

## RESEARCH ARTICLE

# Design of Dynamics Invariant LSTM for Touch Based Human–UAV Interaction Detection

ANEES PERINGAL<sup>1</sup>, MOHAMAD CHEHADEH<sup>1</sup>, (Member, IEEE), RANA AZZAM<sup>1</sup>,  
MAHMOUD HAMANDI<sup>1</sup>, IGOR BOIKO<sup>2</sup>, (Senior Member, IEEE),  
AND YAHYA ZWEIRI<sup>1</sup>, (Member, IEEE)

<sup>1</sup>Khalifa University Center for Autonomous Robotic Systems (KUCARS), Department of Aerospace Engineering, Khalifa University, Abu Dhabi, United Arab Emirates

<sup>2</sup>Department of Electrical Engineering and Computer Science, Khalifa University, Abu Dhabi, United Arab Emirates

Corresponding author: Mahmoud Hamandi (mahmoud.hamandi@ku.ac.ae)

This work was supported by the Khalifa University of Science and Technology under Award CIRA-2020-082 and Award RCI-2018-KUCARS.

**ABSTRACT** The field of Unmanned Aerial Vehicles (UAVs) has reached a high level of maturity in the last few years. Hence, bringing such platforms from closed labs, to day-to-day interactions with humans is important for commercialization of UAVs. One particular human-UAV scenario of interest for this paper is the payload handover scheme, where a UAV hands over a payload to a human upon their request. In this scope, this paper presents a novel real-time human-UAV interaction detection approach, where Long short-term memory (LSTM) based neural network is developed to detect state profiles resulting from human interaction dynamics. A novel data pre-processing technique is presented; this technique leverages estimated process parameters of training and testing UAVs to build dynamics invariant testing data. The proposed detection algorithm is lightweight and thus can be deployed in real-time using off the shelf UAV platforms; in addition, it depends solely on inertial and position measurements present on any classical UAV platform. The proposed approach is demonstrated on a payload handover task between multirotor UAVs and humans. Training and testing data were collected using real-time experiments. The detection approach has achieved an accuracy of 96%, giving no false positives even in the presence of external wind disturbances, and when deployed and tested on two different UAVs.

**INDEX TERMS** Deep learning, LSTM, physical human–robot interaction, self-calibration.

## I. INTRODUCTION

Over the past few years, multirotor unmanned aerial vehicles (UAVs) have been used in abundance in a wide range of civilian and military applications, particularly to perform tasks that are burdensome for humans, such as exploration, aerial photography, and search and rescue [1]. Recent research efforts, on the other hand, have been directed towards developing robots that can work closely with humans, in a variety of fields like manufacturing, healthcare, and entertainment [2], [3]. Very recently, new applications emerged focusing on a direct interaction between a human and UAV. For example, the authors in [4] present a system where a

human is physically connected to a UAV, and where the UAV physically pulls the human towards a desired position, unknown to the human. Similarly, the authors in [5] present a framework to safely operate an aerial manipulator interacting with a human. [5] is part of the flying coworker project,<sup>1</sup> aiming to achieve a safe and reliable human-UAV collaboration. To that end, there is an imperative need for reliable human-robot interaction algorithms in pursuit of the successful deployment of robots in close proximity to humans. Robots need to acquire a set of skills, by means of classical or learning approaches, that facilitate interactions with humans without jeopardizing their safety in their common workplace. For instance, object handover is a very critical skill that robots

The associate editor coordinating the review of this manuscript and approving it for publication was Tao Liu<sup>1</sup>.

<sup>1</sup><https://anr.fr/Project-ANR-18-CE33-0001>

need to get hold of to effectively carry out collaborative tasks with humans [6], [7], [8]. Although this skill has been widely researched for robotic arms, major efforts are yet required to further develop and advance handover algorithms for aerial mobile robots, the mobility of which makes them ideal for such applications. Nevertheless, UAVs suffer from limitations in size, weight, and power, which raise the need for lightweight algorithms that require no additional sensors onboard the vehicle.

For humans to effectively perform collaborative tasks with UAVs, it is crucial to establish a channel of communication between them. From the human's side, observing the behaviour, speed and orientation of the UAV might be sufficient if the UAV is trained to convey its actions adequately to the user [9], [10]. On the other hand, UAVs need to be explicitly trained to detect cues specific to the humans' intention to interact. Upon physically interacting with the UAV, humans exert a certain force, which we conjecture to exhibit unique profiles. Such profiles, if accurately detected and identified, may serve as indicators of how the UAV should move forward with its ongoing task. Therefore, this paper presents a novel learning-based interaction states profile (ISP) detection technique to identify human interactions on UAVs. The approach is demonstrated in an object handover scenario, yet is applicable to any other human-UAV interaction application.

A long-short-term memory (LSTM) network is developed and trained to detect arbitrary dynamics and interaction profiles perceived by the UAV upon physical contact with a human. The proposed approach is designed to be platform-agnostic while relying solely on measurements obtained from the proprioceptive sensors onboard the UAV, i.e. IMU and position sensors. After training and deploying on a physical UAV, the proposed ISP detection algorithm resulted in a successful demonstration of a payload handover to humans through physical interaction.

