---

**Proceedings Paper:**
Candan, F., Beke, A. and Mihaylova, L. orcid.org/0000-0001-5856-2223 (2023) An interacting multiple model approach based on maximum correntropy student's T filter. In: 2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) Proceedings. 2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 01-05 Oct 2023, Detroit, Michigan, USA. Institute of Electrical and Electronics Engineers (IEEE) , pp. 3314-3321. ISBN 9781665491914

https://doi.org/10.1109/IROS55552.2023.10341366

---

# An Interacting Multiple Model Approach based on Maximum Correntropy Student's T Filter

Fethi Candan[1], Aykut Beke[2] and Lyudmila Mihaylova[1]

***Abstract*— This paper presents a novel approach called the Interacting Multiple Model (IMM)-based Maximum Correntropy Student's T Filter (MCStF), which addresses the challenges posed by non-Gaussian measurement noises. The MCStF demonstrates superior performance compared to the IMM algorithm based on Kalman Filters (KFs) in both simulation environments and real-time systems. The Crazyflie 2.0 nano Unmanned Air Vehicle (UAV) model is used in the simulation validation, and results from 3000 independent Monte Carlo runs are shown. After getting the simulation results under monotonously changed non-Gaussian distribution, their performance results have been compared to each other. The same scenario has been applied in the real-time system using Crazyflie 2.0. Next, results from real-time tests are presented in which the position of Crazyflie 2.0 is estimated online.**

## I. Introduction

In recent years, Unmanned Air Vehicle (UAV) systems have received significant attention in many areas, such as agricultural feeding/spraying, cyber-physical, and civilian applications [3], [12], [32]. Although the number of UAV applications is increasing, many aspects of UAVs are still under development. One of these aspects is positioning the UAV or tracking the desired trajectory under high uncertainty. Considering the trajectory tracking part, linear and nonlinear controllers have been applied to UAVs [7], [11], [16], [33], [34]. However, observing or detecting the UAV's position is another challenging problem in a real-time system because of the uncertain nature of its components, such as sensor quality and communication latency [22], [23].

In this paper, we focus on the localisation problem of a nano UAV under environments with measurement uncertainties. Therefore, a novel position estimation method for the UAV is proposed and investigated. In previous works, different methods are developed, such as conventional, extended, cubature, or unscented Kalman filters (KFs) [26], [35] to estimate the position of a single UAV or of multiple UAVs. The KF and other KF variants have been widely used in autonomous systems, especially in UAVs, unmanned ground vehicles (UGVs), surveillance and communications [4], [10], [24], [27].

*This work was supported by the Ministry of National Education of Turkish Government.

[1]Fethi Candan is with Automatic Control and Systems Engineering, University of Sheffield, Sheffield, S1 3JD, United Kingdom fcandan1@sheffield.ac.uk

[2]Aykut Beke Researcher is with Department of Control Systems Design, ASELSAN, Ankara, 06750, Turkiye abeke@aselsan.com.tr

[1]Lyudmila Mihaylova is with Automatic Control and Systems Engineering, University of Sheffield, Sheffield, S1 3JD, United Kingdom l.s.mihaylova@sheffield.ac.uk

The performance of these filters depends especially on the model parameters and noise variance or covariance values. However, knowing the exact parameter values in most real-time system applications is difficult, which could affect the estimation accuracy. These unknown parameters or uncertainties can be characterised by the noise distributions of the sensor data. These noises are usually represented by the Gaussian distribution. Then, the prediction and estimation are achieved by Kalman filters [8], [9]. However, these distributions are not always Gaussian. As a result of this, the predictions and the estimations cannot give high accuracy of the system state estimation under non-Gaussian.

This paper is organised as follows: Section II presents an overview of related works, Section III describes the novel filtering method, a Maximum correntropy Student's T filter (MCStF), and the Interacting Multiple Model (IMM) approach with MCStFs. Section IV presents performance validation of the IMM-MCStF over simulated data and over real data. A comparison of the proposed IMM-MCStF with an MCKF and a standard KF is presented. Finally, Section V summarises the main results and ideas for future work.

## II. Related Works

In this section, the most commonly used filters under non-Gaussian distributions [9], [14], [18] are reviewed.

Before the explanation of this research, the widely used Student's T, which is a non-Gaussian distribution, and its various degrees of freedom (DoF) ($\upsilon$) are illustrated in Fig. 1.

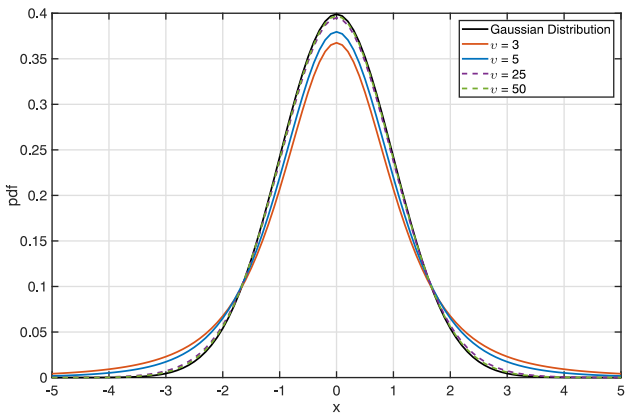


Fig. 1. Illustrations of distribution functions used in this paper (Gaussian and Student's T).

The maximum correntropy Kalman Filter (MCKF) [9] has advantages in dealing with non-Gaussian noises and improves the precision in the tracking and positioning. It

adopts the maximum correntropy criteria (MCC) and shows robustness to environmental changes [13]. In [9], unlike the KF, MCKF is able to perform successfully under the non-Gaussian distribution with a defined error threshold and a small kernel bandwidth. Another research [18] has proposed Multi-Kernel maximum correntropy Kalman filter (MKMCKF). Multivariate non-Gaussian noises and disturbances have been coped with using MKMCKF. In [8] and [17], a Huber-based Kalman filter (HKF) is introduced as a solution to filtering problems in linear systems with the heavy-tailed process and measurement disturbances. This is achieved by minimising a combined *L1* and *L2* norm, where a Huber function is utilised to constrain the residuals. The HKF can be considered a generalised maximum likelihood estimator. However, it should be noted that the HKF's influence function does not redescend, which limits its impact on estimation accuracy [17]. To handle heavy-tailed non-Gaussian noise caused by significant outliers, several maximum correntropy criterion-based Kalman filters (MCCKFs) have been developed. These filters aim to maximize the correntropy of the expected error and residual and can be viewed as maximum posterior estimators. However, the lack of a theoretical foundation for the estimation error covariance matrix weakens the estimation accuracy [14]. Recent studies have addressed the issue of filtering heavy-tailed measurement noise. While [2] and [25] propose variational approximations, [28] suggests a regularisation-based optimisation strategy. However, these approaches presume well-behaved process noise, which is often not the case in practical scenarios such as tracking moving targets. Consequently, these localization strategies encounter challenges in many situations. In [30], improvements for the robustness of Student's T filters under varying degrees of heavy-tailed noise are proposed. The authors suggest a versatile and robust Student's T multi-model strategy, assuming that the unknown degrees of freedom (DoF) parameter follows an interacting Markov process. The multiple Student's T models include the maximum versoria criteria, an efficient optimisation criterion recently employed in the kernel adaptive filtering domain.

