International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. 14, No.1 (Sep-2023)

http://dx.doi.org/10.12785/ijcds/140183

Robust IMU-Monocular-SLAM For Micro Aerial Vehicle Navigation Using Smooth Variable Structure Filter

Elhaouari Kobzili[1], Ahmed Allam[2] and Cherif Larbes[1]

[1]*Electronic Department,National Polytechnic School, 10 Avenue des frères Oudek, ElHarrach 16200, BP 182, Algiers, Algeria*
[2]*Automatic Department,National Polytechnic School, 10 Avenue des frères Oudek, ElHarrach 16200, BP 182, Algiers, Algeria*

Received 16 May. 2022, Revised 30 Jul. 2023, Accepted 29 Aug. 2023, Published 22 Sep. 2023

Abstract: The autonomous navigation of a micro aerial vehicle (MAV) relies faithfully on the capacity of the localization and the building map of the explored environment. This hard task is known as simultaneous localization and mapping (SLAM). To overcome this problem, many approaches have been proposed based on a variety of sensors. The most popular are those based on monocular vision. But practically all the monocular SLAMs (Mono SLAM) suffer from the scale drift due to the difficulty of depth estimation. To avoid this limitation, the use of a second sensor is crucial to retrieving a metric pose to navigate safely. The Mono-SLAM problem has been resolved by many authors as a problem of filtering or optimization. In this paper, we propose a new SLAM scheme based on a robust filter named Smooth Variable Structure Filter (SVSF). The main advantage of our solution compared to previous solutions is the use of an Inertial Measurement Unit (IMU) associated with a single camera. The different results of the IMU-Mono-SLAM obtained from simulation and experimentation on a well-known dataset prove the reliability, robustness, and accuracy of the proposed solution (SVSF-SLAM) compared to the classical approach (EKF-SLAM).

Keywords: Monocular vision, Robust filter, Simultaneous localization and mapping, Autonomous navigation, Sensors fusion

1. INTRODUCTION

The capability of localization of an unmanned aerial vehicle (UAV) is a crucial task to navigate autonomously in the explored area [1], [2]. This task became more and more delicate in the absence of the environment map. This problem has been treated by a well-known field named as simultaneous localization and mapping. This domain has been seeing a huge amount of works. However, the SLAM technique was resolved using many sensors, but the most attractive solution is the visual SLAM, especially in the case of a single camera. The earliest approaches proposed to resolve monocular SLAM (Mono-SLAM) showed that it can exceed stereo vision SLAM in term of result quality. Beyond of this reason, the use of a monocular camera is very suitable in the robotics field. It can be easily installed, and respond to the constraint of low-cost systems. Moreover, the reduced volume of a single camera carries to realize a compact system. Many techniques have been used to provide an accurate localization within a good map. We can divide these techniques into two parts, filtering [3], [4], [5], [6], [7], [8] and optimization methods [9], [10], [11], [12]. Our work can be classified as a mono-SLAM by filtering method. We follow the idea of Civera et al [8], which is based on the classical solution EKF mono SLAM [13] with other features parameterization. Instead of coded features (points) on three coordinates (x,y,z), will be defined with six parameters (x,y,z,theta,phi,rho) as in [8]. In this case, the depth between a point and the camera is defined by its inverse form (rho) in order to involve this point in the filtering process in the first apparition (immediate initialization). The results shown by the authors encourage us to do a deep exploration of the filtering method. However, the use of Extended Kalman Filter (EKF) into a Mono-SLAM scheme presents many drawbacks, such as quadratic complexity, the high sensitivity in the presence of false associations and linearization of prediction and observation models. The hard nonlinearity and modeling errors lead to get an inconsistent solution, except in the particular case of a Gaussian noise. To avoid these shortages, it is possible to use robust filters, as in the work of [14]. In our paper, we suggest using another robust filter (SVSF) [15], which is a closed loop filter (predictor-corrector), developed on the basis of variable structure theory and sliding mode concepts. Our work is based principally on this new filter to elaborate a new SLAM using a monocular camera by exploiting the advantage of the last work realized by Civera et al [8]. Our new technique provides the localization and the map for a MAV accurately with considerable robustness. Our Mono-SLAM provides - as all previous Mono-SLAMs - a trajectory up to scale due to the absence of an appropriate manner to estimate the depth precisely. In order to overcome this limitation, and to get a metric pose, we suggested predicting the MAV localization based on IMU measurements.

The main contributions of this paper are:

- To elaborate a robust IMU-Mono-SLAM based on SVSF.

E-mail address: elhaouari.kobzili@g.enp.edu.dz, ahmed.allam@g.enp.edu.dz, cherif.larbes@g.enp.edu.dz

- To compare, and analyze the performances of EKF and SVSF based IMU-Mono-SLAM in the case of inverse depth parameterization.

This paper is organized as follows : Section. 2 gives the state-of-the-art of the SLAM field. Section. 3 sketches the SLAM designed and an overview of our approach. Section. 4 shows the state estimation using EKF and SVSF. Section. 5 and Section. 6 describe the scenario of simulation and experimentation with some results. In Section. 7, the proposed approach is evaluated in term of real-time constraint. This paper ends with a conclusion and further suggestions.