Experimental results have proven the reliability of the proposed ISP detection approach across a wide range of varying profile complexities, where a success rate of 96% was achieved. The approach was also shown to be robust against random interactions, external wind, and changes in UAV dynamics. In addition, the proposed approach was demonstrated to successfully work on different platforms. Particularly, the ISP detector was trained on a quadrotor and was tested on a vastly different hexarotor. This is attributed to the fact that input to the proposed algorithm is preprocessed to take into account the UAV dynamics and hence resulting in better generalization across different platforms.

## A. RELATED WORK

Detecting an interaction between humans is trivial because a human can obtain information about an interaction both proactively, through vision and audition, and reactively through tactile interactions. Humans also utilize prior experience through their cognitive capabilities to achieve fluid interactions by adequately anticipating and reacting to events and stimuli [11]. When attempting to involve a robotic agent

for physical human-robot interaction, a new set of challenges needs to be addressed, due to the inherent limitations in onboard sensors, the unreliable knowledge of the interaction model, and the difficulty to predict humans' actions [12] and movements [13]. Relevant work in the literature can be broadly categorized based on the platform used in the study. Most work on object handover and physical human interaction is done on a robotic arm. In this paper, however, the focus will be devoted to object handover by a UAV.

### 1) ROBOTIC ARM

Object handover and interaction detection with robotic arms can be classified into proactive and reactive methods. Proactive interaction detection methods get information about the receiver before the actual interaction occurs. These methods make use of vision sensors or motion capture of the human. For example, in [14] a glove fitted with tactile sensor patches, motion capture markers, and force/torque sensors is worn by the user during the handover. The glove is used for position and force estimation of the human during the handover. The glove is used for experiments in human-human handover to obtain insights into trajectories and forces humans use to hand objects over. Using these insights, a robot-to-human handover algorithm is proposed aimed at fluid handovers. This algorithm is also able to reduce the internal forces that act on the object to protect it from damage during handover. Since this method requires the human to be wearing a specialized precalibrated sensor, it is difficult to generalize to different platforms.

Reactive interaction detection methods make use of sensor data after the interaction has begun. This may include sensors such as force/torque sensors, tactile sensors, etc. In [15] and [16], tactile sensors are used to find when the receiver is ready to accept the payload. With the help of the above mentioned sensors, the robot was able to reject external perturbations and release the object only when the human interacts in a particular direction. This approach, however, has the drawback that it requires specialized sensors. In some reactive methods, there is no need for specialized sensors. If good knowledge of the robotic arm is available, then the external force acting on the object can be estimated. Such a method was adopted in [17], where an extended Kalman filter is used to estimate the external forces acting on the robotic arm and implement a force controller for various tasks. Note that in this method, the robot parameters are considered to be known prior to the experiment. On the other hand, the method in [18] employs a recurrent neural network (RNN) trained to identify the dynamics of a two-link robotic arm, where the network estimates the torques generated by the arm from the knowledge of the joint's angles and their derivatives. The RNN is validated against the measurements from a torque sensor.

### 2) UAV PLATFORM

Interaction with UAVs introduces new challenges that need to be accounted for as compared to a robotic arm; (1) a UAV

platform is inherently unstable, hence the stability of the UAV during interaction has to be taken into account. (2) the size, weight and power of a UAV are limited as compared to fixed manipulators.

In [19], an onboard force/torque six-degree sensor is used to measure external forces acting on a UAV. A passivity based controller is then used to guarantee the stability of the platform following an external interaction. This approach was demonstrated on a tethered multirotor UAV which does not require an onboard battery. In [20], an array of force sensors is used along with a disturbance observer to detect human interaction on a UAV platform. The measured force and direction are then used to design an admittance controller. Additional onboard sensors on a UAV will increase the weight of the UAV and thus decrease its performance. Due to the additional cost and weight added by the above sensors, in this work, we will focus solely on sensors available on off-the-shelf UAVs.

In [21], a sensorless force estimation is done using an unscented Kalman filter with a quaternion-based controller for a UAV to estimate the force and torque acting on the UAV. The authors show that this method can estimate the force and torque profiles in real-time. However, it cannot differentiate between types of forces acting on the UAV. These force estimation methods also require good prior knowledge of the UAV being used which hinders the transferability to different platforms. In [22], external wrench applied on a UAV platform is estimated based on the proprioceptive sensors onboard the UAV. The estimated external wrench is used to design an admittance and impedance controller for interactions.

The majority of the surveyed research focuses on robotic arms used for payload handover. Interactions with UAVs mandate considering some aspects that are not critical for robotic arms, such as weight. For example, using additional sensors onboard the UAV is not preferable, given its payload constraints. On the other hand, there exist force estimation methods that do not require additional sensors, but they mostly rely on the knowledge of the UAV model, and hence, can only be used for particular platforms.

To the best of the authors' knowledge, none of the methods proposed in the literature makes use of the profile of states on a UAV for the purpose of differentiating between interactions, which can then be used to encode the communication between a human and a UAV during a collaborative task.