Compared with the related works, we proposed a novel approach, the maximum correntropy Student's T filter (MCStF). It has been applied over a simulated environmental setup and on a real-time system under the assumptions for non-Gaussian distributions. Simultaneous consideration of non-Gaussian processes and measurement noises is necessary for most real-time systems. We proposed Maximum correntropy and Student's T maximum versoria criteria. After that, the designed filter is implemented in interacting with multiple models (IMM). It is because the performance of the traditional single model KF, MCKF or MCStF deteriorates when the motion status is not single [13], [21], [31].

### A. Contributions

This paper proposes a novel IMM-MCStF approach with Student's T distributed measurement noises. Unlike the Gaussian distribution, the Student's T distribution parameters must be calculated with an approximate method. The proposed IMM-MCStF approach is based on the Mercer Kernel, and this kernel allows it to MCStF perform accurately in both under Gaussian and non-Gaussian distributions. Moreover, classical KF under Student's T distribution does not have better results than the KF under Gaussian distribution. However, not only MCStF shows similar results to KF under Gaussian distribution, but it also gives it the ability to perform better than the KF under non-Gaussian distributions.

## III. Maximum Correntropy Student's T Filter

The proposed MCStF has the assumption of non-Gaussian distributed measurement noises. In this section, the maximum correntropy criterion (MCC) and Student's T filter to derive the MCStF formulation have been explained deeply. Unlike under Gaussian noises, the traditional Kalman filter may perform dramatically worse when faced with non-Gaussian noises, and impulsive noises stimulate the particular underlying system.

Considering a linear model [9], [18]

$$\begin{bmatrix}\widehat{\mathbf{x}}(k|k-1)\\ \mathbf{y}(k)\end{bmatrix} = \begin{bmatrix}\mathbf{I}\\ \mathbf{H}(k)\end{bmatrix}\mathbf{x}(k) + \boldsymbol{\upsilon}(k), \tag{1}$$

where $\mathbf{x}(k) \in \mathbf{R}^n$ denotes an $n$-dimensional system state vector at discrete time $k$, $\mathbf{y}(k) \in \mathbf{R}^m$ is an $m$-dimensional measurement vector, $\mathbf{I}$ is the $n \times n$ identity matrix, $\mathbf{H}(k)$ is the measurement matrix and $(\widehat{\mathbf{x}}(k|k-1))$ is the predicted state estimate at time $k$. Following the notations of [9], an aggregated noise vector $\boldsymbol{\upsilon}$

$$\boldsymbol{\upsilon}(k) = \begin{bmatrix}-(\mathbf{x}(k) - \hat{\mathbf{x}}(k|k-1))\\ \mathbf{r}(k)\end{bmatrix} \tag{2}$$

comprises both the state error and measurement noise $\mathbf{r}(k)$. The state error covariance matrix ($\mathbf{P}(k|k-1)$) and measurement noise covariance matrix ($\mathbf{R}(k)$) can be written as

$$\mathbf{E}[\upsilon(k)\upsilon^T(k)] = \begin{bmatrix}\mathbf{P}(k|k-1) & 0\\ 0 & \mathbf{R}(k)\end{bmatrix}, \tag{3}$$

where $\mathbf{E}[.]$ denotes the mathematical expectation operator. Equation (3) can be represented in the form

$$\begin{bmatrix}\mathbf{B}_p(k|k-1)\mathbf{B}_p^T(k|k-1) & 0\\ 0 & \mathbf{B}_r(k)\mathbf{B}_r^T(k)\end{bmatrix} = \mathbf{B}(k)\mathbf{B}^T(k), \tag{4}$$

where $\mathbf{B}(k)$ is obtained by using the Cholesky factorisation of the expression on the left. When multiplying both sides of (1) by $\mathbf{B}^{-1}(k)$, we obtain

$$\mathbf{D}(k) = \mathbf{W}(k)\mathbf{x}(k) + \mathbf{e}(k), \tag{5}$$

in the above equation, $\mathbf{D}(k)$ and $\mathbf{W}(k)$ can be written as

$$\begin{aligned}\mathbf{D}(k) &= \mathbf{B}^{-1}(k)\begin{bmatrix}\hat{\mathbf{x}}(k|k-1)\\ \mathbf{y}(k)\end{bmatrix},\\ \mathbf{W}(k) &= \mathbf{B}^{-1}(k)\begin{bmatrix}\mathbf{I}\\ \mathbf{H}(k)\end{bmatrix}.\end{aligned} \tag{6}$$

In (5), the residual error $\mathbf{e}(k)$ is defined as $\mathbf{e}(k) = \mathbf{B}^{-1}(k)\upsilon(k)$. Moreover, it can be written as $\mathbf{E}[\mathbf{e}(k)\mathbf{e}^T(k)] =$

$\mathbf{I}$ in the case of (5), where $\mathbf{I}$. is theunit matrix Therefore, the residual error $\mathbf{e}(k)$ must be white-coloured.
Considering the MCC iterations, it can be written that the $i$th element of $\mathbf{e}(k)$ is: $e_i(k) = d_i(k) - \mathbf{w}(k)\mathbf{x}(k)$. The MCC cost function $J_L$ is in the form:

$$J_L(\widehat{\mathbf{x}}^*(k|k)) = \frac{1}{L}\sum_{i=1}^{L} G_\sigma(d_i(k) - \mathbf{w}_i(k)\widehat{\mathbf{x}}(k|k-1)), \quad (7)$$

with $L = m + n$ being the dimension of $\mathbf{B}(k)$ and $\mathbf{w}_i(k)$ is the row of $\mathbf{W}(k)$. The optimal state estimate $\widehat{\mathbf{x}}(k|k)$ is calculated from:

$$\widehat{\mathbf{x}}(k|k) = \underset{\mathbf{x}(k|k)}{\operatorname{argmax}} J_L(\mathbf{x}^*(k|k)) = \underset{\mathbf{x}(k|k)}{\operatorname{argmax}} \sum_{i=1}^{L} G_\sigma(e_i(k)) \quad (8)$$

also, $\mathrm{G}_\sigma$ denotes the kernel function which is the Gaussian kernel [9]. Unlike the MCKF, $\mathbf{M}_x$ and $\mathbf{M}_y$ need be calculated, as in (9) and applied in Student's T filter to satisfy this condition

$$\begin{aligned}\mathbf{M}_x(k) &= \operatorname{diag}(\mathrm{G}_\sigma(\mathrm{e}_1(k)), \cdots, \mathrm{G}_\sigma(\mathrm{e}_n(k))),\\ \mathbf{M}_y(k) &= \operatorname{diag}(\mathrm{G}_\sigma(\mathrm{e}_{n+1}(k)), \cdots, \mathrm{G}_\sigma(\mathrm{e}_{n+m}(k)))\end{aligned} \quad (9)$$

and these calculated diagonal matrices have been used to find the error covariance $\mathbf{P}(k|k-1)$ and error measurement covariance $\mathbf{R}(k)$ matrices, in Equations (10) and (11)