2. RELATED WORK

The SLAM task can be realized by displacement of the MAV through a sequence of positions and acquiring data using its embedded sensors. These data are processed properly to provide an estimated localization with an eventual map of the explored environment. In the last decade, the SLAM problem was resolved as a filtering problem. The famous solution is based on the Extended Kalman Filter (EKF-SLAM) [3], Fast-SLAM [5], and SLAM based state dependent Ricatti equation (SDRE) [16]. Many types of sensors are used to obtain an accurate localization within the SLAM techniques, but the most convenient in robot applications is the vision sensor. For a long time, the stereo camera is considered as a spine of the visual SLAM. Recently, including monocular camera on SLAM resolution has become a real challenge due to the absence of depth measurement. This kind of SLAM system is solved by two big methods with respect to real-time condition; the first is filtering, and the second is optimization (direct or indirect methods). The previous succeed of EKF on SLAM-based multi-sensors, push Davison et al to elaborate a real-time SLAM module based just on a single camera [13], and building a sparse map of a limited area. This work is followed by Eade and Drummond's efforts [17]. Their approach relies on the enhancement of the Fast SLAM by applying it in the case of a monocular camera, in which they introduce the terminology of inverse depth to do an instant initialization. This issue allows profiting from the feature at the first apparition (at least for orientation). Staying in the same context, Civera et al [8] presented a new scheme of parameterization for features (points) within Mono-SLAM system based on EKF. This manner of representation allows handling uncertainty properly within instant initialization. Recently, a sophisticated SLAMs techniques are elaborated as a concurred to the classical solution (filtering approaches), these techniques are based on optimization which is inspired from the structure from motion (SFM), which is adapted to be executed in real-time by involving bundle adjustment. All of these approaches cannot function in real-time without a laborious selection of frames (key-frame selection). The most popular is parallel tracking and mapping (PTAM) [9]. This Mono-SLAM was developed to operate in a small workspace. It was the first work in which the localization and mapping were split into two threads. The PTAM can build a sparse map of the environment in real-time, it is considered an indirect method, because it is based on features (points) extraction instead of using the whole image. In the optic of Mono-SLAM, Engel et al proposed a new approach named Large-Scale Direct Monocular SLAM (LSD-SLAM) [10]. It is based essentially on optimization directly through pixel intensities instead of bundle adjustment of the extracted features. It is able to build a semi-dense map, which is suited for robotic applications. In the same context, Raul Mur et al were designing a new direct Mono-SLAM named a versatile and accurate monocular SLAM system (ORB-SLAM) [11]. This SLAM permits to provide a localization and a sparse map accurately with considerable efficiency, especially in the case of loop closing. This framework has improved two times. The first one is called ORB-SLAM2 [1], in which the authors use stereo and RGB-D cameras to obtain the trajectory with a metric scale. The second is called ORB-SLAM3 [12], it is a system able to handle visual, inertial and multi-map SLAM with monocular, stereo and RGB-D cameras. By profiting from the advantage of the Convolutional neuronal network on depth estimation Keisuke proposes a sophisticated Mono-SLAM named CNN-SLAM [18], the strength of this latter is on map construction by involving semantic labeling. In the same context, the CNN can be used to estimate the camera pose by training the CNN architecture on images of the environment of navigation. This pose is used with a structural Mono-SLAM, as is in [19]. It is also possible to use CNN to estimate the scale from a single image. After that, the predicted depth can be used with a Mono-SLAM framework to get a metric scale as given by [2]. All the previous Mono-SLAMs provide a trajectory up to scale, because these SLAMs lack depth estimation accuracy, so it is necessary to be aligned with another sensor [20] as IMU. Fortunately, the majority of MAVs are equipped with an IMU, which facilitates the SLAM task. Implicating the IMU measurement into all the SLAM stages is better than considering the Mono-SLAM as a black box. The accuracy of IMU localization is good in a short time, which obliges us to fuse it with another sensor. In fact, it is difficult or almost impossible for the MAV to navigate safely based on just a single camera. However, the use of the IMU to predict a pose is a good manner to reduce the scale drift of a Mono-SLAM. Targeting the filtering scheme to find an estimated pose is justified by the fact that all the stages of the SLAM (tracking, map management, and loop closing) are processed implicitly by the same filter. The idea of a robust SLAM was tackled by many authors, but to stay near the context of our contribution, we suggest those based on the SVSF [15] as the work of [21]. In this paper, the author used the LSD-SLAM as a black box that provides the Cartesian coordinates and the quaternion to define the camera pose. This means, that the IMU measurements were not used effectively for the prediction step. In the same optic, Demimi et al [22], [23] propose another robust SLAM based on SVSF. His work is limited to 2D robot based on odometer and Laser data. In the same context, a robust SLAM is realized using the SVSF and its ameliorated version, adaptive SVSF (ASVSF) [24], [25] based on the odometer and stereo camera [26]. The robust filters are investigated by researchers in the SLAM field by testing the feasibility of the new filters as in [14]. In this work, a robust least square filter (RLSF) is used. Therefore, the derivative of this filter has been

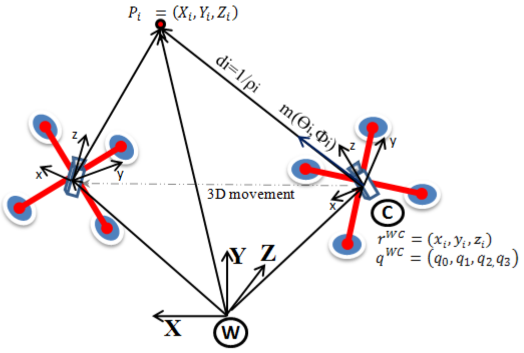


Figure 1. Geometric representation of the designed system

applied to the SLAM to evaluate its performance and applicability. Our proposed solution can be qualified as a Mono-SLAM, but to render it more useful, this Mono-SLAM must be associated with another sensor, which is the IMU, in order to overcome the scale drift. The most critical part of the visual SLAMs based on features of type points, is the matching process. However, in literature, there is a huge development of detector-descriptors [27], [28]. The most appropriate are those dealing with real-time constraint as in [29], [30], [31]. These modern approaches are based on the binary descriptor of patches to provide an identifier of each feature. In our work, we rely on the matching step in our previous descriptor [32] because it is reliable, efficient, and gives a considerable recognition rate. To enhance the SLAM robustness, it is crucial to handle the dynamic environment by taking into consideration the detected object in motion as in [33], [34]. To detect moving objects, semantic segmentation is suitable with the SLAM technique. Moreover, to avoid utilizing these objects in pose and map estimation as in [35], [36], [37]. It is possible to avoid using unstable features to increase the robustness of our SLAM based on dynamic object removal using CNN by involving the model given by [38] to detect and avoid moving objects. As a consequence, the robust filter estimates the pose correctly based just on static point features.

3. METHODOLOGY

A. *The geometric representation*

The problematic treated by our paper is the localization of an MAV navigating through an unknown environment, using its sensors. In this work, it is assumed that the MAV embedded a monocular camera and IMU. The MAV explored the area of navigation by acquiring frames using the camera and the acceleration plus angular rate using IMU. The features are coded using the coordinates of the MAV since the first acquisition by including the inverse depth representation. For more details, see Figure 1.

B. *An overview of the designed system*

The elaborated SLAM system is given by the synoptic diagram of Figure 2. It is divided into four essential steps; prediction, observation, updating, and map management. The details of all the boxes are illustrated by the following subsections.