## B. CONTRIBUTION

Based on the surveyed literature, we propose a LSTM-based learning approach for detecting and identifying the profile of the states during a human interaction on a UAV. The proposed approach has a major advantage over existing methods in that the training data and inference of the LSTM are independent of the platform dynamics. Hence, a single model can be used for ISP detection on multiple UAV platforms, and model training can be done using data collected from any UAV

platform. In summary, the contributions of this paper are as follows:

- A novel LSTM-based interaction states profile (ISP) detection approach is proposed to discern human interactions with UAVs, through proprioceptive measurements like IMU and position sensor measurements.
- A novel data pre-processing technique is developed to make the proposed ISP detection approach invariant to UAV dynamics and hence achieve generality across various platforms without fine-tuning of the trained model. Specifically, the estimated ISP is transformed to a new domain, which we refer to as the training and inference domain (TID), before being processed by the ISP detection approach. The transformation is based on the UAV dynamic parameters identified using the DNN-MRFT approach proposed in [23].
- The validity of the proposed ISP detection approach, and its applicability to different UAVs is demonstrated experimentally for different human-UAV interaction scenarios. The detection accuracy on the experimental test set was more than 96% with no false positives even in the presence of challenging wind conditions, and unwanted random pushes from nearby users. A video summary of the experimental results can be found at [24] (<https://youtu.be/WcXa1H558UA>).

It is worth noting that the proposed dynamics invariant interaction detection approach is not specific to LSTM. Other time-series detection algorithms could be used in the TID so that dynamics independence properties are retained. The motivation behind the selection of LSTM for the Human-UAV interaction application is discussed in Section IV.

## C. PAPER STRUCTURE

This paper is structured as follows. Section II summarizes the nonlinear modelling of UAV dynamics, the linear decoupled approximations, and the identification of the UAV model parameters. In Section III, the pre-processing of the data by a transformation to the TID based on the identified UAV model parameters is presented. In Section IV, the design and training of the LSTM-based neural network, which operates in the TID is presented. The proposed ISP detection is experimentally verified and an application of this method as payload handover between a robot and a human is demonstrated in real experiments in Section V. Finally, Section VI concludes the paper.

## II. MODELING, CONTROL AND IDENTIFICATION

In this section, the nonlinear model of the quadrotor will be presented, considering the propulsion dynamics and time delays. Then, a linear model that can be used for system identification will be described. This modelling method can be easily extended to any multirotor UAV [25], however, in what follows it is derived for a coplanar/collinear platform since a classical quadrotor and a classical hexarotor will be used to demonstrate and verify the proposed approach.

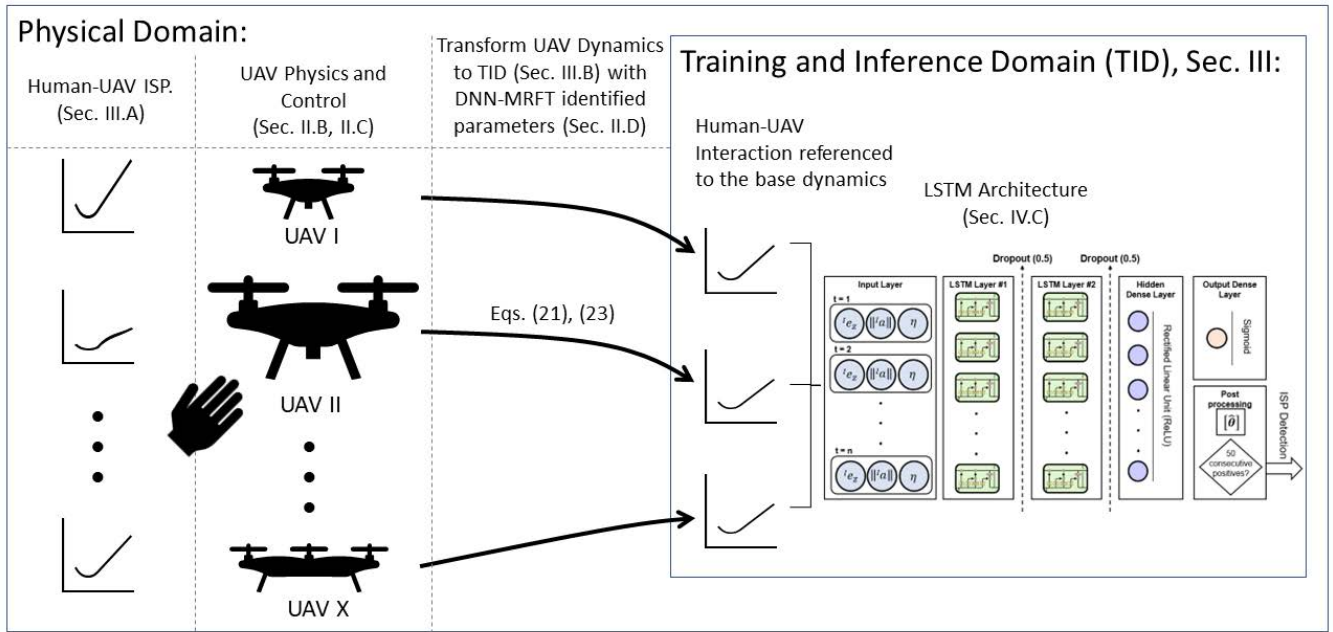


FIGURE 1. Schematic showing the different constituents of the proposed detection approach.