$$\begin{aligned}\mathbf{P}(k|k-1) = \mathbf{B}_p(k|k-1)\left(\widetilde{\mathbf{M}}_x^{(t-1)}(k)\right)^{-1}\\ \mathbf{B}_p^{\mathrm{T}}(k|k-1),\end{aligned} \quad (10)$$

$$\mathbf{R}(k) = \mathbf{B}_r(k)\left(\widetilde{\mathbf{M}}_y^{(t-1)}(k)\right)^{-1}\mathbf{B}_r^{\mathrm{T}}(k), \quad (11)$$

and

$$\begin{aligned}\widehat{\mathbf{x}}^*(k|k) = \widehat{\mathbf{x}}(k|k-1) + \mathbf{P}(k|k-1)\mathbf{H}(k)^{\mathrm{T}}(\mathbf{S}(k))^{-1}\\ (\mathbf{z}(k) - \mathbf{H}(k)\widehat{\mathbf{x}}(k|k-1)),\end{aligned} \quad (12)$$

$$\mathbf{S}(k) = \mathbf{H}(k)\mathbf{P}(k|k-1)\mathbf{H}(k)^{\mathrm{T}} + \mathbf{R}(k). \quad (13)$$

Considering the MCC-based cost function, iteration $\sigma$ and small positive threshold $\varepsilon$ values have been defined. In ref. [6], [9], the effects of $\sigma$ and $\varepsilon$ have been investigated and compared with different values and

$$\frac{\left\|\widehat{\mathbf{x}}^{*,(t)}(k|k) - \widehat{\mathbf{x}}^{*,(t-1)}(k|k)\right\|}{\left\|\widehat{\mathbf{x}}^{*,(t-1)}(k|k)\right\|} \leq \varepsilon. \quad (14)$$

Equation (14) gives the cost function, which depends on the current state estimate $\widehat{\mathbf{x}}^{*,(t)}(k|k)$ and the previous state estimate $\widehat{\mathbf{x}}^{*,(t-1)}(k|k)$. The $t$ index denotes the number of iterations, and $||.||$ denotes the Euclidean norm. Then, the estimated state value can be obtained as $\widehat{\mathbf{x}}_{k|k} = \widehat{\mathbf{x}}^*_{k|k}$ if the calculated value is ($\leq \varepsilon$) in (14). The covariance matrix can be calculated from

$$\begin{aligned}\mathbf{P}^*(k|k) = \tfrac{\upsilon(k-1)+\Delta^2(k)}{\upsilon(k-1)+d}(\mathbf{P}(k|k-1) - \mathbf{P}(k|k-1)\\ \mathbf{H}^{\mathrm{T}}(k)(\mathbf{S}(k))^{-1}\mathbf{H}(k)\mathbf{P}(k|k-1)).\end{aligned} \quad (15)$$

where

$$\begin{aligned}\Delta^2(k) = (\mathbf{z}(k) - \mathbf{H}(k)\widehat{\mathbf{x}}(k|k-1))^{\mathrm{T}},\\ (\mathbf{S}(k))^{-1}(\mathbf{z}(k) - \mathbf{H}(k)\widehat{\mathbf{x}}(k|k-1)).\end{aligned} \quad (16)$$

In (15), the updated error covariance matrix has been calculated after the necessary condition is done. In this work, a flexible Student's T distribution has been used under heavy-tailed noise. A method including noises with heavy-tailed distributions is proposed in [28], [30], in particular with a Student's T filter and the covariance matrix is represented as:

$$\mathbf{P}(k|k) = \tfrac{\upsilon^*(k)}{\upsilon^*(k)-2}\tfrac{\upsilon(k)-2}{\upsilon(k)}\mathbf{P}^*(k|k). \quad (17)$$

In (17), the updated error covariance matrix of the flexible Student's T filter has been defined. In the equation, $\upsilon$ is the degree of freedom, and $d$ is the dimension of the state vector. In addition, it is obtained as $\upsilon^*(k) = \upsilon(k-1) + d$. After all the mathematical background, the MCStF's implementation pseudocode of the MCStF is shown in Algorithm (1).

**Algorithm 1** The implementation pseudocode for one time-step of the MCStF

**Inputs:** $\hat{\mathbf{x}}(k-1|k-1)$, $\mathbf{P}(k-1|k-1)$, $\mathbf{Q}(k-1)$, $\mathbf{R}(k)$, $\upsilon$, $\varepsilon$, $\sigma$

**Time update:**

1) $\hat{\mathbf{x}}(k|k-1) = \mathbf{F}(k)\hat{\mathbf{x}}(k-1|k-1)$.
2) $\mathbf{P}(k|k-1) = \mathbf{F}(k)\mathbf{P}(k-1|k-1)\mathbf{F}^{\mathrm{T}}(k) + \mathbf{Q}(k-1)$.

**Measurement update:**

1) $\mathbf{B}_p = \mathrm{Chol}(\mathbf{P}(k|k-1))$, $\mathbf{B}_r = \mathrm{Chol}(\mathbf{R}(k))$, $\hat{\mathbf{x}}^{(t=t_0)}(k) = \hat{\mathbf{x}}(k|k-1)$.
2) $\mathbf{M}_x(k) = \operatorname{diag}(\mathrm{G}_\sigma(\mathrm{e}_1(k)), \cdots, \mathrm{G}_\sigma(\mathrm{e}_n(k)))$, $\mathbf{M}_y(k) = \operatorname{diag}(\mathrm{G}_\sigma(\mathrm{e}_{n+1}(k)), \cdots, \mathrm{G}_\sigma(\mathrm{e}_{n+m}(k)))$.
3) $\mathbf{P}(k|k-1) = \mathbf{B}_p(k|k-1)(\widetilde{\mathbf{M}}_x^{(t-1)}(k))^{-1}\mathbf{B}_p^{\mathrm{T}}(k|k-1)$, $\mathbf{R}(k) = \mathbf{B}_r(k)(\widetilde{\mathbf{M}}_y^{(t-1)}(k))^{-1}\mathbf{B}_r^{\mathrm{T}}(k)$.
4) $\mathbf{S}(k) = \mathbf{H}(k)\mathbf{P}(k|k-1)\mathbf{H}(k)^{\mathrm{T}} + \mathbf{R}(k)$, $\widehat{\mathbf{x}}^{*,(t)}(k|k) = \widehat{\mathbf{x}}^{(t)}(k|k-1) + \mathbf{P}(k|k-1)\mathbf{H}(k)^{\mathrm{T}}$ $(\mathbf{S}(k))^{-1}(\mathbf{z}(k) - \mathbf{H}(k)\widehat{\mathbf{x}}(k|k-1))$.
5) $\frac{\left\|\widehat{\mathbf{x}}^{*,(t)}(k|k) - \widehat{\mathbf{x}}^{*,(t-1)}(k|k)\right\|}{\left\|\widehat{\mathbf{x}}^{*,(t-1)}(k|k)\right\|} \leq \varepsilon$, where $\hat{\mathbf{x}}(k|k) = \hat{\mathbf{x}}^{*,(t)}(k|k)$ is performed and go to Step 6 if the termination condition is satisfied; else, Step 2 is returned, and the iteration is continued after setting $t = t + 1$.
6) $\Delta^2(k) = (\mathbf{z}(k) - \mathbf{H}(k)\widehat{\mathbf{x}}(k|k-1))^{\mathrm{T}}(\mathbf{S}(k))^{-1}$ $\mathbf{z}(k) - \mathbf{H}(k)\widehat{\mathbf{x}}(k|k-1)$.
7) $\mathbf{P}^*(k|k) = \frac{\upsilon(k-1)+\Delta^2(k)}{\upsilon(k-1)+d}(\mathbf{P}(k|k-1) - \mathbf{P}(k|k-1)$ $\mathbf{H}^{\mathrm{T}}(k)(\mathbf{S}(k))^{-1}\mathbf{H}(k)\mathbf{P}(k|k-1))$.
8) $\mathbf{P}(k|k) = \frac{\upsilon^*(k)}{\upsilon^*(k)-2}\frac{\upsilon(k)-2}{\upsilon(k)}\mathbf{P}^*(k|k)$.

