

VINS-Fusion [34], [35] developed as an extension of VINS-Mono [36], is based on optimized multi-sensor fusion for theses sensor combinations: stereo camera; stereo camera + IMU; monocular camera + IMU. The system uses IMU preintegration and feature point observations to achieve precise device self-location. In contrast to other state-of-the-art VIO algorithms, VINS-Fusion continuously calibrates the extrinsic parameters between the camera and IMU and allows loop closures to further reduce drift. An online temporal calibration feature also aims to achieve high accuracy in time offset calibration and system motion estimation [37]. The authors demonstrate that VINS-Fusion achieves locally accurate and globally drift-free pose estimation. Besides that, the authors indicate that the framework can fuse sensor data with different settings in a unified pose graph optimization and is generalized to use other sensors besides cameras and IMU.

III. VIO Approaches Designed for UAV Deployment in Demanding Real-world Conditions

This part of the paper is focused on aspects required for the deployment of VIO approaches using such low-cost and light-weight sensory setup onboard UAVs in demanding real-world conditions to facilitate transitions of these systems from simulations and laboratories to real UAV missions. We will tackle difficulties of challenging light conditions that may change from direct sun-light to darkness instantaneously due to the transition between indoor and outdoor and obstructions of the sun by objects appearing in the proximity of UAV. The obstacles in the workspace and changing weather conditions may also introduce UAV motion disturbances and vibrations, caused by aerodynamic effects of flying close to objects and wind gusts that are not present in laboratory conditions.

A. Camera orientations

We have theoretically and also experimentally found that the camera orientation influences the number of visual features that can be detected in the image stream during different phases of flight, as well as the ability to track these features under aggressive maneuvers even more in the real environment. Due to some mechanical constraints of real robots, such as the required placement of the camera between the front legs of the UAV, the pitch angle of the camera changes the part of the scene that is captured. Four pitch angles are evaluated in this paper: $0°$ (forward), $10°$, $30°$, $90°$ (downward). The position of the camera as well as all the mounting orientations in both real and simulated UAV platforms are shown in Fig. 2.

The advantage of the forward orientation of the camera is the number of available features during translation motions. However, this setup might suffer from rapid illumination changes and high contrast, especially in the outdoor environment in sunny weather. Besides that, fast rotations in the yaw angle might lead to losing track of many features.

The ratio of ground features with a pitched camera is higher, and their detection is possible even from higher altitudes. Thus the pitched camera partially avoids the unreliable and unstable features in the distance and in the sky. More ground features are successfully tracked between frames during fast rotations than with the forward-looking camera.

The down-looking orientation gives a considerable advantage in stable feature detection during challenging lighting conditions. Ground features are easy to track even during fast yaw rotations. However, fast translation and roll/pitch motions at low altitudes are challenging due to rapid scene changes, as many features are lost between frames. Most importantly, taking off is a challenging task for the down-looking camera due to features being too close to the camera. However, if the UAV is taking off from the feature-rich ground, this problem is not significant.

Similar observations led manufacturers of commercial drones to equip their systems with a set of cameras (in some setups up to 12) pointing to all directions to increase reliability in the challenging conditions mentioned above. The proposed work is aimed at designing a less sensory demanding setup relying on a single stereo camera with IMU for deployment in real environments, which could allow us to

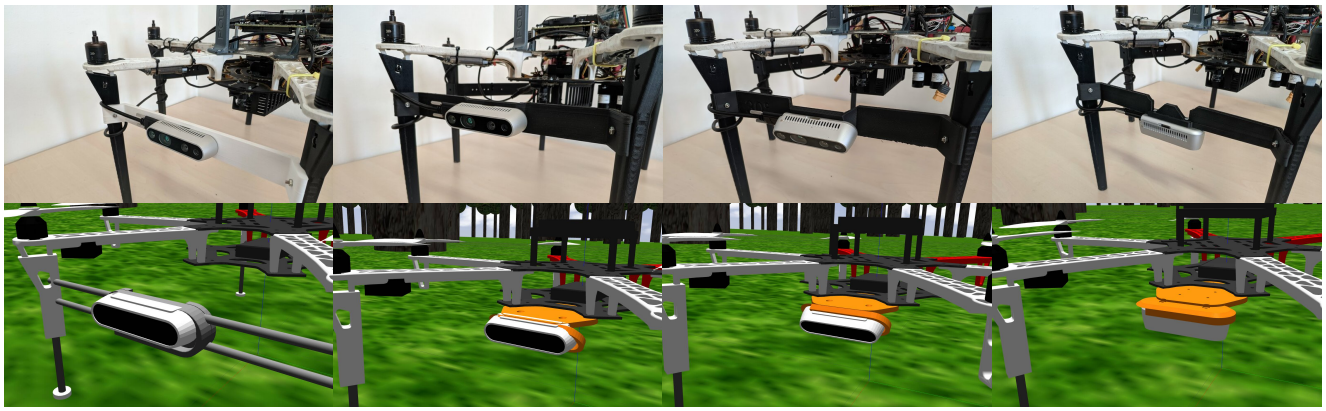

Fig. 2: Camera mounting position between the front legs of the real UAV (top row) and the simulated (bottom row) UAV model. The orientations of the camera considered in this paper are depicted from left to right in this order: $0°$, $10°$, $30°$ and $90°$.

reach the size in the order of centimeters of fully autonomous UAVs in near future.

B. Integration into the MRS UAV control system

The MRS UAV system, designed by our team for experimental verification of multi-robot approaches, is composed of the control pipeline and state estimation shown in Fig. 3. The *Mission planner* module generates desired trajectories based on the specific mission or task, in this case, the trajectories are square-shaped sequences of positions with sampling corresponding to the desired velocities. These sequences are processed in the *MPC tracker* [38], which generates a dynamically feasible full-state reference based on the dynamic model of the UAV. The *SE(3) controller* employs geometric state feedback to produce thrust and attitude reference for the *Attitude controller* embedded in the Pixhawk autopilot, which in turn controls the speed of the motors propelling the UAV.