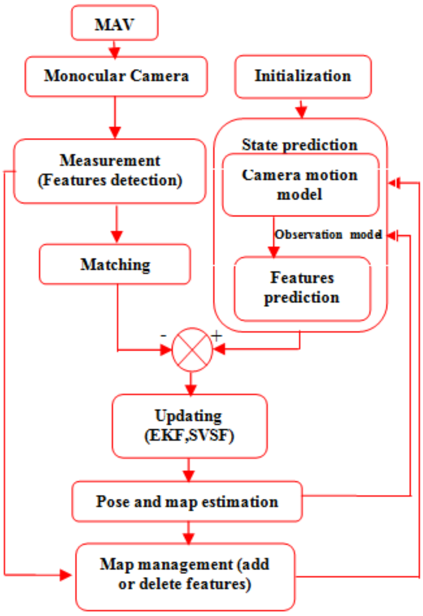


Figure 2. Overview of the designed system

1) *The state vector*

The full state vector is constituted of two parts, the first part is the camera motion vector x_v of 13 elements. These elements are: the Cartesian coordinates of the camera r^{WC}, and the quaternion defined by the camera orientation q^{WC} in the navigation frame. Also, the linear and angular velocities v^W, ω^C. The second part is the features vector. Each point is coded with six parameters (1). The detail of the geometric representation is given by Figure 1.

$$\Upsilon_i = (x_i y_i z_i \theta_i \phi_i \rho_i)^T \tag{1}$$

The Cartesian coordinates of a point can be reproduced by the following equations (2,3).

$$(X_i Y_i Z_i)^T = (x_i y_i z_i)^T + \frac{1}{\rho_i} m(\theta_i, \phi_i) \tag{2}$$

$$m(\theta_i, \phi_i) = (\cos(\phi_i)\sin(\theta_i), -\sin(\phi_i), \cos(\phi_i)\cos(\theta_i)) \tag{3}$$

The vector Υ_i defines the ray realized by the camera position r^{WC} for the first time of a feature observation plus a unit directional vector $m(\theta_i, \phi_i)$ multiplied by the inverse direct distance between the camera center and the point $\rho_i = 1/d_i$. The angles θ_i and ϕ_i are successively the azimuth and the elevation of a feature represented in the world frame. The full state vector is the concatenation of the camera motion vector and the observed feature vectors as given by (4).

$$Y = \left(X_v^T, \Upsilon_1^T, \Upsilon_2^T, ..., \Upsilon_n^T\right)^T \tag{4}$$

2) *Measurement and matching*

To execute the SLAM process, the MAV must be able to acquire and track the appropriate landmarks to locate itself and to build a map of the explored area. Our designed SLAM is a point-based landmark. To keep a track, the selected key points must be stable and invariant. The points are extracted from the images of the monocular camera which is embedded on the MAV. Since it is using a monocular camera, the key points (corners) are defined in two dimensions of coordinates $(u_d, v_d)^T$. In this paper, we based on our work on [32] to detect the invariant points. The key points detected are in a multi scale using FAST and No maxima suppression [39]. For the matching process, we opt to involve the hamming distance between binary descriptors [32] of the point's patches.

The observation model used in our developed SLAM is given by (5).

$$h_\rho^C = R^{CW}\left(\rho_i\left(\left(x_{\,i}y_iz_i\right)^T - r^{WC}\right) + m\left(\theta_i, \phi_i\right)\right) \qquad (5)$$

In fact, based on a monocular camera, we observe the feature in the image plane. However, a pinhole camera model is applied as given by (6).

$$h_u = (u_u, v_u)^T = \left(\left(U_0 - \frac{f}{d_x}\cdot\frac{h_x^C}{h_z^C}\right), \left(V_0 - \frac{f}{d_y}\cdot\frac{h_y^C}{h_z^C}\right)\right)^T \qquad (6)$$

The previous model is valid, when it is assumed a pure projective model. To deal with the distortions of real lenses, a radial distortion model (7) is involved by the same manner of [8].

$$h_d = (u_d, v_d)^T = f\left(u_u, v_u\right) \qquad (7)$$

The equation of measurement is defined by the following equation (8).

$$Y_{m,k} = HZ_k \qquad (8)$$

where Z_k is the measured vector of the observed feature, each point being given by distorted coordinates in the image plane $(u_d, v_d)^T$, and H is the observation matrix given by (9).

$$H = \left(\frac{\partial h_1^C}{\partial Y}, \ldots, \frac{\partial h_i^C}{\partial Y}, \ldots, \frac{\partial h_m^C}{\partial Y}\right)^T \qquad (9)$$

3) *Initialization*

The initialization of the pose is defined by the first position of the MAV with respect to the geometric sets of the sensors (matrix of transformation relies to the Camera and the IMU). The initial velocity and angular rate are set by the user to be close to the operating point. In case of EKF, the probability of pose can be initialized based on the intrinsic characteristic of IMU based on [40]. In case of SVSF, the posterior error $E_{0/0}$ is initialized by taking the maximum limit of the system's incertitude.

The feature initialization follows the same manner of [8], however, from the first observation, it is impossible to detect the feature depth despite many efforts in this problematic [2], [18]. For, this reason, we suppose at first a very weak depth (near to the camera), at least any detected feature, participate at the start in the pose orientation, and smoothly the quality of depth is improved.

The initial parameters of a new observed feature included in the full state vector is given by (10).

$$\hat{\Upsilon}_i\left(\hat{r}^{WC}, \hat{q}^{WC}, h, \rho_0\right) = \left(\hat{x}_i, \hat{y}_i, \hat{z}_i, \hat{\theta}_i, \hat{\phi}_i, \hat{\rho}_i\right)^T \qquad (10)$$

Any observed feature is initialized in the first time by the parameters of the actual estimated pose (11).

$$(\hat{x}_i, \hat{y}_i, \hat{z}_i)^T = \hat{r}^{WC} \qquad (11)$$

The initial angles $\hat{\theta}_i$ and $\hat{\phi}_i$ are obtained by using the following equations (12,13).

$$\left(h_x^W, h_y^W, h_z^W\right) = R^{WC}\cdot(u_u, v_u, 1)^T \qquad (12)$$

$$(\theta_i, \phi_i)^T = \left(\arctan\left(\,h\,\right)_x^W, h_z^W, \arctan\left(-h_y^W, \sqrt{\left(h_x^W\right)^2 + \left(h_z^W\right)^2}\right)\right)^T \qquad (13)$$

The initial value of ρ_0 is defined by suggesting that at first the detected feature is closer to the camera front. The initial value of ρ_0 is not very important in the accuracy of filters [8]. The standard deviation σ_ρ set by knowledge of the explored environment (to define the range of features). The noise covariance matrix R of features is obtained from the standard deviation in the image plan. In case of SVSF the initial posterior error $E_{0/0}$ is set to be the maximum value of the standard deviation in the image plan.