**Outputs:** $\hat{\mathbf{x}}(k|k)$, $\mathbf{P}(k|k)$.

The presumed state-space model that accurately represents the real system significantly impacts how well state estimation techniques. However, model uncertainties appear when the system is complicated and challenging to model precisely [29]. If measurement noise statistics are unknown, uncertainties may also exist from the method and their statistics [14]. The rapid shift in the system dynamics brought

by, for example, target movement is another typical source of model uncertainty [19]. The presence of model uncertainty may severely hamper the state estimation performance.

In contrast, several candidate models are used in multiple-model techniques to address model uncertainty [20]. IMM produces the state estimate, combining the filtering results. In this work, IMM has been combined with multiple MCStFs by using different models. These models will be explained in the next section.

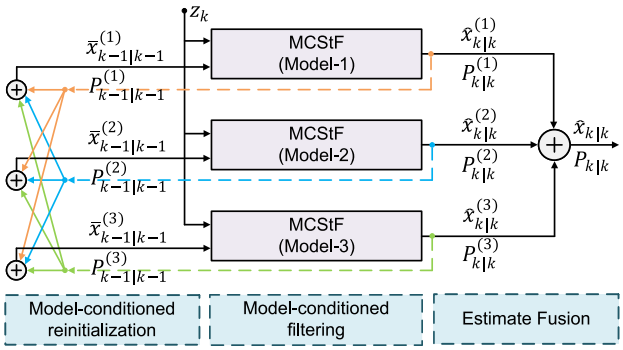


Fig. 2. Architecture of IMM-MCStF.

The architecture diagram of the IMM-MCStF is shown in Fig. 2. The IMM-MCStF structure consists of three MCStFs that operate with three different motion models: the nearly constant velocity and coordinated turn models [19], [31], which are explained in the next section. Each motion model has been specified for each MCStF. The benefits of the IMM have been widely discussed in the literature [19], [30].

## IV. Performance Evaluation and Validations

This section explains how to perform numerical simulation and real-time experiments of IMM-based KF, MCKF and MCStF methods. All experiments are started on the simulation environment, and non-Gaussian noises are implemented on the system model. Then, the state estimation results are evaluated and compared using Root Mean Square Error (RMSE) as a performance measure. Afterwards, we present the real-time experimental results to show the performance of the proposed IMM-MCStF against the conventional IMM-KF and IMM-MCKF. Note that the height was set and fixed as 40 cm in all the conducted experimental studies.

### *A. Numerical Simulation*

The simulations have been performed on Crazyflie 2.0 UAV model. The detailed model has been explained in many research [1], [6], [7]. For the controller, interval type-2 fuzzy PID has been chosen, and the linear velocities ($v_x$ and $v_x$) of the UAV have been chosen as reference signals. Moreover, the altitude level has been set at 40 cm. The model contains Crazyflie 2.0's positions in the 2-D plane and is used as the reference benchmark data set to test the performances of the implemented filters; IMM-KF, IMM-MCKF, and IMM-MCStF. During the simulation, it is assumed that the measurements are corrupted with a Student's T distribution noise which can be defined as one of the non-Gaussian distribution types. Thus, the reference benchmark data set has been corrupted with a noise that has a Student's T distribution. In other words, the employed IMM methods are tested in an environment with noise with Non-Gaussian distribution.

In [30], the effects of DoF have been explained and shown in Fig. 1. In this study, the distribution has been chosen between $[3-25]$, which means the noise is variable on intermediate and extreme levels heavy-tailed.

A constant velocity model and coordinated turn models with state transition matrices respectively ($\mathbf{F}_{CVM}$) and ($\mathbf{F}_{CTM}$) represent the different types of UAV motion. Two coordinated turn models have been implemented, representing two different directions (left turn and right turn motions). To determine the non-Gaussian process model, ($\mathbf{F}_{CTM}$). var(x) $= \alpha V$ has been merged with the defined covariance matrices.

The system state transition matrices of the constant velocity model and of the coordinated turn models have the form:

$$\mathbf{F}_{CVM} = \begin{bmatrix} 1 & \Delta\mathrm{T} & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \Delta\mathrm{T} \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{18}$$

$$\mathbf{F}_{CTM} = \begin{bmatrix} 1 & sin(\omega_i\Delta\mathrm{T}) & 0 & -(1-cos(\omega_i\Delta\mathrm{T})/\omega_i \\ 0 & cos(\omega_i\Delta\mathrm{T}) & 0 & -sin(\omega_i\Delta\mathrm{T}) \\ 0 & (1-cos(\omega_i\Delta\mathrm{T}))/\omega_i & 1 & sin(\omega_i\Delta\mathrm{T})/\omega_i \\ 0 & sin(\omega_i\Delta\mathrm{T}) & 0 & cos(\omega_i\Delta\mathrm{T}) \end{bmatrix}, \tag{19}$$

and the respective system noise covariance matrix of the coordinated turn model is:

$$\mathbf{Q} = \alpha \begin{bmatrix} \frac{\Delta\mathrm{T}^4}{4} & \frac{\Delta\mathrm{T}^3}{2} & 0 & 0 \\ \frac{\Delta\mathrm{T}^3}{2} & \Delta\mathrm{T}^2 & 0 & 0 \\ 0 & 0 & \frac{\Delta\mathrm{T}^4}{4} & \frac{\Delta\mathrm{T}^3}{2} \\ 0 & 0 & \frac{\Delta\mathrm{T}^3}{2} & \Delta\mathrm{T}^2 \end{bmatrix}, \tag{20}$$

where the $\alpha$ value on the model side has been constant and chosen as "5", but the $\upsilon$ value on the measurement side has been changeable between 3 and 25, and step size has been 2 for each Monte-Carlo step. Moreover, 3000 Monte-Carlo independent repetitions have been used for the experiment. After the 3000 times repetition, all processing time has been measured for each IMM structure. IMM-KF, IMM-MCKF and IMM-MCStF have given 766 $sec$, 4724 $sec$ and 5121 $sec$, respectively. Although IMM-KF has been faster than the others, their RMSE results have shown real differences between them. In addition, the process time differences problem can be eliminated in a real-time system.