The *State estimation* module fuses various sensor measurements in a bank of Kalman filters to obtain hypotheses of lateral position, altitude, and orientation. After a valid hypothesis is found, the full-state estimate for the *SE(3) controller* feedback is formed. If no valid hypothesis exists, the control system performs a swift emergency landing, to prevent dangerous motion resulting from diverging state estimate. This safety feature is critical when testing VIO algorithms in feedback in challenging conditions, in which the algorithms are expected to fail often.

The VIO output is integrated as another sensor entering the *State estimation* module among GNSS, RTK, IMU, barometer, and laser rangefinder. A Kalman filter with a point-mass model up to the 2^{nd} derivative provides such estimate by propagating the accelerations from IMU through the model with position and velocity, if available, corrections from the VIO. The orientation must be supplied by the IMU because the delay introduced by the VIO computation, image capture, and transfer would destabilize the delay-sensitive attitude controller. The control system requires a state estimate at a specific rate (MRS UAV system works at $100\,\text{Hz}$), while VIO algorithms provide measurements at the rate of the camera or IMU. During the update step of the filter, the model state can be propagated to a requested time at desired rate regardless of the VIO rate.

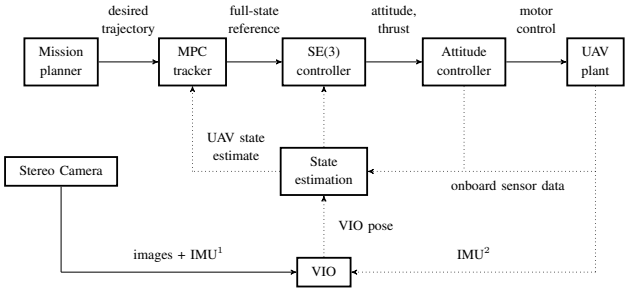

Fig. 3: Diagram of the control pipeline. The reference trajectory serves as a setpoint for the MPC tracker, which outputs a command for the non-linear $SE(3)$ controller. The non-linear controller produces the orientation and thrust reference for the embedded attitude controller. IMU^1 symbolizes the variant when the IMU data comes from the same source as images, e.g, the RealSense D435i camera. IMU^2 means that the IMU data is from the flight control unit (built-in IMU), e.g., in the Gazebo simulator.

C. Trajectory shaping

A typical approach to generating trajectories involves the equidistant sampling of a geometric path to obtain a trajectory with a constant velocity profile. These are valid trajectories that are accepted by the MPC tracker, which transforms them into a reference that respects the dynamic constraints of the UAV, but the reference might deviate from the original trajectory. Moreover, the maximum acceleration of the UAV is so high that it induces aggressive tilts of the UAV, which negatively impacts the accuracy of VIO due to the disappearance of a large part of tracked features from the Field of view (FoV) of the camera. Fig. 4a shows how the UAV tracks a simple square-shaped trajectory with a constant velocity profile.

The proposed trajectory shaping method includes constraints of VIO approaches into the UAV motion planning and adapts the trajectory according to the UAV dynamics so that the motion is respecting the requirements of reliable usage of the VIO-based control mechanism in real-world conditions. First, critical points where the UAV motion (mainly acceleration) exceeds motion constraints required by a particular VIO approach are found in the trajectory.

Then, an iterative approach of adding trajectory samples in the proximity of these points is applied and the motion constraints are consequently evaluated and verified using the state estimation and motion prediction approaches in [38]. The smoothed trajectory that respects the constraints of VIO approaches and the tracking of this trajectory by the UAV are shown in Fig. 4b. Details on the VIO motion constraints specification and determination can be found in the following sections.

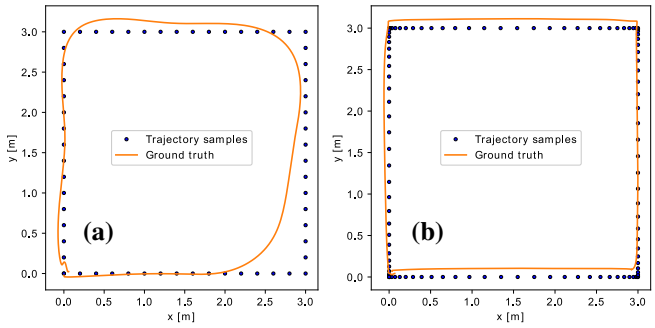

Fig. 4: Trajectory with constant velocity profile (a) violates acceleration constraints during the UAV flight. The UAV platform used in this flight is described in Sec. IV-C. The smoothed trajectory (b) is constrained to $0.4\,\mathrm{m\,s^{-2}}$, which improves tracking precision, and prevents aggressive tilting that would result in losing image features.

IV. Meaningful Evaluation of Vision-based Approaches in Demanding Real-world Conditions

A crucial aspect of comparing localization methods, which is one of the key contributions of this paper, is evaluating the translation and rotation drifts, i.e., the deviation of estimated position from ground truth, over a period of time. This section defines the metrics used to evaluate the selected state-of-the-art VIO methods w.r.t. precise ground truth, describes the hardware and software of the experimental platform that was used to carry out the experiments in real-world conditions, and introduces the specific scenarios that were used for the comparison of the algorithms to provide fair and meaningful (w.r.t. real deployment) evaluation.