### A. REFERENCE FRAMES AND COORDINATE SYSTEM

First, two right-handed reference frames are defined; (1) an Inertial Earth-fixed frame  $\mathcal{F}_I$  with axes  $\{\hat{i}_x, \hat{i}_y, \hat{i}_z\}$ , and (2) a body-fixed frame  $\mathcal{F}_B$  with axes  $\{\hat{b}_x, \hat{b}_y, \hat{b}_z\}$ . The axis  $\hat{i}_z$  is chosen to point upwards opposite to gravity, and  $\hat{b}_z$  parallel to the direction in which actuators exert thrust. The center of the body-fixed frame is at the center of gravity of the UAV platform and the rotations around the inertial axes are given by Euler angles  $\eta = [\phi \ \theta \ \psi]$  which represent roll, pitch and yaw. For convenience, in what follows a vector may be expressed in a particular reference frame by indexing its components using the frame's symbol. For example, the position vector  ${}^I\mathbf{p}$  is defined with reference to the inertial frame and has the components  $[{}^I p_x \ {}^I p_y \ {}^I p_z]^T$ . In addition, we use the notion of  $\bar{v}$  to represent a 2D vector obtained from the projection of a 3D vector  $v$  onto the  $x$ - $y$  defined plane.

### B. NON-LINEAR MULTIROTOR MODEL

Based on the frames of reference defined above, the non-linear model of the UAV will be derived based on the Newton-Euler equations. First let us denote by  $n \in \mathbb{N}$  as the total number of propellers, where  $n = 4$  for a quadrotor, and  $n = 6$  for a hexarotor. The total thrust force  $f_T$ , pitching and rolling moments  $\tau_{\phi, \theta}$ , and yawing moments  $\tau_\psi$  generated by the corresponding propellers, and expressed in  $\mathcal{F}_B$  are related to the rotation speed  $\Omega$  of each propeller and are described by:

$$f_T = \sum_{i=1}^n f_i = \sum_{i=1}^n k_f \Omega_i^2 \quad (1)$$

$$\tau_{\phi, \theta} = \sum_{i=1}^n k_f l_{\phi, \theta} \Omega_i^2 \begin{bmatrix} \sin(\gamma_i) \\ \cos(\gamma_i) \end{bmatrix} \quad (2)$$

$$\tau_\psi = \sum_{i=1}^n (-1)^{i+1} k_\tau \Omega_i^2 \quad (3)$$

where  $f_i$  is the thrust produced by the corresponding propeller about its  $z$ -axis,  $k_f$  and  $k_\tau$  are the thrust and drag coefficient of each propeller,  $l_{\phi, \theta}$  is the moment arm connecting the platform's center of mass to the each propeller's center of mass, and  $\gamma_i$  is the angle between  $\hat{b}_x$  and the arm connecting the platform to the corresponding propeller.

The UAV is assumed to be symmetric about  $\hat{b}_x$  and  $\hat{b}_y$ , with its center of gravity coinciding with the origin of  $\mathcal{F}_B$ . With these assumptions, the product of inertia of the UAV is zero, and therefore, the inertia matrix is  $\mathbf{J} = \text{diag}(J_x, J_y, J_z)$ . As such, the Newton-Euler equations for the multirotor UAV are given by:

$$\begin{bmatrix} m\mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{J} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{V}} \\ \dot{\boldsymbol{\omega}} \end{bmatrix} = \begin{bmatrix} {}^B \mathbf{F} \\ {}^B \boldsymbol{\tau} \end{bmatrix} \quad (4)$$

$${}^B \mathbf{F} = f_T \hat{b}_z - {}^B \mathbf{R} m \mathbf{g} \hat{i}_z - \boldsymbol{\alpha} \quad (5)$$

$${}^B \boldsymbol{\tau} = \begin{bmatrix} \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} - \boldsymbol{\lambda} \quad (6)$$

where  $m$  is the mass of the UAV,  $\dot{\mathbf{V}}$  is the acceleration of the UAV, and  $\dot{\boldsymbol{\omega}}$  is the angular acceleration of the UAV.  $\boldsymbol{\alpha}$ , and  $\boldsymbol{\lambda}$  are arbitrary constants that depend on the translational and rotational drag on the UAV, and captures motion inflow and blade flapping drags on the propellers. If the cross-coupling, observed in (4), (5) and (6) is neglected, the model can be

simplified as follows.

$$\begin{aligned}
{}^l\ddot{p}_x &= \frac{1}{m} \left( (c_\psi s_\theta c_\phi + s_\psi s_\phi) f_T - \alpha_x ({}^l\dot{p}_x, \eta, \Omega) \right), \\
{}^l\ddot{p}_y &= \frac{1}{m} \left( (s_\psi s_\theta c_\phi - c_\psi s_\phi) f_T - \alpha_y ({}^l\dot{p}_y, \eta, \Omega) \right), \\
{}^l\ddot{p}_z &= \frac{1}{m} \left( c_\phi c_\theta f_T - g - \alpha_z ({}^l\dot{p}_z, \eta, \Omega) \right), \\
\ddot{\phi} &= \frac{1}{J_x} (\tau_\phi - \lambda_\phi(\omega, \Omega)), \\
\ddot{\theta} &= \frac{1}{J_y} (\tau_\theta - \lambda_\theta(\omega, \Omega)), \\
\ddot{\psi} &= \frac{1}{J_z} (\tau_\psi - \lambda_\psi(\omega, \Omega)).
\end{aligned} \tag{7}$$