4) *The prediction model*

At first, the camera motion prediction is supposed as a constant angular and linear velocity model given by (14) as in our previous work [41]. In this situation an up to scale trajectory was obtained, but in case of real navigation the pose gets by just a monocular camera is not suited due to the lack of a metric localization despite the use of our work of scale recovering [20], [42]. In this optic, the utilization of a second sensor is not avoidable, however, to considerate the prediction model based on the inertial measurement unit (IMU) is better than involved this sensor measurement at the back end of a Mono-SLAM [20], [42].

$$X_v = \begin{pmatrix} r^{WC}_{k+1} \\ q^{WC}_{k+1} \\ v^{W}_{k+1} \\ \omega^{C}_{k+1} \end{pmatrix} = \begin{pmatrix} r^{WC}_k + \left(v^{WC}_k + V^W_k\right)\Delta T \\ q^{WC}_k \times q\left(\left(\omega^C_k + \Omega^C\right)\Delta T\right) \\ v^W_k + V^W_k \\ \omega^C_k + \Omega^C \end{pmatrix} \qquad (14)$$

The state vector of the camera X_v is constituted of the following terms: r^{WC} is the Cartesian coordinate of the camera center, q^{WC} is the quaternion defined the camera orientation in the world. The linear and angular velocity is v^W, ω^C. This last produces at each step, an impulse of linear velocity $V^W = a^W \Delta T, \Omega^C = \alpha^C \Delta T$. The state prediction is based on the inertial navigation system (INS) mechanization using IMU measurements given by (15).

$$a^b_{IMU} = \left(a_x, a_y, a_z\right), \omega^b_{IMU} = (p, q, r) \qquad (15)$$

Before using this measurement, it must be filtered because it is very noisy and by consequence keep the position reliability. For this reason, we were based on an ameliorated version of a low-pass filter [43]. The Figure 3 gives a sample of the accelerations and angular velocities filtered.

Now, the predicted pose at each step can be calculated based on (16,17) by time integration. Where $Roll_I, Pitch_I, Yaw_I$ are the angles calculated by time integration.

$$a^n_{IMU,k} = C^n_b\,(k-1)\,a^b_{IMU}\,(k) + g^n \qquad (16)$$

$$\omega^n_{IMU,k} = E^n_b\,(k-1)\,\omega^b_{IMU}\,(k) \qquad (17)$$

Where $C^n_b\,(k-1)\,, E^n_b\,(k-1)$ represent the direction cosine, and the rotation rate transformation matrixes between the body and the navigation frame. The terms $a^n_{IMU,k}, \omega^n_{IMU,k}$ represent the acceleration, and the angular velocity vectors in the navigation frame. The angles, roll and pitch can be calculated more accurately based on the complementary filter as in [43] using (18,19), where $aRoll, aPitch$ are the roll and pitch estimated with the accelerometers measurements as given by (20,21), but the yaw angle cannot be evaluated with just the use of the accelerometer due to the definition of z axis in the direction of the gravity.

$$Roll = \mu \cdot Roll_I + (1-\mu) \cdot aRoll \qquad (18)$$

$$Pitch = \mu \cdot Pitch_I + (1-\mu) \cdot aPitch \qquad (19)$$

$$aRoll = arctan2\left(a_y, \sqrt{a_x^2 + a_z^2}\right) \qquad (20)$$

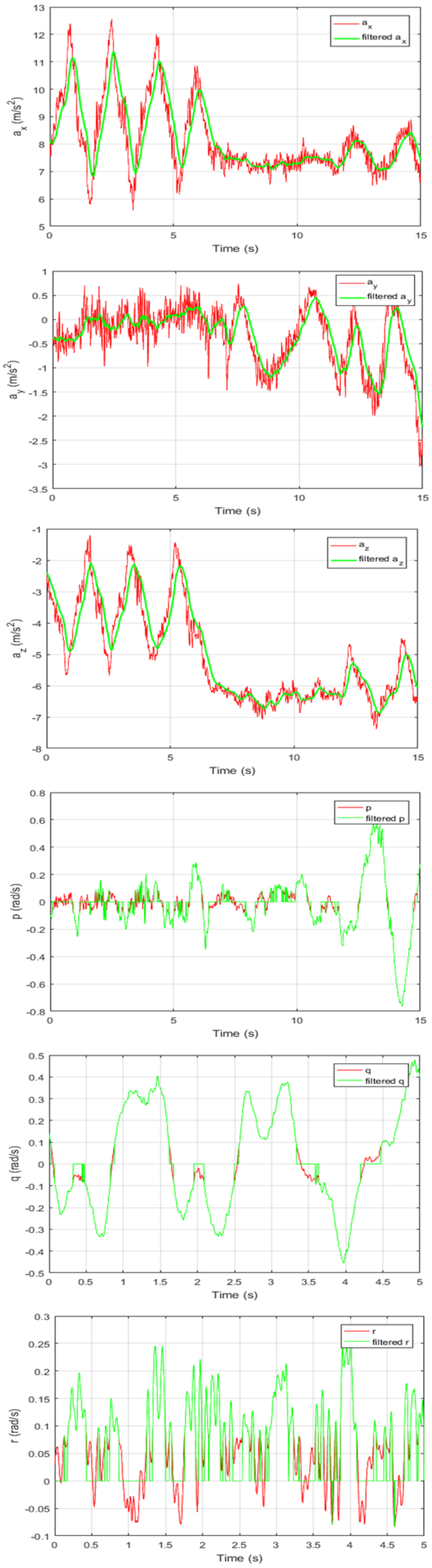


Figure 3. The filtered accelerations and angular velocities

$$aPitch = arctan2\left(-a_x, \sqrt{a_y^2 + a_z^2}\right) \tag{21}$$

The lower the parameter μ the more the Roll/Pitch listens to the accelerometers. We mention that, the gyroscopes are more reliable on the short term, however a high μ is recommended. We notice that the angles $Roll, Pitch, Yaw$ are converted to the quaternion representation. The complete prediction model in the case of the use of IMU measurement is given by (22).

$$X_v = \begin{pmatrix} r_{k+1}^{WC} \\ q_{k+1}^{WC} \\ v_{k+1}^{W} \\ \omega_{k+1}^{C} \end{pmatrix} = f\left(a_{IMU,k}^{n}, q_k^{WC}, \Delta T\right) \tag{22}$$

5) The Map management

During the MAV displacement, the embedded camera observes new features which are added immediately to the map constructed. These points contribute on pose estimation, particularly for the orientation at the start. For each feature, we associate two counters, one for prediction and other for measurement. The following pseudo code sketches the steps of the map management.

```
Initialization
$Prediction\ counter = 0.$
$Measurement\ counter = 0.$
$Number\ of\ features\ detected = 0.$
$number\ max\ per\ frame =$ defined by user.
$Predict\ threshold =$ defined by user.

For each new frame
Detect all features.
For all features belongs the frame
If (the feature is not associated) && (the number of feature detected ¡ number max per frame)
Add the new feature.
$Number\ of\ features\ detected = Number\ of\ features\ detected + 1.$
Else
Delete the new feature.
End
$Predictioncounter = Prediction\ counter + 1.$
$Measurementcounter = Measurement\ counter + 1.$

If (Measurement counter ¡ half Prediction counter)&& (Prediction counter ¿ Predict threshold) Delete the feature.
$Prediction\ counter = 0.$
$Measurement\ counter = 0.$
Else
Maintain the feature.
End.

End.
$Number\ of\ features\ detected = 0.$
End.
```

4. STATE ESTIMATION

The state vector estimation is based on two filters EKF and SVSF. All the stages of the SLAM designed turn into a closed mechanism. The following sub sections will summary the process of the IMU-Mono-SLAM in case of the classical solution (EKF), and in case of our contribution (SVSF).