A. Ground truth

The ground truth for all experiments with the UAV platform was obtained using an RTK GNSS receiver mounted on the UAV (Fig. 5a) in combination with a fixed ground base station (Fig. 5b) that was sending corrections to the receiver. Thanks to these corrections and phase shift analysis, the RTK system achieves horizontal precision of up to $1\,\mathrm{cm}$ in the most precise mode of operation — RTK FIX. However, when the receiver and the base station do not see enough common satellites, the mode of operation can degrade into a worse precision RTK FLOAT solution with a standard deviation of up to $0.5\,\mathrm{m}$. Vertical positioning has two times larger error than horizontal measurements, and as described in Sec. IV-C the UAV has other sensors that accurately measure its altitude. Thus, the vertical component of the ground truth is not used for the evaluation of the algorithms. The accuracy of orientation estimated by the VIO algorithms is also not considered explicitly in the evaluation as the orientation error implicitly causes translation error during movement [39]. Moreover, to fully measure the orientation of the UAV, at least 3 RTK receivers would have been mounted on the UAV, which was not feasible with the platform used to conduct the experiments.

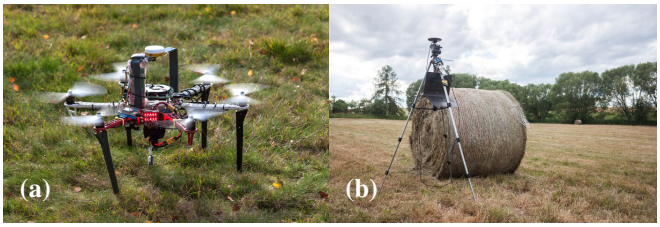

Fig. 5: The RTK receiver on top of the UAV (a) and the RTK base station (b) sending corrections for the UAV.

B. Evaluation metrics

The experiments conducted to evaluate the performance of the VIO algorithms focus on the evaluation of the lateral error, i.e., the x and y components of the UAV position in the reference frame defined by the pose of the UAV at the start of the experiment. Two metrics that are commonly used for evaluating localization methods — the ATE (absolute trajectory error) and RPE (relative pose error) — as defined in [39] were used to compare the VIO algorithms during real deployment.

The VIO odometry and the ground truth trajectories in time are defined as a series of poses $\mathbf{P}_1, ..., \mathbf{P}_n \in SE(3)$ and $\mathbf{Q}_1, ..., \mathbf{Q}_n \in SE(3)$, respectively. As the VIO and RTK positions are usually in the different inertial systems, the Horn method [40] was adopted to align the trajectories prior to the evaluation. The transformation $\mathbf{S}$ is used to align the VIO trajectory $\mathbf{P}_i$ to the ground truth trajectory $\mathbf{Q}_i$ giving absolute trajectory error $\mathbf{F}_i$ at time step i:

$$\mathbf{F}_i := \mathbf{Q}_i^{-1}\mathbf{S}\mathbf{P}_i. \tag{1}$$

1) ATE—Absolute trajectory error: The ATE metric is used as the primary evaluation metric, comparing the whole estimated trajectory with the reference ground truth. The metric is given by the RMSE (Root Mean Square Error) calculated over the previously aligned trajectory:

$$\mathrm{ATE}(\mathbf{F}_{1:n}) := \left(\frac{1}{n} \sum_{i=1}^{n} \|\mathrm{trans}(\mathbf{F}_i)\|^2 \right)^{\frac{1}{2}}, \tag{2}$$

where $\mathrm{trans}(\mathbf{F}_i)$ symbolizes the translation component of the trajectory error $\mathbf{F}_i$.

2) RPE—Relative pose error: The VIO algorithms often have a local drift in the set of IMU and camera measurements. This type of error is reflected in the RPE metric, which is calculated similarly to ATE over the previously aligned trajectory. Contrary to ATE, RPE captures the local drift by evaluating the error over a fixed time interval Δ. The local pose error is calculated as:

$$\mathbf{E}_i := \left(\mathbf{Q}_i^{-1}\mathbf{Q}_{i+\Delta}\right)^{-1}\left(\mathbf{P}_i^{-1}\mathbf{P}_{i+\Delta}\right). \quad (3)$$

A total of $m = n - \Delta$ local pose errors is obtained from n pairs of $\mathbf{P}_i$, $\mathbf{Q}_i$ poses. Then, RMSE is calculated over all m errors $\mathbf{E}_i$:

$$\mathrm{RPE}(\mathbf{E}_{1:n}, \Delta) := \left(\frac{1}{m}\sum_{i=1}^{m}\left\|\mathrm{trans}(\mathbf{E}_i)\right\|^2\right)^{\frac{1}{2}}. \quad (4)$$

The value of the interval Δ is set to $1\,\mathrm{s}$ in all experiments presented in this paper.

C. UAV platform

The UAV platform (Fig. 6) that was used for the experiments is based on a DJI F550 hexacopter frame equipped with E310 DJI motors. PixHawk 4 flight control unit handles the low-level control of attitude and attitude rate of the UAV with the help of measurements from the integrated IMU. In addition, Pixhawk contains a barometer that is fused with accelerations from the IMU and range measurements from the downward-facing laser rangefinder to estimate the altitude of the UAV.

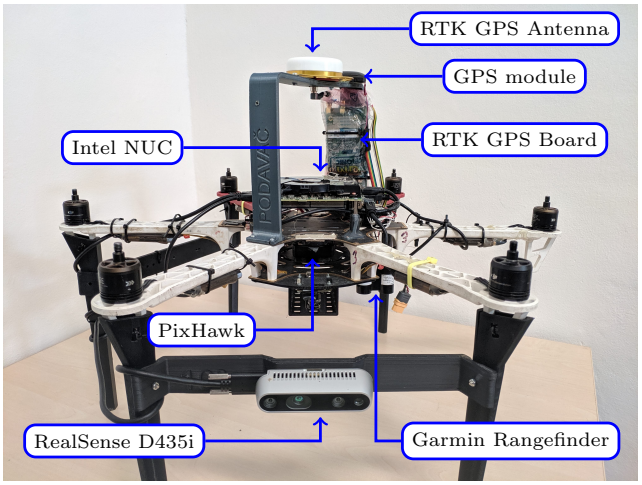

Fig. 6: The UAV platform that was used for the experimental evaluation.