A brushless DC motor (BLDC) is used for propulsion, controlled with an electronic speed controller (ESC). Each ESC receives the desired propeller rotational speed  $\Omega_i$ , and controls the BLDC to achieve the desired command  $u_i$ , assumed to be proportional to  $\Omega_i^2$ . The relationship between the desired command applied by the ESC and the produced thrust or torque by the corresponding BLDC is approximated by a first-order plus time delay model [26], and is given by:

$$\dot{f}_i(t)T_p + f_i(t) = K_p u_i(t - \tau_p) \tag{8}$$

where  $T_p$  is the propulsion time constant,  $K_p$  is the propulsion dynamics gain,  $\tau_p$  is the propulsion dynamics delay and  $t$  is the time variable. From bench tests on the propulsion system, we can assume  $T_p$  is constant in the operating range.

### C. LINEARIZED INNER LOOP DYNAMICS

The inner loop dynamics of a multirotor UAV consists of the altitude dynamics, the attitude (i.e. roll and pitch) dynamics, and the yaw dynamics. Both altitude and attitude dynamics of a multirotor UAV have the same model structure, yet the model parameters differ for each. For example, the model for the pitch loop is given by:

$$T_1 \ddot{\theta}(t) + \dot{\theta}(t) = K \tau_u \tag{9}$$

where  $T_1$  represents drag time constant, and  $K$  represents the loop gain. The pitch state variable  $\theta$  might be simply substituted by the state variables  $\phi$ , and  ${}^l p_z$  to obtain the other models. Yet, it is required to substitute the input torque  $\tau_u$  in (9) with the propulsion dynamics presented in (8). The full pitch dynamics are given by

$$T_1 T_p \ddot{\theta}(t) + (T_1 + T_p) \dot{\theta}(t) + \theta(t) = K_{eq} u_\theta(t - \tau_\theta) \tag{10}$$

Here,  $\tau_\theta$  includes the total loop delay, i.e. due to propulsion, communication, sensors, and digital circuits. Communication delay is inevitable and needs to be handled properly in the identification and control phases [27], [28]. System parameters are considered constant throughout a flight test.

#### a: Note on the validity of the linearized model

UAV dynamics can be considered nonlinear in normal operation due to the existence of aerodynamic drag and trigonometric nonlinearities. During the interaction between a human

and a UAV, the UAV is expected to operate at low speed and low tilt angles to ensure human safety. In such operating conditions, the effect of aerodynamic drag is negligible as compared to the human dynamics and the feedback controller dynamics. Under these conditions, as aerodynamic and trigonometric nonlinearities are negligible, the UAV model can be reasonably assumed linear.

The yaw dynamics has a different model structure compared to the other inner loop dynamics of a multirotor UAV and can be modelled as a second-order system [23]:

$$T_p \ddot{\psi}(t) + \dot{\psi}(t) = K_{eq} u_\psi(t - \tau_\psi) \tag{11}$$

Note that the propulsion dynamics are shared among all inner loops. Also, the drag time constant  $T_1$  and the equivalent gain  $K_{eq}$  would be different for every inner loop.

### D. MODEL PARAMETERS IDENTIFICATION

The unknown altitude and attitude models parameters can be identified in real-time using the recently developed approach of DNN-MRFT [29]. This method required decoupling of the UAV dynamics into SISO systems, as done in the previous section. Let the vector  $\mathbf{d} = [K_{eq} T_1 T_p \tau] \in D$  represent the model parameters of a given altitude or attitude loop. The model parameters' bounds provided in  $D$  are chosen such that all UAV designs of interest fall within. The DNN provides a map between a test signal produced experimentally, and the unknown parameters vector  $\mathbf{d}$ . The modified relay feedback test (MRFT) [30] is used to produce the system response, which is given by:

$$\begin{aligned}
u_M(t) &= \begin{cases} h & : e(t) \geq b_1 \vee (e(t) > -b_2 \wedge u_M(t-) = h) \\ -h & : e(t) \leq -b_2 \vee (e(t) < b_1 \wedge u_M(t-) = -h) \end{cases} \\
&\tag{12}
\end{aligned}$$

where  $b_1 = -\beta e_{min}$  and  $b_2 = \beta e_{max}$ .  $e_{max}$  and  $e_{min}$  are the maximum and minimum error signal values when the system undergoes stable oscillations.  $\beta$  is a tunable parameter which determines the phase of the excited oscillations as,  $\varphi = \arcsin \beta$ . In this paper,  $\beta$  is chosen as the global optimal  $\beta$  that minimizes ISE error for a step test as described in [31] and [32]. The MRFT always produces stable oscillations for the case of inner loops of multirotor UAVs as was shown in [23].

A classification DNN would provide more advantages compared to a regression DNN. First, the discretization of DNN output increases the efficiency of training of the DNN with a custom loss function that depends on optimal control parameters [29]. Second, a classification DNN would allow us to obtain optimal controllers of the UAV in real-time based on  $\mathbf{d} \in \bar{D}$ , where  $\bar{D}$  represents a discretization of  $D$  and  $|\bar{D}|$  is the number of the output classes of the DNN. Based on the simulated MRFT responses of all systems in  $\bar{D}$ , the DNN would select an element from  $\bar{D}$  that best describes the system under test.