In Fig. 3, noised reference, which means reference with Non-Gaussian distribution, IMM-KF, IMM-MCKF and IMM-MCStF, has been coloured as black, green, blue, and red, respectively. The reference square reference has tracked the UAV in this figure. It shows that IMM-KF faces out of reference lines. It is expected due to discrepancies in the

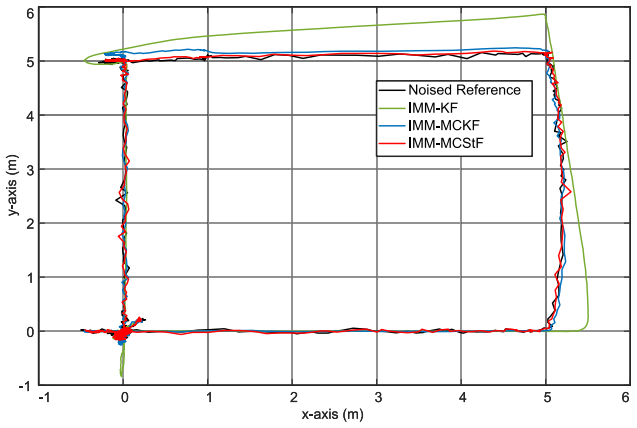


Fig. 3. Reference trajectory tracking for each IMM structure.

covariance matrices. Considering the IMM-KF and IMM-MCKF, process covariance matrices depend on Gaussian distribution. On the other hand, IMM-MCStF has worked under non-Gaussian noises.

The changes in the $x$ and $y$ coordinates (shown as a trajectory in Fig. 3) are presented separately in Fig. 4 and Fig. 5. Unlike the compared IMM algorithms, IMM-MCStF has the best result considering the 3000 times Monte-Carlo running and reference trajectory.

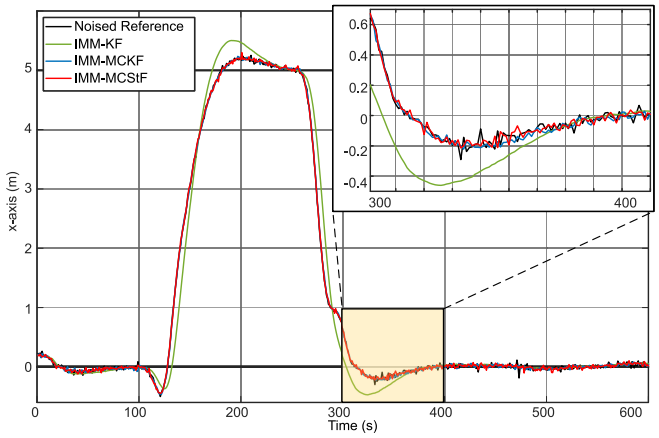


Fig. 4. Results of compared IMM methods on the x-axis.

From the trajectory given in Fig. 4, it is evident that the IMM-KF algorithm shows an overshoot of the estimated trajectory in some parts, while the IMM-MCKF shows steadily converging state estimation error. These overshoots and steady-state errors affect the overall performance results.

Like the x-axis results, the same performance results can be observed on the y-axis; this has been shown in Fig. 5. Moreover, the results for a specific area demonstrate that the IMM-MCStF tracks the reference line perfectly compared with other algorithms.

Both position results and axes velocities have been investigated and graphed. The system velocity responses can be seen in Figs. 6 and 7. The reference velocity has aggressive responses between $100-200$ and $250-350$ iterations.

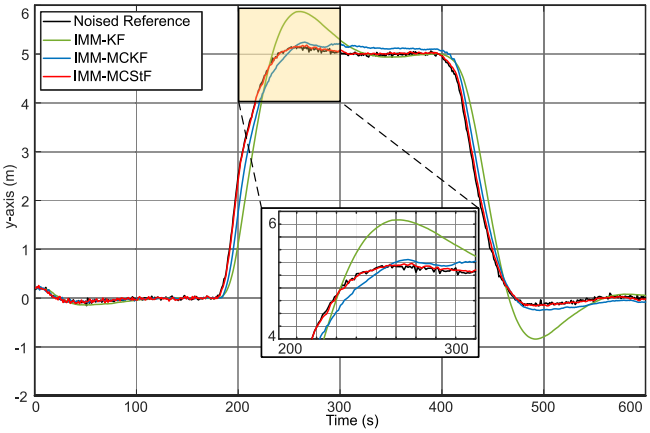


Fig. 5. Results of compared IMM methods on the y-axis.

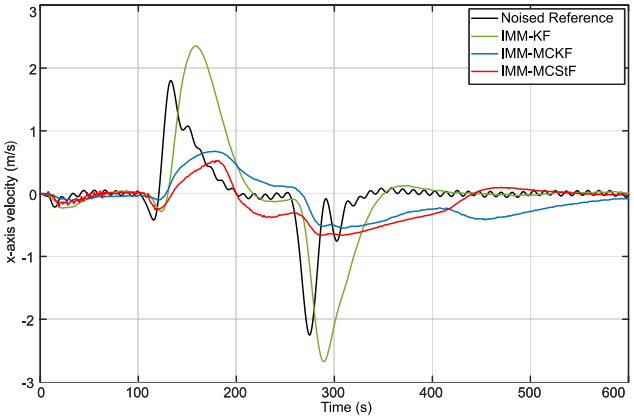


Fig. 6. Results of compared IMM methods x-axis velocities.

At these points, noise or distribution effects can be seen. Although IMM-KF has not had a good result, IMM-MCKF and IMM-MCStF have eliminated the distribution effect more than IMM-KF.

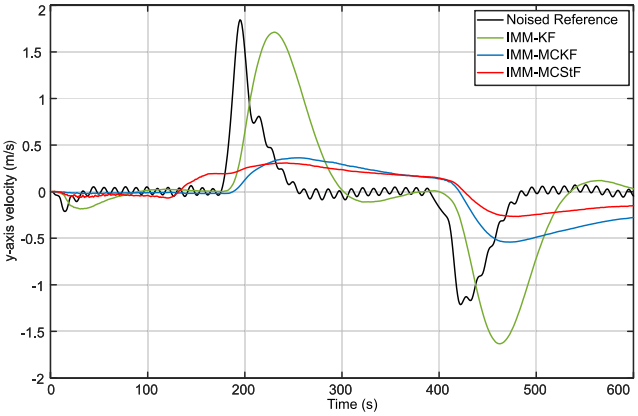


Fig. 7. Results of compared IMM methods y-axis velocities.

Although 3000 independent Monte Carlo repetition has been done, the number of samples $j = 1-100$ results have been shown in Figs. 8 and 9. Each sample shows average RMSE value of the proposed and the compared methods.

With these figures, it can be seen that the errors are repeated after 100 independent iterations.