On top of the UAV is mounted the RTK GNSS receiver, which in tandem with the stationary base station provides position estimates with $1\,\mathrm{cm}$ precision. These position estimates are used as the ground truth as described in Sec. IV-A.

The continuous support along with easy setup, small form factor, a wide range of possible operating modes, and mainly meeting all requirements of all considered VIO approaches were the reasons for choosing the Intel RealSense D435i camera for the experimental evaluation of the VIO algorithms. RealSense D435i is a compound camera consisting of rolling shutter RGB $1920\times1080@30\,\mathrm{Hz}$ sensor, stereo pair of global shutter greyscale $1280\times720@30\,\mathrm{Hz}$ sensors, and Bosch BMI055 IMU unit. In the presented experiments, only the stereo camera and IMU in RealSense were used and the system can be operated with any other camera satisfying the requirements of a particular VIO method.

All onboard software runs on the Intel NUC 8i7BEH computer that is capable of running 8 threads at $2.7\,\mathrm{GHz}$ base clock frequency.

D. Gazebo simulator

The UAV platform relying on the above-mentioned MRS system, along with the whole sensor suite, may be simulated in the realistic Gazebo simulator [41] including the PX4 flight control stack to facilitate the safe integration of new methods into the control pipeline. The PX4 model allows parametrizing the IMU parameters, most importantly the noise density and random walk values of the measured angular velocities and linear accelerations. As such, the simulated IMU is a realistic model of the IMU of the Pixhawk flight controller onboard the UAV platform. The RTK does not need to be simulated, as the simulator outputs the exact position of the UAV model in the world, which is used as ground truth instead of the RTK.

The camera plugin allows simulation of the RealSense D435i camera with added noise to emulate the challenging visual conditions. However, the images from the simulated camera are not degraded by insufficient illumination and direct sunlight, which we are interested to evaluate the most. Also, the motion blur, autoexposure, and nonlinearities of the lens are not modeled. Although the simulator is a quick and cheap way to reject parameter combinations that provide mediocre results even under favorable conditions, real-world experiments are necessary to provide meaningful results.

E. Testing scenarios

The considered VIO algorithms were first compared in simulations with various combinations of camera orientations, frame rates, resolutions, and flight heights to find the optimal parameters for further experiments with the real UAV platform. Then the algorithms were evaluated onboard a UAV using the ATE and RPE metrics during demanding experiments, where the UAV was flying a square trajectory in an outdoor environment with challenging lighting conditions and distant features. Such trajectory enabled repeatability and imposed sharp turns as well as straight segments that both can be problematic using different VIO methods. The last set of experiments that were conducted both in simulation and onboard with the real UAV, evaluates the reliability of the algorithms when used in the feedback of the control loop of the UAV as better results of metrics Sec. IV-B obtained by a passive data gathering do not necessarily imply better feedback control performance.

A trajectory of a square shape with a total length of $80\,\mathrm{m}$ was generated for the evaluation. The velocity of UAV flight influences the distance that the image features move between individual camera frames. To analyze the influence of the velocity on the reliability of the algorithms, the experiments are conducted on trajectories generated with $1\,\mathrm{m\,s^{-1}}$, $2\,\mathrm{m\,s^{-1}}$ and $5\,\mathrm{m\,s^{-1}}$ velocities.

V. Experiments

This section first presents simulation experiments aimed at finding optimal camera parameters. Then, selected VIO

algorithms are compared using data gathered by onboard sensors of real UAVs. Lastly, the performance of the VIO approach in control feedback is analyzed in simulations and real outdoor environments.

A. Frame rate optimization

In this test, different FPS rates are tested for $90°$ camera orientation to evaluate the feature tracking performance during rapid movement. Generally, the higher the frame rate, the higher the computational time. However, the higher frame rate reduces the distance of feature movement between frames, which could improve the performance during fast movements. Thus, the desired UAV velocity is $5\,\mathrm{m\,s^{-1}}$ in this experiment to highlight the impact of increased frame rate. The camera resolution is set to 640×360 and the UAV altitude is kept at $3\,\mathrm{m}$. The results of this experiment are summarized in Table I.

Camera FPS	UAV velocity $[\mathrm{m\,s^{-1}}]$	S-MSCKF		SVO		VINS-Fusion	
		ATE	RPE	ATE	RPE	ATE	RPE
30	5	1.4281_2	0.1974_2	2.2822	0.3562	9.2172_1	0.6869_1
60	5	$\mathbf{0.3141}_1$	$\mathbf{0.1534}_1$	0.4384	0.1048	**0.0567**	**0.0712**
90	5	1.0513_1	0.9427_1	**0.4095**	**0.0912**	x	x

TABLE I: Results from the frame rate evaluation experiment in the simulation environment. Each scenario has been repeated 5 times with the same parameters to verify the robustness and precision. The value in the table is the median from these five trials. The subscript, if present, indicates the count of successful test repetitions giving the calculated values. If not present, all repetitions were successful. The x symbol means none of the tests were successful. The best frame rate for each algorithm is in bold.

The S-MSCKF algorithm struggles with fast feature changes at $30\,\mathrm{Hz}$. Increasing the frame rate to $60\,\mathrm{Hz}$ improves both metrics, but at $90\,\mathrm{Hz}$ the error starts to rise again as the algorithm approaches the limit of the available CPU resources.