A. EKF monocular slam

The EKF is involved in all the steps of IMU-Mono-SLAM based on the equations (23-27) of prediction, observation, and updating.

$$\hat{Y}_k = \left(X_v^T + n_k, \Upsilon_{k,1}^T, \Upsilon_{k,2}^T, \cdots, \Upsilon_{k,n}^T\right)^T \tag{23}$$

$$P^- = FP_kF^T + FQF^T \tag{24}$$

With F is the Jacobian matrix of the prediction system, and n_k is the process noise. The matrix P^- defines the prior covariance matrix. The updating of the full state vector is based on (25-27).

$$K_k^{Kalman} = P_k^- H^T \left(HP_k^- H^T + R\right)^{-1} \tag{25}$$

$$Y_k = \hat{Y}_k + K_k^{Kalman}\left(\hat{Y}_k - Y_{m,k}\right) \tag{26}$$

$$P_k = \left(I - K_k^{Kalman} H\right) P^- \tag{27}$$

Where K_k^{Kalman} is the Kalman gain. The matrix P_k defines the posterior covariance matrix. The matrix H is the Jacobian matrix of the observation model. The term $Y_{m,k}$ is the observation vector of features extracted from the acquired images.

B. SVSF monocular SLAM

In 2007 S. Habibi [15] developed on the base of variable structure theory with the concept of sliding mode a new predictor corrector filter named SVSF. The base of this filter is a prior and posterior errors between prediction and observation. In the SVSF formulation, it is supposed to exist of a subspace and a smoothing boundary layer to order the estimation of the state vector element to stay between a bounded region. This assumption carries up the estimation process to be robust and stable as it was proved in [15]. The Figure 4 shows an overview of the SVSF convergence. The major advantages of SVSF are the robustness against perturbation and modeling assumption. The linearization is not necessary, and it does not need a noise model. The limitation of SVSF is the lack of optimality especially in the Gaussian case, the chattering, and the difficulty of defining the superior limits of incertitude.

In the case of SVSF the prediction model is the same of the EKF (1). In case of SVSF, the prediction model linearization is not necessary. The full state vector elements are initialized by Y_0 and the posterior error of estimation is initialized by $E_{0/0}$. The SVSF needs a prior (28) and a posterior (31) errors to calculate the SVSF gain. The observation vector $Y_{m,k/k-1}$ is calculated based on the observation equations defined in the subsection. 3-B.

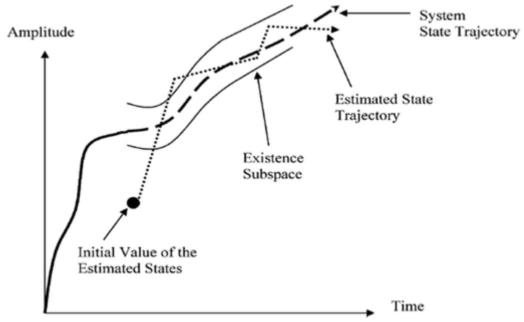

Amplitude
System State Trajectory
Estimated State Trajectory
Existence Subspace
Initial Value of the Estimated States
Time

Figure 4. Principle of SVSF [15]

$$E_{k/k-1} = \hat{Y}_k - Y_{m,k/k-1} \tag{28}$$

The updating of the full state vector in case of SVSF is given by (29-31).

$$K_k^{SVSF} = H^{+}diag\left[(\left|E_{k/k-1}\right|_{Abs} + \gamma\left|E_{k-1/k-1}\right|_{Abs})^{\circ} \; sat(\bar{\psi}^{-1}E_{k/k-1})\right] \cdot [diag(E_{k/k-1})]^{-1} \tag{29}$$

$$Y_{k/k} = \hat{Y}_k + K_k^{SVSF}E_{k/k-1} \tag{30}$$

$$E_{k/k} = \hat{Y}_k - Y_{k/k} \tag{31}$$

Where $E_{k/k-1}$and $E_{k/k}$ represent the prior and the posterior errors, K_k^{SVSF} is the SVSF gains at step k, and H^{+} is the pseudo inverse of the observation matrix which is constructed from the observation matrix of EKF (9) by extracting the rows of just the observed features at the step k. The symbol $\bar{\psi}$ defines the diagonal matrix of smoothing boundary layer widths (32) (The selection of $\bar{\psi}$ reflects the level of uncertainties of the system and the measurement). The parameter $\gamma \in R^{m\times m}, 0 < \gamma \leq 1$ is a diagonal matrix. Its elements define the convergence rate of the state vector elements. It is supposed $\gamma = 0.5$.

$$\bar{\psi}^{-1} = \begin{pmatrix} 1/\psi_{1,1} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 1/\psi_{m,m} \end{pmatrix} \tag{32}$$

The operator ° is the *schur* function (it is an element-by-element multiplication of two vectors). The function *sat* is a saturation (it is possible to use the *sgn* or *tangent* functions to reduce the chattering phenomenon of the state vector elements, but these functions are not suited for real-time processing).

5. SCENARIO OF SIMULATION AND RESULTS

To perform the simulation of the designed IMU-Mono-SLAM, we follow these steps. We suppose that the MAV navigates in a 3D textured environment. The MAV is embedded by two sensors (Camera and IMU). To be near the reality, we suppose that the MAV follows an arbitrary trajectory as given by [40]. During the navigation phase, the camera acquires features according to the model of simulation presented in Figure 5. However, the detected features are a 3D points dispatched randomly around the proposed trajectory. So, to qualify a 3D point as a feature, it must verify the following conditions. The processed point must be in front of the camera within a distance inferior than a fixed distance defined by the user. The point must also be between two angles that represent the camera view angle. Before doing the previous steps, it is necessary to transform the 3D points to the camera reference according to the transformation matrix between the body, the MAV, and the camera frames. Then, a view camera model is applied on a candidate feature in order to reject those outside the image size (image of size 320x240).

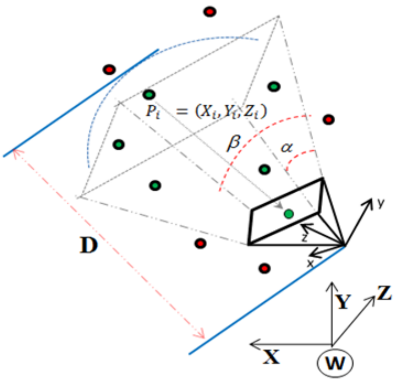


Figure 5. Simulation of the camera model

In the simulation the matching step is ignored despite the sensibility of any SLAM in front of this step, to verify the quality of the proposed approach, it is supposed that the association process is assured based on tracking each acquired feature by a unique identifier.