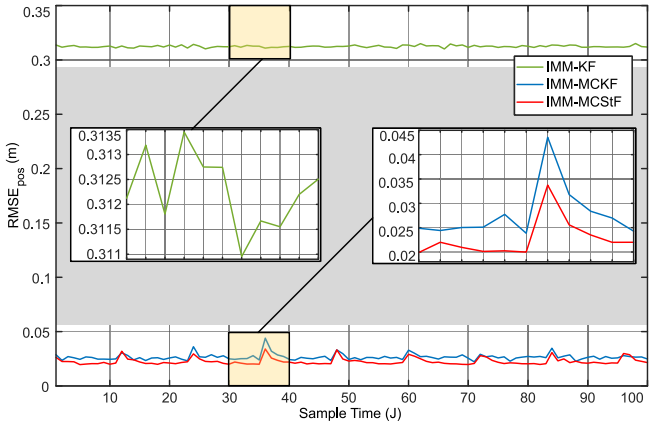


Fig. 8. RMSE Results of compared IMM methods on the x-axis.

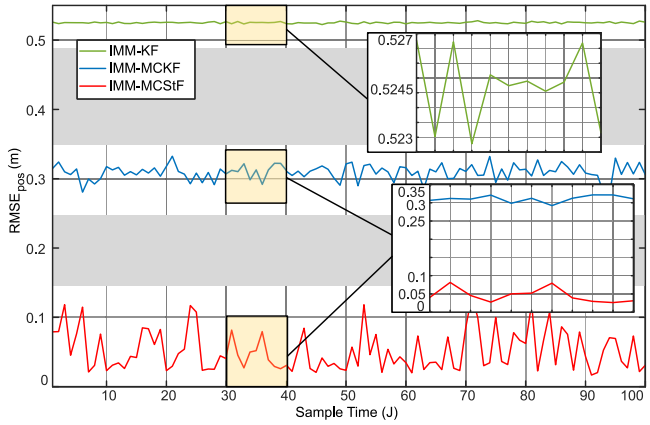


Fig. 9. RMSE Results of compared IMM methods on the y-axis.

With these 100 RMSE values, MCKF and MCStF have almost the same error values on the x-axis. On the contrary, these filters have 10 times better results than IMM-KF. Moreover, in Fig. 9, it can be seen that IMM-MCStF has 6 and 10 times better results on the y-axis than IMM-MCKF and IMM-KF, respectively. The UAV movement dynamics directly affect the axes. The UAV has long distances on the axis, and the heading angle direction is on the y-axis. Therefore, part of the errors can be eliminated by the controller of the UAV. That is why the UAVs have almost the same RMSE values on the x-axis.

TABLE I

RMSE VALUES OF TESTED IMM METHODS.

| RMSE Values | IMM-KF | IMM-MCKF | IMM-MCStF |
|---|---|---|---|
| $x$-axis results (m) | 0.2413 | **0.0221** | 0.0230 |
| $v_x$-results (m/s) | 0.6321 | 0.4566 | **0.4408** |
| $y$-axis results (m) | 0.4108 | 0.1841 | **0.0388** |
| $v_y$-results (m/s) | 0.5687 | 0.4245 | **0.3834** |

Instead of 100 independent iterations, 3000 independent Monte-Carlo RMSE results have been shown in Table I. Unlike RMSE figures, RMSE values on velocities have been shown in the table. The results show that the proposed method has significantly good results when compared to methods except for the results on the y-axis. On the x-axis, the proposed and the compared methods have almost the same response. It has shown the same performance values on both axes in the figures and table of RMSE.

### *B. Real-time Application Results*

In this subsection, the real-time experimental results of the implemented IMM filters are demonstrated on Crazyflie 2.0 commercial mini UAV [5] as seen in Fig. 10. In the simulation studies, the performances are compared through an offline analysis with collected position data. On the other hand, real-time experiments are conducted online in the real-time. In this context, We have constructed an experiment environment as shown in Fig. 11. On top of the experiment environment, an Intel real-sense camera system [15] is located that is used for detecting and tracking the positions of the Crazyflie 2.0 during the experiments in real-time. The frames are collected via the camera system and then processed by a vision-based localisation structure to obtain the positions in 2-D coordinate $(x, y)$.

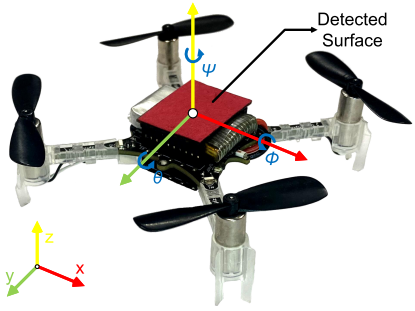


Fig. 10. Crazyflie 2.0 coordinate frame.

The experiment area is constrained by a protection net so that the mini UAV does not fly out, and also we limit the mini UAV's flying area. Intel real-sense camera, which provides accurate position information with low noise, has been used in the experiments. Therefore, we inject other noise sources into the real-world experiment area to decrease camera accuracy, assuming they have non-Gaussian distribution. Testing the performance of IMM-MCStF has been modelled for Non-Gaussian or Gaussian distribution, such as changing environment temperature, different initial levels of the battery, wind sources, and other inaccessible reasons. In [6], [7], the dynamic model of the Crazyflie 2.0 is given afterwards; state estimation, attitude, altitude controllers and position controllers are explained in detail, which are the same control structures used in this.

**Remark 1:** Intel real sense has significant accuracy in getting the position of the desired object. For this reason, non-Gaussian noise with $[3-25]$ DoF has been injected as monotonic increasing and decreasing.

In Fig. 12, Crazyflie 2.0's reference trajectory and the resulting performance trajectories of the implemented IMM

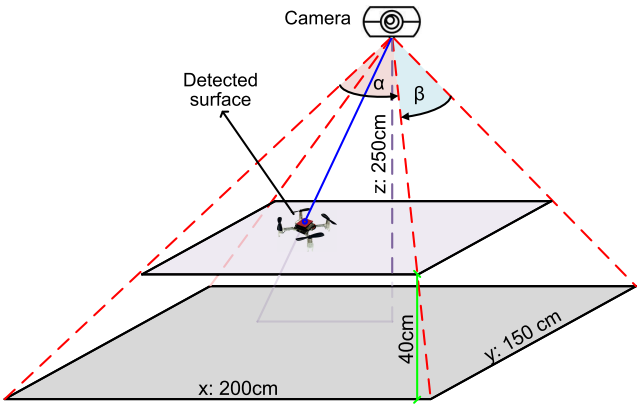


Fig. 11. Global Camera Location.

filters under a noisy environment are presented. Here, point 0 represents the initial point of Crazyflie 2.0 just after take-off, while point 4 represents the final point of Crazyflie 2.0. One needs to remark that Crazyflie 2.0 started from the same point on the ground for all experiments conducted with all implemented IMM filters, and point 0 in Fig. 12 shows the first positions in hovering just after taking.