The SVO results proved its authors' claim that the higher camera frame rate reduces the computational cost per frame. This allows operations with low errors even for scenarios with $5\,\mathrm{m\,s^{-1}}$ UAV velocity. Both $60\,\mathrm{Hz}$ and $90\,\mathrm{Hz}$ rates improved the precision of the SVO result, but the increase from $30\,\mathrm{Hz}$ to $60\,\mathrm{Hz}$ had a much higher impact (ATE decreased more than 5 times) than the increase from $60\,\mathrm{Hz}$ to $90\,\mathrm{Hz}$ (ATE decreased by $7\,\%$).

Similarly to S-MSCKF, VINS-Fusion performed best when the frame rate was increased from $30\,\mathrm{Hz}$ to $60\,\mathrm{Hz}$ but did not converge at $90\,\mathrm{Hz}$.

B. Evaluation using data gained onboard of real UAVs outdoor

In this experiment, the VIO algorithms were compared on the data captured by onboard sensors during tracking of trajectories with $0.5\,\mathrm{m\,s^{-1}}$, $1\,\mathrm{m\,s^{-1}}$ and $2\,\mathrm{m\,s^{-1}}$ desired velocities. During these experiments, direct sunlight significantly influences the camera. The comparison is repeated for all four camera orientations from Sec. III-A to see if higher pitch angles improve the performance by not pointing into the sun. Furthermore, IMU measurements are severely degraded by the vibrations induced by UAV propellers during flight. The accelerometer noise parameters of VINS-Fusion and S-MSCKF had to be increased, otherwise, their estimates started diverging.

Camera orientation	UAV velocity $[\mathrm{m\,s^{-1}}]$	S-MSCKF	SVO	VINS-Fusion
$0°$	0.5	**0.5533**	1.6090	1.7885
$0°$	1	**0.6878**	2.0226	1.8431
$0°$	2	**0.3553**	2.5472	0.6605
$10°$	0.5	**0.2797**	2.7256	0.4704
$10°$	1	**0.6952**	2.4049	0.9081
$30°$	0.5	x	5.5443	**0.5744**
$30°$	1	x	4.8295	**1.2863**
$90°$	0.5	**0.7793**	9.1213	3.0201_1
$90°$	1	**0.4793**	6.0972	5.3223_1

TABLE II: ATE results (in meters) of camera orientation evaluation experiment using three different datasets from real UAV flights. The value in the table is the median from these 3 trials. The subscript, if present, indicates the count of successful test repetitions giving the calculated values. If not present, all repetitions were successful. The x symbolizes that none of the repetitions were successful. The best algorithm with each camera orientation and UAV velocity is in bold.

Table II indicates the results of algorithms on datasets taken by the real UAV. S-MSCKF algorithm clearly outperformed the other two algorithms in all tests, except in the $30°$ camera orientation.

VINS-Fusion, the best candidate from the simulation experiments, worked well in all trials except the $90°$ camera orientation. The proximity of all visible features during takeoff seems to misalign the VINS-Fusion scale at startup, giving worse ATE results.

The SVO underperformed in every test case, which seems to be due to the poor quality of the camera images. Even though enough features are detected, the scale of the trajectory is not correct and the loosely-coupled IMU configuration cannot improve the estimation performance.

Surprisingly, S-MSCKF failed to converge with the $30°$ pitch angle of the camera, since the D435i camera suffered from occasional flickering, especially under outdoor sunlight. This was caused by an issue in the auto-exposure controller of the camera, which has been fixed since the dataset creation. Although it was a hardware problem of the camera, it shows that S-MSCKF is more sensitive to fast exposure changes than the other tested algorithms.

In general, all camera orientations worked well on the real UAV datasets. The advantage of the $10°$ pitch of the camera is the absence of direct sunlight on the camera while still keeping most of the features in front of the UAV. This improvement is especially prominent in the results of the VINS-Fusion.

C. Feedback in simulation

In this test, the VIO algorithms are tested in the feedback of the UAV control system in simulation. The results for $0°$, $10°$ and $30°$ pitch angles of the camera were not vastly different in prior simulation tests due to the absence of sudden exposure changes and other visual issues that are

present in the real-world dataset from Sec. V-B. That is why only $0°$ and $90°$ orientations are tested in the feedback.

SVO has proved its robustness in a higher frame rate, so $60\,\mathrm{Hz}$ rate is used. S-MSCKF and VINS-Fusion algorithms did not perform well with high frame rates in Sec. V-A, hence they are evaluated at $30\,\mathrm{Hz}$ camera rate.

The altitude is set to $5\,\mathrm{m}$ for the $90°$ camera orientation and $3\,\mathrm{m}$ for the $0°$ camera orientation, which should achieve the best performance based on initial simulations. All results are summarized in Table III.

Camera orientation	UAV velocity $[\mathrm{m\,s^{-1}}]$	S-MSCKF		SVO		VINS-Fusion	
		ATE	RPE	ATE	RPE	ATE	RPE
$0°$	1	0.6473	0.0871	0.6182	0.0610	**0.3369**	**0.0451**
$0°$	2	**0.3202**	0.0967	0.7480	0.0781	0.4145	**0.0944**
$0°$	5	0.4518	0.1390	1.2714	0.0994	**0.3961**	**0.0773**
$90°$	1	0.0647	0.0504	0.3470	0.1041	**0.0402**	**0.0294**
$90°$	2	0.1185	0.0630	0.2481	0.0973	**0.0378**	**0.0371**
$90°$	5	0.2437	0.1426	0.3894	0.1131	**0.1078**	**0.0631**

TABLE III: The results of feedback control test in the simulation environment. Each scenario has been repeated 3 times to avoid outliers. The value in the table is the median from these 3 trials. The best scenario accomplished for all trials for each algorithm and every camera orientation is in bold.