For a process  $G_i(s)$  with model parameters  $d_i$ , there exists an optimal controller  $C_i(s)$  which minimizes a particular cost functional.  $\bar{D}$  is discretized in such a way that the percentage change in the integrated square error (ISE) when the controller  $C_i(s)$  is used with an adjacent process  $G_j$  is less than some predefined value  $J^*$  (in this paper we chose  $J^* < 10\%$  as in [29]). This can be formulated using the relative sensitivity function, which indicates the robustness of the system to the changes in process parameters and is governed by the following equation [33]:

$$J_{ij} = \frac{Q(C_i, G_j) - Q(C_j, G_j)}{Q(C_j, G_j)} \times 100\% \quad (13)$$

where  $J_{ij}$  represents the degradation in performance due to applying controller  $C_i$ , which is the optimal controller for the process  $G_i$  and a sub-optimal controller for the process  $G_j$ .  $Q$  denotes the integral square error (ISE) of the step response of the closed loop system:

$$Q(G(s), C(s)) = \frac{1}{T_s} \int_0^{T_s} e(G, C)^2 dt \quad (14)$$

### III. DESIGN OF TRAINING AND INFERENCE DOMAIN

To address the shortcoming of platform specific learning, we introduce the concept of a physical domain at which the UAV dynamics are characterized, and a TID at which the AI model is trained and the real-time inference occurs. For this to be achievable, the AI-model in TID has to be invariant to the specific UAV dynamics. Fig. 1 shows how a human-UAV ISP can be transformed from the physical domain to the TID using the DNN-MRFT identified model parameters. In this work, we have designed three different types of human-UAV interactions to be detected by three different LSTM models. These interactions are:

- 1) Single downward pull (SDP) of the UAV.
- 2) Consecutive double downward pulls (CDDP) of the UAV.
- 3) Single yawing twist (SYT) of the UAV.
- 4) Compound sequential interaction (CSI). Consists of multiple pull/push and yaw twists. The purpose of CSI is to highlight the applicability of the proposed approach to deal with more complex interactions.

A spring-mass damper system, depicted in Fig.2, can be used as an analogy for the analysis of the human-UAV interaction. In this analogy, the mass corresponds to the UAV, the first set of spring-damper corresponds to the closed loop UAV system, and the second set of spring-damper corresponds to the dynamics of the human arm. Human dynamics have been modelled as a spring-mass damper system in the literature [4]. The force exerted by the human is  $F_h$ . The spring mass damper system is described by:

$$m\ddot{x} = F_1 - F_2 \quad (15)$$

$$F_1 = -K_1(x) - b_1\dot{x} \quad (16)$$

$$F_2 = K_2(x - x_h) + b_2(\dot{x} - \dot{x}_h) \quad (17)$$

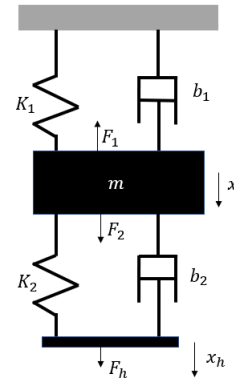


FIGURE 2. Spring mass damper analogy of the interaction.

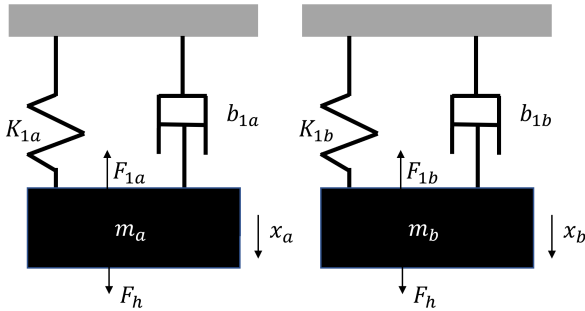
Comparing this system to the real human-UAV interaction system, the stiffness  $K_2$  and damping  $b_2$  terms of the human arm cannot be identified. Similarly, while the position  $x$  and acceleration  $\ddot{x}$  of the UAV are measured, while the position of the human arm  $x_h$  is not. The proposed learning-based method should implicitly estimate the force of the human hand  $F_h$  given the measurable parameters. As the human dynamics are unobservable, the LSTM should be robust enough that differences in  $K_2$  and  $b_2$  between different humans do not affect its output. This is done by including training data obtained from tests with different humans. This makes human dynamics part of the TID and hence, the LSTM effort is devoted to learning the human-invariant interaction behavior.

It should be noted that using the spring-damper analogy, the LSTM would overfit the dynamics characterized by  $K_1$ , and  $b_1$ . To overcome this limitation, we assume that the human exerted force profile is the same regardless of the UAV in use. To verify this assumption, we have conducted an experimental test where we have asked seven human subjects to detach a payload carried by a UAV. To make the handover maneuver as natural as possible, we used a cup of water as a payload and the subjects were not provided with any instructions apart from asking them to receive the payload by pulling it downwards. We recorded the estimated  $F_h$  of the first downward pull of the seven human subjects, and it was seen that the peak force differs only by 10% between all the subjects. As such, if  $F_h$  is assumed to be similar in all platforms, a relation between the responses of different spring mass damper UAV systems can be obtained. Under this assumption, consider two new spring-mass damper systems representing UAV (a) and UAV (b), with the same applied  $F_h$ , as in Fig.3. The response to an external force  $F_h$  is modelled by:

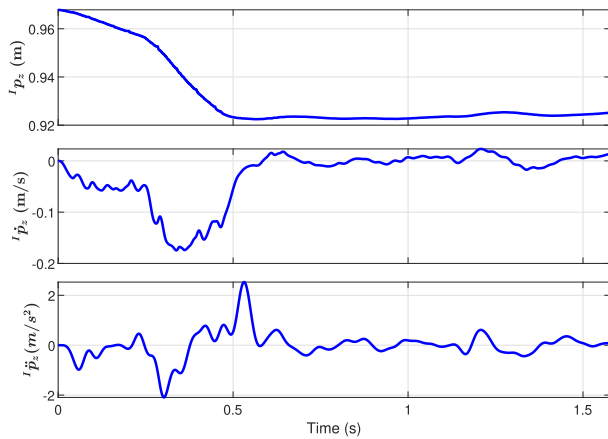
$$\begin{aligned} m_a \ddot{x}_a &= -K_{1a}x_a - b_{1a}\dot{x}_a + F_h \\ m_b \ddot{x}_b &= -K_{1b}x_b - b_{1b}\dot{x}_b + F_h \end{aligned} \quad (18)$$

equating the two forces gives the relation (written in the Laplace domain for convenience):

$$\frac{X_a(s)}{X_b(s)} = \frac{m_b s^2 + b_{1b}s + k_{1b}}{m_a s^2 + b_{1a}s + K_{1a}} \quad (19)$$



**FIGURE 3.** Two spring mass damper systems to compare the response to the same input force.



**FIGURE 4.** Experimentally obtained position, velocity and acceleration during a single downward pull interaction.

so if  $x_a$  is selected as a feature for LSTM training and all training happens based on the UAV platform (a), (19) can be used to transform data collected from UAV platform (b). In this case, the UAV platform (a) resides in the TID, and its dynamics are referred to as the *base* dynamics. UAV platform (b) resides in the physical domain, which might contain an arbitrary number of different UAV dynamics. In the rest of this section, we discuss the aspects of the selection of features for the considered interactions, i.e. SDP, CDDP, and SYT, and we derive transfer functions for the transformation of the proposed features from the physical domain to the TID.

### A. FEATURE SELECTION

To avoid the installation of additional sensors onboard a multirotor UAV, the interaction detection LSTM must depend on observable states measured by the UAV avionics. These selected states are called features of the ISP, from which the human-UAV interaction is detectable by the LSTM. Obviously, the features that are selected will depend on the type of interaction that needs to be detected. The states that can be estimated from a UAV platform fitted with IMU and a position and heading sensors is tabulated in Table 1. Note that Table 1 assumes the use of a motion capture (Mocap) system, which can be replaced by other positioning and heading measurement sources.

We have performed initial testing for the SDP, CDDP, and SYT interactions to select suitable states that would define the ISP for each. The features selected for the SDP, and CDDP interactions are shown in Table 2. The selected features for the SYT interaction detection LSTM are the following states: pitch  $\theta$ , roll  $\phi$ , yaw  $\psi$ , body accelerations  $[{}^B a_x \ {}^B a_y \ {}^B a_z]^T$ , and controller output of the yaw loop. For CSI we have concatenated the states used for SDP and SYT LSTM training.

**TABLE 1.** The state measurements that can be obtained for low-cost UAVs. In outdoor settings, the Mocap can be replaced by a combination of GPS, barometer, and magnetometer.

Estimated State	Sensor	Notes
${}^I_B R, \eta$	IMU and Mocap	Mocap for yaw
${}^B \omega$	Gyroscope	
${}^I p$	Mocap	
${}^I \dot{p}$	Mocap, and accelerometer	Sensor fusion
${}^I \ddot{p}$	Accelerometer	Uses ${}^I_B R$ to remove ${}^I g$

**TABLE 2.** LSTM features selected to represent the ISP for SDP and CDDP interactions. The state  $\eta$  represents the tilt angle of  $b_z$  with respect to  $i_z$ .

Feature	Equation
${}^I e_z$	${}^I p_{z_{ref}} - {}^I p_z$
$\ {}^I a\ $	$\sqrt{{}^I a_x^2 + {}^I a_y^2 + {}^I a_z^2}$
$\eta$	$\arctan\left(\frac{\sqrt{({}^B R i_x \cdot i_x)^2 + ({}^B R i_y \cdot i_y)^2}}{{}^B R i_z \cdot i_z}\right)$

### B. DYNAMICS TRANSFORMATION TO THE TRAINING AND INFERENCE DOMAIN

The UAV specific dynamics are required to define the transformation to the TID. The UAV inner loop dynamic parameters are assumed to be available through DNN-MRFT as described in Section II-D.