To render the simulation consistent, the feature is affected by a no Gaussian noise in the image plan. This noise allows to verify the efficiency of the proposed solution. We did not mention the Gaussian simulation, because certainly the EKF is more optimal than SVSF in this case. In fact, any proposed solution must be tested in the worst conditions to confirm the robustness quality. The results of the simulation are given by the Table I and Figure 6. It is observable that the IMU-Mono-SLAM based on SVSF provides a better pose estimation compared to EKF. The SVSF shows a considerable accuracy and robustness against no Gaussian noise compared to EKF. From the Table I it is observable that the root-mean-square error (RMSE) of the obtained map based on SVSF is more consistent than the map built by the EKF due to the concept of sliding mode which provides the robust action against uncertainty. The simulation results give for us, an idea about the performance of the designed SLAM despite neglecting the association process which is considered as an essential task for any SLAM.

6. SCENARIO OF EXPERIMENTATION AND RESULTS

To validate the designed IMU-Mono-SLAM experimentally, we follow this method. We use the data of a

TABLE I. Mean RMSE of the simulated trajectory and the map using EKF and SVSF

	X(m)	Y(m)	Z(m)	XYZ (m)	Map (m)
EKF	0.212	0.195	0.162	0.331	0.302
SVSF	0.104	0.127	0.062	0.175	0.293

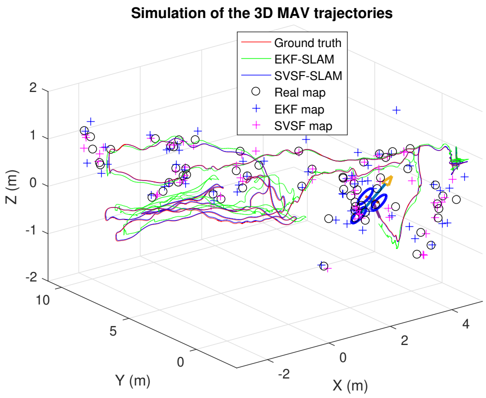


Figure 6. The simulation of the 3D MAV trajectories with maps of Ground truth, EKF and SVSF

MAV (an AscTec Firey hex-rotor helicopter) presented in a well-known dataset [40] which is represented by the picture of Figure 7. The environment of navigation is a large industrial machine hall given by Figure 8. This MAV embedded two monochromes front-down cameras type MT9V034 of a frequency 20 Hz (20 frame/s) and an IMU type ADIS16448 technology MEMS of a frequency 200 Hz. The Figure 9 represents a synoptic configuration of the sensors sets. The standard deviations of noises and biases are mentioned for each sensor within the files provided for the collected data. In the file, we can find also the transformation matrixes of sensors with respect to the body frame. For more precision about all the system of data collection refer to [40]. In our work, we just used a two sensors of this MAV (the couple Camera and IMU).

To provide the experimental comparison between our approach based on SVSF and EKF, we propose the Figure 10 and Figure 11. These figures show the path followed by the MAV in 3D using both filter SVSF and EKF with respect to the real trajectory (ground truth) followed by the robot. The path of IMU-Mono-SLAM based on SVSF is concisely following the ground truth, with small error compared to IMU-Mono-SLAM based on EKF Figure 11. The EKF fails catastrophically with false matching because a bad association in case of EKF carries the filter to instability which is not the case for SVSF filter. In the two cases of SLAM using SVSF or EKF, the building map is sparse, it needs to be improved in term of 3D structure. From Table 6, we learn that the RMSE obtained along the traveled trajectory for SVSF-SLAM is smaller compared to EKF-SLAM. From Table I and Table 6, the overall RMSE for SVSF and EKF in the simulation is better than those presented in experimental results. This is due to the neglecting of a primordial task which is the matching process.

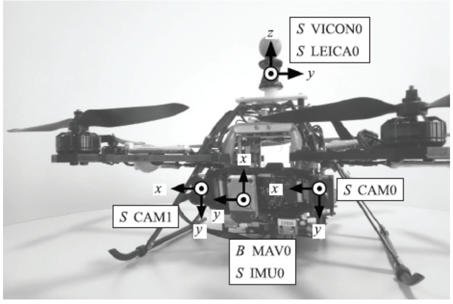


Figure 7. The MAV picture used for dataset collection [40]

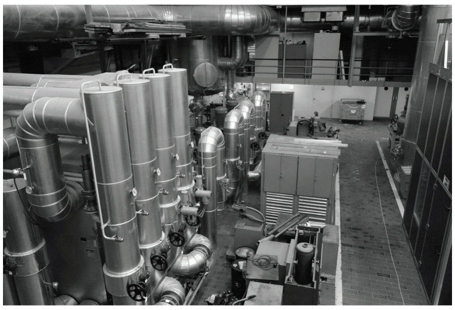


Figure 8. A sample of a frame took from the dataset [40]

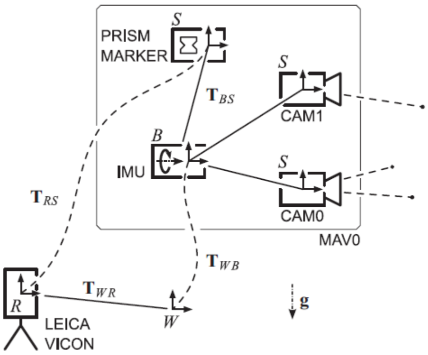


Figure 9. A The synoptic sets of the MAV sensors [40]

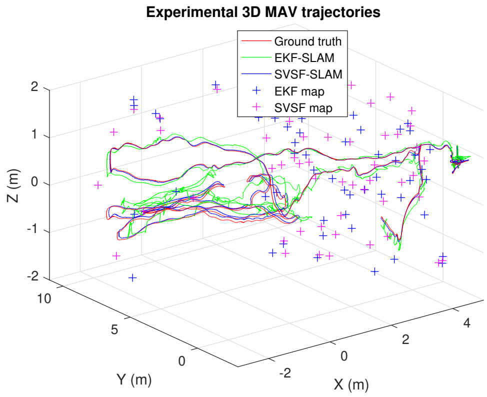


Figure 10. The experimental of the 3D MAV trajectories with maps of Ground truth, EKF and SVSF.