In general, the best precision was achieved with $90°$ camera orientation and lower velocities. But, the UAV could not take off with the VIO control feedback due to the lack of features while staying on the ground. So the UAV has to take off using an additional odometry sources, such as GNSS, to initialize the VIO. The $0°$ camera orientation results are slightly worse than the $90°$ camera orientation due to the faster disappearance of image features during fast translation and yaw movement.

VINS-Fusion achieved an almost driftless trajectory with $90°$ camera orientation, which makes it the top candidate for the real deployment. The stability is assured by using measurements from both visual features and IMU in the optimization process. S-MSCKF results show that it also performs well with the $90°$ camera orientation. The stability of the UAV is satisfactory among all velocities for $90°$ camera orientation. The EKF-based approach of features and IMU fusion improves the stability during rapid movement changes, such as during accelerations when features are changing faster. SVO precision for $90°$ camera orientation is better than the results for $0°$ camera orientation. Contrary to the other two algorithms, the SVO relies primarily on the features. The IMU measurements are used only for motion corrections, improving the result. The number of features decreases rapidly during aggressive maneuvers, resulting in positional drift, which is partially compensated by the higher frame rate.

D. Feedback control with the real UAV

After verifying the feasibility of integrating the algorithms into the feedback control in the simulated scenario, the algorithms were tested on a real UAV. All tests are performed at the $30\,\mathrm{Hz}$ frame rate and the default altitude set to $3\,\mathrm{m}$. Except for the $90°$ camera orientation with the $1\,\mathrm{m\,s^{-1}}$ UAV

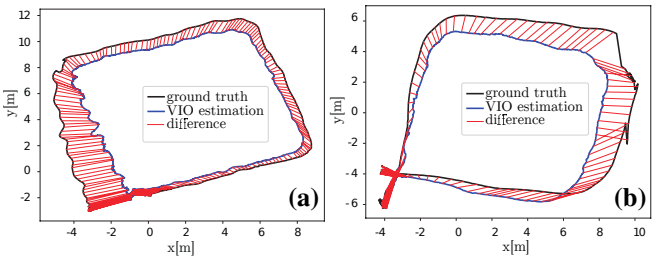

Fig. 7: Trajectories flown with SVO in the feedback of the control system with $0°$ camera orientation. ATE of $1.3395\,\mathrm{m}$ was achieved on trajectory (a) with $0.5\,\mathrm{m\,s^{-1}}$ desired velocity. Faster trajectory (b) with $1\,\mathrm{m\,s^{-1}}$ desired velocity reached $1.3038\,\mathrm{m}$ ATE.

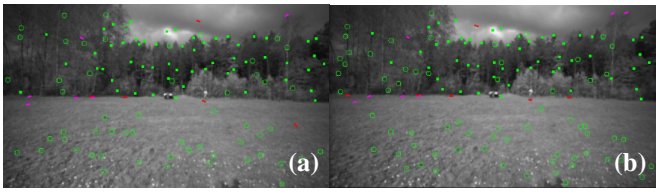

Fig. 8: Left (a) and right (b) camera images with SVO features during the feedback experiment with $0°$ camera orientation.

velocity, where the altitude is set to $5\,\mathrm{m}$ to facilitate feature tracking during maneuvers involving aggressive tilting.

S-MSCKF algorithm presented satisfying results in the experiment in Sec. V-B and thus it was possible to test it in the feedback loop of the UAV control system. SVO algorithm was not as successful, but it was stable enough to test its performance. Unfortunately, VINS-Fusion could not be safely tested in the feedback loop, because it was unstable. Hence only S-MSCKF and partly SVO results are presented. Top-down snapshots of the experiments are shown in Fig. 1.

Despite the volatile performance during the real dataset experiments, SVO pose estimation was usable in the feedback loop, as shown in Fig. 7. The camera images with the features detected by the algorithm can be seen in Fig. 8. However, the position error was huge due to inaccurate estimation of the metric scale.

S-MSCKF flight with $0°$ camera orientation is shown in Fig. 9a. The relatively constant light conditions during the whole flight and minimized camera rapid movements/rotations provide a stable amount of features as shown in Fig. 10. Surprisingly, the flight with $3\,\mathrm{m\,s^{-1}}$ velocity shown in Fig. 9b was successful with minimal drift, despite high acceleration values. Consequently, S-MSCKF might be better on fast flights, as the authors also demonstrated on their custom camera setup.

VI. Conclusions

This paper provides the first comprehensive study of performance of VIO algorithms under challenging conditions of outdoor deployment of fully autonomous UAVs in high-contrast scenes with direct sunlight. To achieve a reliable performance of such a lightweight vision system, a trajectory shaping approach based on iterative sampling was proposed to improve the amount of tracked features by reducing the tilting of the UAV. It was also empirically verified that the negative influence of sunlight is reduced by tilting the camera

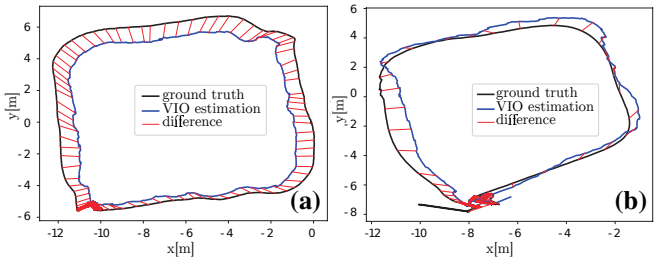

Fig. 9: Trajectories flown with S-MSCKF in the feedback of the control system. Figure (a) shows trajectory for $0°$ camera orientation, $1\,\mathrm{m\,s^{-1}}$ desired velocity with $0.7574\,\mathrm{m}$ ATE. Figure (b) shows trajectory for $90°$ camera orientation, $3\,\mathrm{m\,s^{-1}}$ desired velocity with $0.4617\,\mathrm{m}$ ATE.