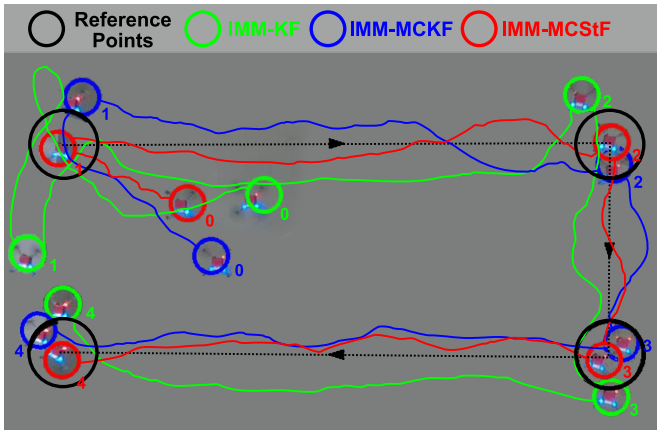


Fig. 12. Crazyflie 2.0 reference trajectory results on the real-time experiments area.

**Remark 2:** It should be noted that initial values of covariance, measurement, and process noise matrices have been set as the same for all the implemented IMM filters, and they selected as suggested in [6].

The real-time performance of the implemented Interacting Multiple Models (IMM) filters for the reference trajectories in the $x$-$y$ plane is illustrated in Fig. 13 and Fig. 14. These figures depict the mean values of the resulting trajectories from 10 independent experiments for each IMM filter, considering the stochastic nature of the environment. The performance measures obtained from these experiments are presented in Table II.

Fig. 13 and Fig. 14 clearly indicate that the IMM-MCStF outperforms its counterparts in both axes. The response of the IMM filters in Fig. 13 shows similar performance, with the IMM-MCStF achieving even better results compared to the

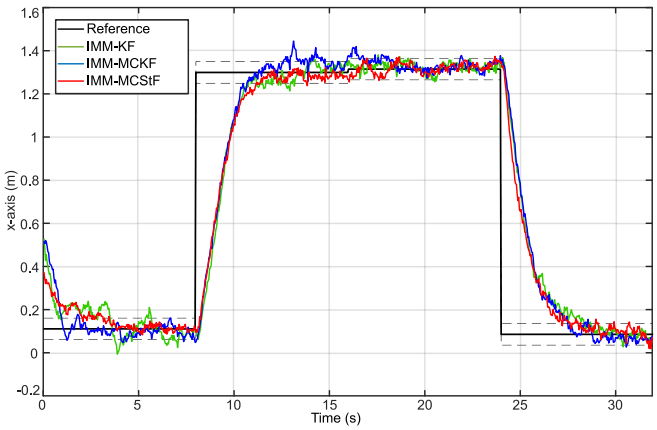


Fig. 13. Crazyflie 2.0 reference trajectory results on the x-axis.

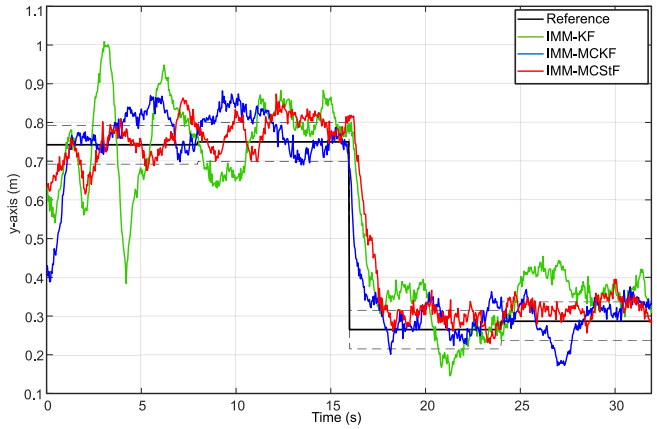


Fig. 14. Crazyflie 2.0 reference trajectory results on the y-axis.

IMM-MCKF. However, Fig. 14 reveals distinct behaviour, with the IMM-MCKF exhibiting oscillations around the reference trajectory within the interval of 0 to 15 seconds, while the IMM-MCStF demonstrates an almost perfect response on the $y$-axis. For instance, at the 9th second, the IMM-MCKF shows a deviation of 9 cm from the reference, whereas the IMM-MCStF maintains a near-perfect response. Additionally, the coupling effect is observed in Fig. 14 at the 24th second when a reference change occurs on the $x$-axis. It can be concluded that the IMM-MCStF exhibits a more robust performance in handling this effect.

TABLE II

REAL-TIME RMSE VALUES OF TESTED IMM METHODS

| RMSE Values | IMM-KF | IMM-MCKF | IMM-MCStF |
|---|---|---|---|
| $x$-axis results (m) | 0.1368 | 0.1306 | **0.1222** |
| $y$-axis results (m) | 0.0929 | 0.0560 | **0.0342** |

From Table II, it can be firstly commented that the IMM-MCStF outperformed the IMM-KF and IMM-MCKF significantly as it improved the RMSE value by an amount of $40\%$ for the y-axis when compared to its IMM-MCKF counterpart. On the other hand, the RMSE value is by an amount of $10\%$ for the x-axis compared to its IMM-KF

counterpart.

## V. Conclusion and Future Works

A novel estimation method called Interacting Multiple Model Maximum Correntropy Student's T Filter (MCStF), has been proposed in this study. The proposed method has been compared with the IMM-based Kalman filter (KF) and maximum correntropy Kalman filter (MCKF). For the comparison, non-Gaussian-based distributions have been applied to the reference trajectory. Moreover, the selected non-Gaussian distribution, Student's t based, had variable degrees of freedom. It changed between 3 and 25. Therefore, the noises are between extreme and intermediate level heavy-tailed. In addition, the simulation and real-time experiment environments have been applied to analyse the MCStF response. Both environments gave the same error results. That means simulation and real-time results proved each other.

Future work will focus on implementing the MCStF on a swarm of UAVs whose system covariance matrices are unknown. Also, MCStF will be combined with local path planning methods, such as the velocity obstacle and reciprocal velocity methods. After these implementations, the MCStF will be investigated for collision avoidance.

## References


[1] S. Abdelmoeti and R. Carloni, "Robust control of uavs using the parameter space approach," in *2016 IEEE/RSJ International Conf. on Intel. Robots and Systems (IROS)*. IEEE, 2016, pp. 5632–5637.

[2] G. Agamennoni, J. I. Nieto, and E. M. Nebot, "Approximate inference in state-space models with heavy-tailed noise," *IEEE Trans. on Signal Process.*, vol. 60, no. 10, pp. 5024–5037, 2012.

[3] B. Alzahrani, O. S. Oubbati, A. Barnawi, M. Atiquzzaman, and D. Alghazzawi, "UAV assistance paradigm: State-of-the-art in applications and challenges," *Journal of Network and Computer Tech.*, vol. 166, p. 102706, 2020.

[4] F. Auger, M. Hilairet, J. M. Guerrero, E. Monmasson, T. Orlowska-Kowalska, and S. Katsura, "Industrial apls. of the Kalman filter: A review," *IEEE Trans. on Industrial Electronics*, vol. 60, no. 12, pp. 5458–5471, 2013.

[5] C. Bitcraze Crazyflie, "Bitcraze Crazyflie 2.0," https://www.bitcraze.io/products/old-products/crazyflie-2-0, accessed: 2022-06-16.