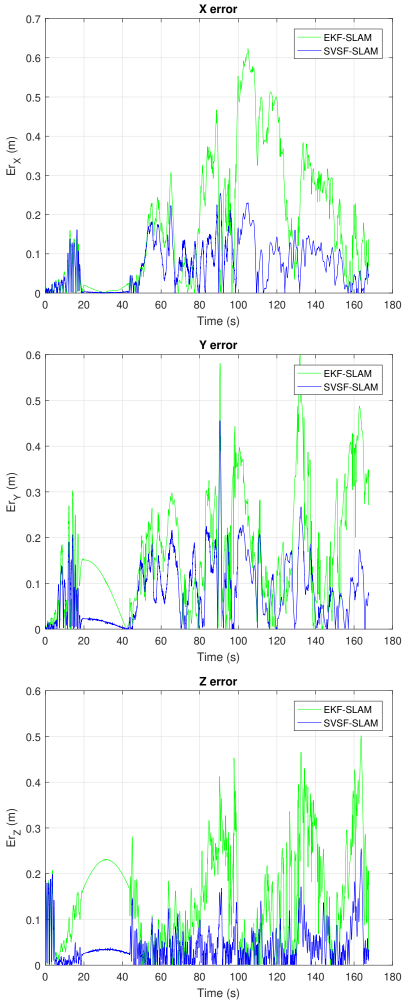


Figure 11. The error of trajectories along X,Y, and Z in case of EKF, and SVSF.

TABLE II. Mean RMSE of the experimental trajectory using EKF and SVSF

	X(m)	Y(m)	Z(m)	XYZ (m)
EKF	0.299	0.254	0.227	0.453
SVSF	0.114	0.129	0.070	0.186

7. THE REAL TIME EVALUATION

Our designed SLAM is running on a computer of the following characteristic: Intel(R) Core (TM), Processor i5 of frequency 2.4 *GHz*, 8 *Go* RAM. The dataset of evaluation [40] provides data with different frequencies of acquisition. However, the frequency of IMU is 200 *Hz* and the camera rate is 20 *Hz*. Our SLAM algorithm functions in real-time with the following assumptions: the image size is 320×240, the measurement vector is limited to 15 features by image, and a full vector size depends on the number max of the measured vector. The latter is the power of the SVSF because the state vector updating is partial. This means that the feature observed is updated, but those not observed did not change. In case of EKF the updating is high computational because at each iteration all the state vector is corrected. The time available to do all the steps of our SLAM is 50 *ms* (about 20 images per second). In case of our SLAM, the time of processing is bonded, we can treat at each iteration a vector of size max 103 elements (13 elements for the camera state vector and 90 elements to code 15 features of size 6 of each one). The processing time depends just on the predefined number of features per image at each iteration. But, in case of EKF, the environment of navigation must be smaller to not exceed the capacity of features supported by the designed algorithm. Therefore, our IMU-Mono-SLAM can be utilized in a large outdoor environment with the assumption of feature's number limitation per image. The disadvantage of our approach resumes on the lack of loop closing sensitivity. The loop closing task is not immediate compared to EKF-SLAM.

8. CONCLUSION

In this paper, we elaborate a new SLAM for an MAV. To this end, a SLAM based on a filtering scheme is enhanced by the fusing of the IMU data and a monocular camera using the SVSF filter mechanism. This new filter was involved to improve the efficiency of the designed SLAM in terms of accuracy and robustness. The IMU data have been used essentially to surmount the problem of scale drift. We compare our solution with available numerical simulations and experimental data [40] and good agreement has been obtained. In our approach, the full state vector is updated for just some elements, which brings us to realize a real-time IMU-Mono-SLAM. The EKF map is limited to a small area of navigation, which is not the case for our approach. Our solution can be easily used in an outdoor environment. Our SLAM suffers from the sensitivity of the loop closing compared to EKF (it closes the loop implicitly). As future work, we suggest using the adaptive version of SVSF (ASVSF) [24], [25] to provide an idea about the uncertainty of the pose and the built map. It is necessary to enhance our SLAM for loop closing. The map obtained is based on points; it is not dense and it cannot be competitive compared to the recent SLAMs. This kind of map limits the reuse for another task of navigation, Therefore, it must be dense for an ulterior autonomous navigation mission.

References

[1] R. Mur-Artal and J. D. Tardós, "Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras," *IEEE transactions on robotics*, vol. 33, no. 5, pp. 1255–1262, 2017.

[2] A. Steenbeek and F. Nex, "Cnn-based dense monocular visual slam for real-time uav exploration in emergency conditions," *Drones*, vol. 6, no. 3, p. 79, 2022.

[3] R. Smith, M. Self, and P. Cheeseman, "Estimating uncertain spatial relationships in robotics," in *Autonomous robot vehicles*. Springer, 1990, pp. 167–193.

[4] M. G. Dissanayake, P. Newman, S. Clark, H. F. Durrant-Whyte, and M. Csorba, "A solution to the simultaneous localization and map building (slam) problem," *IEEE Transactions on robotics and automation*, vol. 17, no. 3, pp. 229–241, 2001.

[5] A. Stentz, D. Fox, and M. Montemerlo, "Fastslam: A factored solution to the simultaneous localization and mapping problem with unknown data association," in *In proceedings of the AAAI national conference on artificial intelligence*. Citeseer, 2003.

[6] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping (slam): Part i," *IEEE robotics & automation magazine*, vol. 13, no. 2, pp. 99–110, 2006.

[7] W. Brink, C. Van Daalen, and W. Brink, "Stereo vision as a sensor for ekf slam," in *22nd Annual Symposium of the Pattern Recognition Association of South Africa*, 2011, pp. 19–24.

[8] J. Civera, A. J. Davison, and J. M. Montiel, "Inverse depth parametrization for monocular slam," *IEEE transactions on robotics*, vol. 24, no. 5, pp. 932–945, 2008.

[9] G. Klein and D. Murray, "Parallel tracking and mapping for small ar workspaces," in *2007 6th IEEE and ACM international symposium on mixed and augmented reality*. IEEE, 2007, pp. 225–234.

[10] J. Engel, T. Schöps, and D. Cremers, "Lsd-slam: Large-scale direct monocular slam," in *European conference on computer vision*. Springer, 2014, pp. 834–849.

[11] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, "Orb-slam: a versatile and accurate monocular slam system," *IEEE transactions on robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.

[12] C. Campos, R. Elvira, J. J. G. Rodríguez, J. M. Montiel, and J. D. Tardós, "Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam," *IEEE Transactions on Robotics*, vol. 37, no. 6, pp. 1874–1890, 2021.

[13] A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse, "Monoslam: Real-time single camera slam," *IEEE transactions on pattern analysis and machine intelligence*, vol. 29, no. 6, pp. 1052–1067, 2007.

[14] S. Mondal and W. K. Chung, "Robust least square filter for simultaneous localization and mapping," in *2020 International Conference on Emerging Frontiers in Electrical and Electronic Technologies (ICEFEET)*. IEEE, 2020, pp. 1–3.

[15] S. Habibi, "The smooth variable structure filter," *Proceedings of the IEEE*, vol. 95, no. 5, pp. 1026–1059, 2007.

[16] A. Nemra, "Robust airborne 3d visual simultaneous localisation and mapping," Ph.D. dissertation, Cranfield University, 2011.

[17] E. Eade and T. Drummond, "Scalable monocular slam," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, vol. 1. IEEE, 2006, pp. 469–476.