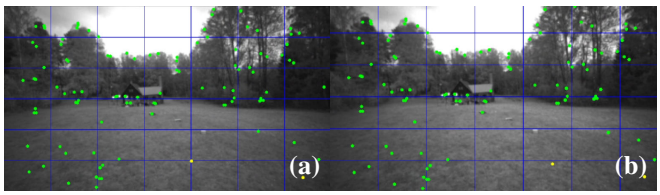

Fig. 10: Left (a) and right (b) camera images with S-MSCKF features during the feedback experiment with $0°$ camera orientation.

in the pitch angle, which also increases the number of stable image features.

Selected VIO algorithms were integrated into the feedback of the open-source MRS UAV control system. All of the tested algorithms provided pose estimates suitable for stable flight in simulation, where the $90°$ camera orientation achieved the best results according to the used metrics. As expected, with higher UAV velocity, slightly worse results were achieved, but the safe velocity for all tested algorithms is up to $2\,\mathrm{m\,s^{-1}}$, which is sufficient for most intended applications.

In the real-world the algorithms struggled with sunlight and propeller-induced vibrations, which add noise to IMU measurements. Unfortunately, the VINS-Fusion algorithm was not stable enough to be tested directly in the feedback loop of the real UAV, although it achieved the best results in simulations. In the feedback loop of the real UAV control system, the S-MSCKF algorithm achieved the best results from tested algorithms in these challenging conditions. The advantage of using the S-MSCKF algorithm is that the implementation based on the Kalman filter combines images and IMU messages resulting in satisfying estimation precision during fast movements. On the other hand, the UAV state is observable only during motion which introduces a position error during hovering. The SVO algorithm is more sensitive on light conditions because it relies more on captured images than S-MSCKF, but it achieves better performance in slower motions and hovering.

Acknowledgments

This research was supported by by CTU grant no SGS20/174/OHK3/3T/13, by Technology Agency of the Czech Republic (TACR) project No. FW03010020, by the Czech Science Foundation (GAČR) under research project No. 20-29531S, by OP VVV funded project CZ.02.1.01/0.0/0.0/16 019/0000765 "Research Center for Informatics", by project no. DG18P02OVV069 in program NAKI II, by the European Union's Horizon 2020 research and innovation program AERIAL-CORE under grant agreement no. 871479.

References

[1] F. Nekovář, J. Faigl, and M. Saska, "Multi-tour set traveling salesman problem in planning power transmission line inspection," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 6196–6203, October 2021.

[2] G. Silano, J. Bednar, T. Nascimento, J. Capitan, M. Saska, and A. Ollero, "A multi-layer software architecture for aerial cognitive multi-robot systems in power line inspection tasks," in *2021 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2021, pp. 1624–1629.

[3] M. Petrlík, T. Báča, D. Heřt, M. Vrba, T. Krajník, and M. Saska, "A robust uav system for operations in a constrained environment," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2169–2176, 2020.

[4] R. Meshcheryakov, A. Salomatin, D. Senchuk, and A. Shirokov, "Scenario of search, detection, and control of invasive plant species using unmanned aircraft systems," *Agriculture Digitalization and Organic Production*, vol. 245, pp. 259–270, 2021.

[5] W. Qian, Y. Huang, Q. Liu, W. Fan, Z. Sun, H. Dong, F. Wan, and X. Qiao, "Uav and a deep convolutional neural network for monitoring invasive alien plants in the wild," *Computers and Electronics in Agriculture*, vol. 174, p. 105519, 2020.

[6] K. C. Vivaldini, T. H. Martinelli, V. C. Guizilini, J. R. Souza, M. D. Oliveira, F. T. Ramos, and D. F. Wolf, "Uav route planning for active disease classification," *Autonomous Robots*, vol. 43, no. 5, pp. 1137–1153, 2018.

[7] S. Aggarwal and N. Kumar, "Path planning techniques for unmanned aerial vehicles: A review, solutions, and challenges," *Computer Communications*, vol. 149, pp. 270–299, 2020.

[8] J. Jeon, S. Jung, E. M. Lee, D. Choi, and H. Myung, "Run your visual-inertial odometry on nvidia jetson: Benchmark tests on a micro aerial vehicle," *IEEE Robotics and Automation Letters*, vol. 6, pp. 5332–5339, 2021.

[9] S. Kohlbrecher, O. von Stryk, J. Meyer, and U. Klingauf, "A flexible and scalable slam system with full 3d motion estimation," in *2011 IEEE International Symposium on Safety, Security, and Rescue Robotics*, 2011, pp. 155–160.

[10] J. Zhang and S. Singh, "Loam: Lidar odometry and mapping in real-time." in *Robotics: Science and Systems*, vol. 2, 2014, p. 9.

[11] Z. Liang, H. Shen, T. Chen, C. Gu, and Q. Song, "Research on the localization of unmanned flight vehicle based on the monocular vision," *Advances in Guidance, Navigation and Control*, vol. 644, pp. 3269–3280, 08 2022.

[12] B. Patel, T. D. Barfoot, and A. P. Schoellig, "Visual localization with google earth images for robust global pose estimation of uavs," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 6491–6497.

[13] J. Engel, V. Usenko, and D. Cremers, "A photometrically calibrated benchmark for monocular visual odometry," *arXiv preprint arXiv:1607.02555*, 2016.

[14] M. Burri, J. Nikolic, P. Gohl, T. Schneider, J. Rehder, S. Omari, M. Achtelik, and R. Siegwart, "The euroc micro aerial vehicle datasets," *The International Journal of Robotics Research*, vol. 35, 01 2016.