[6] F. Candan, A. Beke, C. Shen, and L. Mihaylova, "An interacting multiple model correntropy Kalman filter approach for unmanned aerial vehicle localisation," in *Proc. of the Internat. Conf. on INnovations in Intel. Sys. and Apls. (INISTA)*. IEEE, 2022, pp. 1–6.

[7] F. Candan, O. F. Dik, T. Kumbasar, M. Mahfouf, and L. Mihaylova, "Real-time interval type-2 fuzzy control of an unmanned aerial vehicle with flexible cable-connected payload," *Algorithms*, vol. 16, no. 6, p. 273, 2023.

[8] L. Cao, D. Qiao, and X. Chen, "Laplace $L1$ Huber based cubature Kalman filter for attitude estimation of small satellite," *Acta Astronautica*, vol. 148, pp. 48–56, 2018.

[9] B. Chen, X. Liu, H. Zhao, and J. C. Principe, "Maximum correntropy Kalman filter," *Automatica*, vol. 76, pp. 70–77, 2017.

[10] S. Y. Chen, "Kalman filter for robot vision: A survey," *IEEE Trans. on Industrial Electronics*, vol. 59, no. 11, pp. 4409–4420, 2012.

[11] A. S. Elkhatem and S. N. Engin, "Robust LQR and LQR-PI control strategies based on adaptive weighting matrix selection for a UAV position and attitude tracking control," *Alexandria Engineering Journal*, vol. 61, no. 8, pp. 6275–6292, 2022.

[12] B. Fan, Y. Li, R. Zhang, and Q. Fu, "Review on the technological development and application of UAV sys." *Chinese Journal of Electronics*, vol. 29, no. 2, pp. 199–207, 2020.

[13] X. Fan, G. Wang, J. Han, and Y. Wang, "Interacting multiple model based on maximum correntropy Kalman filter," *IEEE Trans. on Circuits and Sys. II: Express Briefs*, vol. 68, no. 8, pp. 3017–3021, 2021.

[14] Y. Huang, Y. Zhang, N. Li, Z. Wu, and J. A. Chambers, "A novel robust Student's t-based Kalman filter," *IEEE Trans. on Aerospace and Electronic Sys.*, vol. 53, no. 3, pp. 1545–1554, 2017.

[15] C. Intel Realsense, "Intel Realsense D415," https://www.intelrealsense.com/depth-camera-d415, accessed: 2022-06-16.

[16] M. Jacquet, M. Kivits, H. Das, and A. Franchi, "Motor-level N-MPC for cooperative active perception with multiple heterogeneous UAVs," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 2063–2070, 2022.

[17] C. D. Karlgaard, "Nonlinear regression Huber–Kalman filtering and fixed-interval smoothing," *Journal of Guidance, Control, and Dynamics*, vol. 38, no. 2, pp. 322–330, 2015.

[18] S. Li, L. Li, D. Shi, W. Zou, P. Duan, and L. Shi, "Multi-kernel maximum correntropy Kalman filter for orientation estimation," *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 6693–6700, 2022.

[19] X. R. Li and V. P. Jilkov, "Survey of maneuvering target tracking. Part I. Dynamic models," *IEEE Trans. on Aerospace and Electronic Sys.*, vol. 39, no. 4, pp. 1333–1364, 2003.

[20] ——, "Survey of maneuvering target tracking. Part V. Multiple-model methods," *IEEE Trans. on Aerospace and Electronic Sys.*, vol. 41, no. 4, pp. 1255–1321, 2005.

[21] X. Liu, X. Liu, W. Zhang, and Y. Yang, "Interacting multiple model uav navigation algorithm based on a robust cubature Kalman filter," *IEEE Access*, vol. 8, pp. 81 034–81 044, 2020.

[22] J. P. Matos-Carvalho, R. Santos, S. Tomic, and M. Beko, "GTRS-Based algorithm for UAV navigation in indoor environments employing range measurements and odometry," *IEEE Access*, vol. 9, pp. 89 120–89 132, 2021.

[23] J. A. Paredes, F. J. Álvarez, M. Hansard, and K. Z. Rajab, "A Gaussian process model for UAV localization using millimetre wave radar," *Expert Sys. with Apls.*, vol. 185, p. 115563, 2021.

[24] Y. Pei, S. Biswas, D. S. Fussell, and K. Pingali, "An elementary introduction to Kalman filtering," *Comms. of the ACM*, vol. 62, no. 11, pp. 122–133, 2019.

[25] R. Piché, S. Särkkä, and J. Hartikainen, "Recursive outlier-robust filtering and smoothing for nonlinear systems using the multivariate Student-t distribution," in *Proc. of the IEEE Internat. Workshop on Machine Learning for Signal Process.* IEEE, 2012, pp. 1–6.

[26] K. Qiu, T. Liu, and S. Shen, "Model-based global localization for aerial robots using edge alignment," *IEEE Robotics and Automation Letters*, vol. 2, no. 3, pp. 1256–1263, 2017.

[27] M. M. Rana, N. Halim, M. M. Rahamna, and A. Abdelhadi, "Position and velocity estimations of 2D-moving object using Kalman filter: Literature review," in *Proc. of the 22nd Internat. Conf. on Advanced Comm. Tech. (ICACT)*. IEEE, 2020, pp. 541–544.

[28] M. Roth, E. Özkan, and F. Gustafsson, "A Student's t filter for heavy tailed process and measurement noise," in *Proc. of the IEEE Internat. Conf. on Acoustics, Speech and Signal Process.* IEEE, 2013, pp. 5770–5774.

[29] A. H. Sayed, "A framework for state-space estimation with uncertain models," *IEEE Trans. on Automatic Control*, vol. 46, no. 7, pp. 998–1013, 2001.

[30] C. Shen and L. Mihaylova, "A flexible robust Student's t-based multimodel approach with maximum versoria criterion," *Signal Process.*, vol. 182, p. 107941, 2021.

[31] Z. Sutton, P. Willett, and Y. Bar-Shalom, "Target tracking applied to extraction of multiple evolving threats from a stream of surveillance data," *IEEE Trans. on Computational Social Systems*, vol. 8, no. 2, pp. 434–450, 2021.

[32] D. C. Tsouros, S. Bibi, and P. G. Sarigiannidis, "A review on UAV-based applications for precision agriculture," *Information*, vol. 10, no. 11, p. 349, 2019.

[33] H. D. Tuan, A. A. Nasir, A. V. Savkin, H. V. Poor, and E. Dutkiewicz, "MPC-based UAV navigation for simultaneous solar-energy harvesting and two-way communications," *IEEE Journal on Selected Areas in Comms.*, vol. 39, no. 11, pp. 3459–3474, 2021.

[34] J. Velagić, N. Osmić, V. Klovo, and H. Lačević, "Design of LQR controller for 3D trajectory tracking of octocopter unmanned aerial vehicle," in *Proc. of the 8th Internat. Conf. on Control, Decision and Inf. Tech. (CoDIT)*, vol. 1. IEEE, 2022, pp. 63–68.

[35] W. You, F. Li, L. Liao, and M. Huang, "Data fusion of UWB and IMU based on unscented Kalman filter for indoor localization of quadrotor UAV," *IEEE Access*, vol. 8, pp. 64 971–64 981, 2020.