[18] K. Tateno, F. Tombari, I. Laina, and N. Navab, "Cnn-slam: Real-time dense monocular slam with learned depth prediction," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 6243–6252.

[19] X. Ban, H. Wang, T. Chen, Y. Wang, and Y. Xiao, "Monocular visual odometry based on depth and optical flow using deep learning," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–19, 2020.

[20] E. Kobzili, C. Larbes, and A. Allam, "Multi-rate robust scale estimation of monocular slam," in *2017 6th International Conference on Systems and Control (ICSC)*. IEEE, 2017, pp. 1–5.

[21] A. Nemra, L. M. Bergasa, E. López, R. Barea, A. Gómez, and Á. Saltos, "Robust visual simultaneous localization and mapping for mav using smooth variable structure filter," in *Robot 2015: Second Iberian Robotics Conference*. Springer, 2016, pp. 557–569.

[22] F. Demim, A. Nemra, and K. Louadj, "Robust svsf-slam for unmanned vehicle in unknown environment," *IFAC-PapersOnLine*, vol. 49, no. 21, pp. 386–394, 2016.

[23] F. Demim, A. Nemra, K. Louadj, Z. Mehal, M. Hamerlain, and A. Bazoula, "Simultaneous localization and mapping algorithm for unmanned ground vehicle with svsf filter," in *2016 8th International Conference on Modelling, Identification and Control (ICMIC)*. IEEE, 2016, pp. 155–162.

[24] S. A. Gadsden and S. R. Habibi, "A new form of the smooth variable structure filter with a covariance derivation," in *49th IEEE Conference on Decision and Control (CDC)*. IEEE, 2010, pp. 7389–7394.

[25] S. A. Gadsden, M. El Sayed, and S. R. Habibi, "Derivation of an optimal boundary layer width for the smooth variable structure filter," in *Proceedings of the 2011 American Control Conference*. IEEE, 2011, pp. 4922–4927.

[26] A. Allam, M. Tadjine, A. Nemra, and E. Kobzili, "Stereo vision as a sensor for slam based smooth variable structure filter with an adaptive boundary layer width," in *2017 6th International Conference on Systems and Control (ICSC)*. IEEE, 2017, pp. 14–20.

[27] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.

[28] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (surf)," *Computer vision and image understanding*, vol. 110, no. 3, pp. 346–359, 2008.

[29] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "Orb: An efficient alternative to sift or surf," in *2011 International conference on computer vision*. Ieee, 2011, pp. 2564–2571.

[30] A. Bellarbi, S. Otmane, N. Zenati, and S. Benbelkacem, "[poster] mobil: A moments based local binary descriptor," in *2014 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. IEEE, 2014, pp. 251–252.

[31] W. Beiyi, Z. Xiaohong, and W. Weibing, "Feature matching method based on surf and fast library for approximate nearest neighbor search," *Integrated Ferroelectrics*, vol. 218, no. 1, pp. 147–154, 2021.

[32] E. Kobzili, C. Larbes, A. Allam, and F. Demim, "3d polynomial interpolation based local binary descriptor," in *International Conference on Computer Science and its Applications*. Springer, 2018, pp. 204–214.

[33] R. Wang, W. Wan, Y. Wang, and K. Di, "A new rgb-d slam method with moving object detection for dynamic indoor scenes," *Remote Sensing*, vol. 11, no. 10, p. 1143, 2019.

[34] J. Chang, N. Dong, and D. Li, "A real-time dynamic object segmentation framework for slam system in dynamic scenes," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–9, 2021.

[35] T. Ran, L. Yuan, J. Zhang, D. Tang, and L. He, "Rs-slam: A robust semantic slam in dynamic environments based on rgb-d sensor," *IEEE Sensors Journal*, vol. 21, no. 18, pp. 20 657–20 664, 2021.

[36] S. Wen, P. Li, Y. Zhao, H. Zhang, F. Sun, and Z. Wang, "Semantic visual slam in dynamic environment," *Autonomous Robots*, vol. 45, no. 4, pp. 493–504, 2021.

[37] Y. Fan, Q. Zhang, Y. Tang, S. Liu, and H. Han, "Blitz-slam: A semantic slam in dynamic environments," *Pattern Recognition*, vol. 121, pp. 108–225, 2022.

[38] D. Bolya, C. Zhou, F. Xiao, and Y. J. Lee, “Yolact++: Better real-time instance segmentation,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 44, no. 2, pp. 1108–1121, 2022.

[39] E. Rosten and T. Drummond, “Machine learning for high-speed corner detection,” in *European conference on computer vision*. Springer, 2006, pp. 430–443.

[40] M. Burri, J. Nikolic, P. Gohl, T. Schneider, J. Rehder, S. Omari, M. W. Achtelik, and R. Siegwart, “The euroc micro aerial vehicle datasets,” *The International Journal of Robotics Research*, vol. 35, no. 10, pp. 1157–1163, 2016.

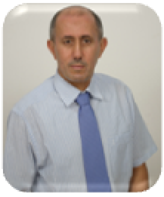
[41] E. Kobzili, C. Larbes, A. Allam, F. Demim, and A. Boucheloukh, “Geometric binary descriptor based monocular slam,” in *2018 3rd International Conference on Pattern Analysis and Intelligent Systems (PAIS)*. IEEE, 2018, pp. 1–6.

[42] E. Kobzili, C. Larbes, and A. Allam, “Robust absolute scale estimation of monocular slam using a combined svsf-ekf strategy for mav navigation,” *The 1st Algerian Multi Conference on Computer, Electrical and Electronic Engineering*, 2017.

[43] C. Treffers and L. van Wietmarschen, “Position and orientation determination of a probe with use of the imu mpu9250 and a atmega328 microcontroller,” 2016.

Elhaouari Kobzili received his BS and MS degrees in automatic from the Polytechnic Military School of Algiers, Algeria in 2006 and 2009, respectively. He is a Doctor at the National Polytechnic School of Algiers, Algeria. His research interests include computer vision, SLAM, robotics, neuronal networks, and embedded architecture.

Ahmed Allam received his BS and MS degrees in automatic from the Polytechnic Military School of Algiers, Algeria in 2010 and 2015, respectively. He is a Doctor at the National Polytechnic School of Algiers, Algeria. His research interests include control and modeling of multi-robots system, data fusion, and SLAM.



Cherif Larbes received his engineering degree in electronics from National Polytechnic School of Algiers, Algeria, in 1985 and his Ph.D. degree from the University of Lancaster, UK, in 1990. He is currently a Full Professor with the Department of Electronics, National Polytechnic School, Algiers, Algeria. His current research interests include intelligent control algorithms, power converters, electrical machine drives and unmanned aerial vehicle (UAV). He has published more than 60 journal and conference papers .