[15] J. Delmerico and D. Scaramuzza, "A benchmark comparison of monocular visual-inertial odometry algorithms for flying robots," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 2502–2509.

[16] A. Quattrini Li, A. Coskun, S. M. Doherty, S. Ghasemlou, A. S. Jagtap, M. Modasshir, S. Rahman, A. Singh, M. Xanthidis, J. M. O'Kane *et al.*, "Experimental comparison of open source vision-based state estimation algorithms," in *International Symposium on Experimental Robotics*. Springer, 2016, pp. 775–786.

[17] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 3354–3361.

[18] R. Duan, D. P. Paudel, C. Fu, and P. Lu, "Stereo orientation prior for uav robust and accurate visual odometry," *IEEE/ASME Transactions on Mechatronics*, pp. 1–11, 2022.

[19] T. Baca, M. Petrlik, M. Vrba, V. Spurny, R. Penicka, D. Hert, and M. Saska, "The mrs uav system: Pushing the frontiers of reproducible research, real-world deployment, and education with autonomous unmanned aerial vehicles," *Journal of Intelligent & Robotic Systems*, vol. 102, no. 26, pp. 1–28, May 2021.

[20] L. Rocha, M. Aniceto, I. Araújo, and K. Vivaldini, "A uav global planner to improve path planning in unstructured environments," in *2021 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2021, pp. 688–697.

[21] T. Baca, D. Hert, G. Loianno, M. Saska, and V. Kumar, "Model predictive trajectory tracking and collision avoidance for reliable outdoor deployment of unmanned aerial vehicles," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018.

[22] M. Petrlík, V. Vonásek, and M. Saska, "Coverage optimization in the cooperative surveillance task using multiple micro aerial vehicles," in *2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)*, 2019, pp. 4373–4380.

[23] R. Pěnička, J. Faigl, M. Saska, and P. Váňa, "Data collection planning with non-zero sensing distance for a budget and curvature constrained unmanned aerial vehicle," *Autonomous Robots*, vol. 43, pp. 1937–1956, 2019.

[24] K. Sun, K. Mohta, B. Pfrommer, M. Watterson, S. Liu, Y. Mulgaonkar, C. J. Taylor, and V. Kumar, "Robust stereo visual inertial odometry for fast autonomous flight," *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 965–972, 2018.

[25] M. Bloesch, S. Omari, M. Hutter, and R. Siegwart, "Robust visual inertial odometry using a direct ekf-based approach," in *2015 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2015, pp. 298–304.

[26] M. Bloesch, M. Burri, S. Omari, M. Hutter, and R. Siegwart, "Iterated extended kalman filter based visual-inertial odometry using direct photometric feedback," *The International Journal of Robotics Research*, vol. 36, no. 10, pp. 1053–1072, 2017.

[27] C. Forster, Z. Zhang, M. Gassner, M. Werlberger, and D. Scaramuzza, "Svo: Semidirect visual odometry for monocular and multicamera systems," *IEEE Transactions on Robotics*, vol. 33, no. 2, pp. 249–265, 2017.

[28] T. Qin, J. Pan, S. Cao, and S. Shen, "A general optimization-based framework for local odometry estimation with multiple sensors," *arXiv preprint arXiv:1901.03638*, 2019.

[29] S. Leutenegger, S. Lynen, M. Bosse, R. Siegwart, and P. Furgale, "Keyframe-based visual-inertial odometry using nonlinear optimization," *The International Journal of Robotics Research*, vol. 34, 02 2014.

[30] A. I. Mourikis and S. I. Roumeliotis, "A multi-state constraint kalman filter for vision-aided inertial navigation," in *Proceedings 2007 IEEE International Conference on Robotics and Automation*. IEEE, 2007, pp. 3565–3572.

[31] M. Li and A. I. Mourikis, "High-precision, consistent ekf-based visual-inertial odometry," *The International Journal of Robotics Research*, vol. 32, no. 6, pp. 690–711, 2013.

[32] M. Li and A. Mourikis, "Optimization-based estimator design for vision-aided inertial navigation," in *Robotics: Science and Systems*, 2023, pp. 241–248.

[33] C. Forster, M. Pizzoli, and D. Scaramuzza, "Svo: Fast semi-direct monocular visual odometry," in *2014 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2014, pp. 15–22.

[34] T. Qin, S. Cao, J. Pan, and S. Shen, "A general optimization-based framework for global pose estimation with multiple sensors," *ArXiv*, vol. abs/1901.03642, 2019.

[35] T. Qin, J. Pan, S. Cao, and S. Shen, "A general optimization-based framework for local odometry estimation with multiple sensors," *ArXiv*, vol. abs/1901.03638, 2019.

[36] T. Qin, P. Li, and S. Shen, "Vins-mono: A robust and versatile monocular visual-inertial state estimator," *IEEE Transactions on Robotics*, vol. 34, no. 4, pp. 1004–1020, 2018.

[37] T. Qin and S. Shen, "Online temporal calibration for monocular visual-inertial systems," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 3662–3669.

[38] T. Baca, D. Hert, G. Loianno, M. Saska, and V. Kumar, "Model predictive trajectory tracking and collision avoidance for reliable outdoor deployment of unmanned aerial vehicles," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 1–8.

[39] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of rgb-d slam systems," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2012, pp. 573–580.

[40] B. K. Horn, "Closed-form solution of absolute orientation using unit quaternions," *Optical Society of America*, vol. 4, no. 4, pp. 629–642, 1987.

[41] N. Koenig and A. Howard, "Design and use paradigms for gazebo, an open-source multi-robot simulator," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Sendai, Japan, Sep 2004, pp. 2149–2154.