For the  ${}^I e_z$  feature, the altitude loop dynamics have to be considered, with the assumption that  $b_z$  and  $i_z$  are almost aligned (note that it is unsafe to interact with the drone otherwise). By proceeding with the assumption that  $F_h$  acting on two different UAVs are the same as stated in (18), we can write the following relation between the errors on two different UAVs differentiated by the indices 1 and 2. During experimentation, UAV 1 will be the one where the FTP detection system is trained and UAV 2 will be the new UAV where we test the dynamic transformation:

$$\begin{aligned} {}^I e_{z1} \frac{(K_{c1} + sK_{d1}) K_{p1} + m_1 s^2 (T_{p1} s + 1)}{T_{p1} s + 1} \\ = {}^I e_{z2} \frac{(K_{c2} + sK_{d2}) K_{p2} + m_2 s^2 (T_{p2} s + 1)}{T_{p2} s + 1} \end{aligned}$$

where  $K_c$  represents the proportional controller gain,  $K_d$  represents the derivative controller gain,  $K_p$  and  $T_p$  are the

propulsion system parameters from (8). Note that we assumed that the drag force and the delay in propulsion to be negligible during the interaction (i.e.  $T_1 = 0$ ). The above relation can be rearranged in a transfer function form to give, (20), as shown at the bottom of the page.

When fully defined, the transfer function in (20) provides a transformation of the  ${}^I e_{z_1}$  of UAV 2 in the physical domain, to the base dynamics in the TID. But realizing (20) requires two considerations. First, the states usually available on commercial multirotor UAVs do not include estimates for  ${}^I p_z^{(3)}$  and  ${}^I p_z^{(4)}$ . We assume the ratio of these higher order terms is close to unity, and hence we rewrite (20) to obtain the simplified transformation:

$$\frac{{}^I e_{z_1}(s)}{{}^I e_{z_2}(s)} = \frac{K_{c_2}K_{p_2} + K_{p_2}(T_{p_1}K_{c_2} + K_{d_2})s + (T_{p_1}K_{d_2}K_{p_2} + m_2)s^2}{K_{c_1}K_{p_1} + K_{p_1}(T_{p_2}K_{c_1} + K_{d_1})s + (T_{p_2}K_{d_1}K_{p_1} + m_1)s^2} \quad (21)$$

which is easy to realize due to the presence of  ${}^I \dot{p}_z$  and  ${}^I \ddot{p}_z$  estimates (refer to Table 1). The denominator in (21) acts as a filter for the noisy  ${}^I \ddot{p}_z$  measurements, which provides advantages for practical real-time realizations. The transformation of the feature  ${}^I \dot{p}_z$  can be obtained by neglecting the polynomial terms corresponding to position form (21), and similarly, a transformation for the acceleration feature  ${}^I a_z \equiv {}^I \ddot{p}_z$  can be obtained. Note that for the feature  $\|{}^I \mathbf{a}\|$  in Table 2 the same transformation as of  ${}^I a_z$  is assumed, which is a valid choice due to the UAV underactuated nature.

The second consideration for the practical realization of (21) is the availability of all the transformation parameters. The DNN-MRFT identifies the time parameter  $T_p$  and the equivalent loop gain  $K_{eq}$  as defined in (10), where  $K_{eq} = K_p/m$ , and also provides controller parameters  $K_c$  and  $K_d$ . But (21) requires the knowledge of both parameters  $K_p$  and  $m$ , so that one of these must be found prior to the interaction through lab tests. Obtaining the static gain parameters  $K_p$  and  $m$  using lab experiments is straightforward and is much easier compared to the other dynamic parameters present in (21) which are obtained through the DNN-MRFT. Thus, the DNN-MRFT provides an important automated step in the design of these dynamic transformations.

A similar transformation applied to altitude can be obtained for the angular dynamics. From Table 2, we have selected the tilt angle  $\eta$  as a feature which we assume to have the same dynamics as the pitch angle presented in (10). This assumption is valid for symmetric multirotor UAVs. Thus, in analogy to the altitude dynamics, the transformation for

$\eta$  dynamics is given by:

$$\frac{{}^I e_{\eta_1}(s)}{{}^I e_{\eta_2}(s)} = \frac{K_{c_2}K_{p_2} + K_{p_2}(T_{p_1}K_{c_2} + K_{d_2})s + (T_{p_1}K_{d_2}K_{p_2} + I_{y_2}/l_{y_2})s^2}{K_{c_1}K_{p_1} + K_{p_1}(T_{p_2}K_{c_1} + K_{d_1})s + (T_{p_2}K_{d_1}K_{p_1} + I_{y_1}/l_{y_1})s^2} \quad (22)$$

where  $I_{xy}$  corresponds to the rotational inertia of the tilt dynamics, and  $l_{xy}$  is the equivalent motor to center distance projected on  $\mathbf{b}_x \times \mathbf{b}_y$  which is assumed to be constant due to multirotor UAV symmetric design. Unlike the altitude case, a measurement for angular acceleration is not available. Thus, we truncate the transformation in (22) to be a first order transformation:

$$\frac{{}^I e_{\eta_1}(s)}{{}^I e_{\eta_2}(s)} = \frac{K_{c_2}K_{p_2} + K_{p_2}(T_{p_1}K_{c_2} + K_{d_2})s}{K_{c_1}K_{p_1} + K_{p_1}(T_{p_2}K_{c_1} + K_{d_1})s} \quad (23)$$

which is independent of  $I_{xy}$  and  $l_{xy}$ .

Another point to consider when transferring ISPs from the physical domain to the TID is the difference in the sampling rate used in both domains. The sampling rate is adjusted appropriately, either through downsampling, or by upsampling with linear interpolation. The LSTM is trained based on the sampling rate of the TID, which was chosen to be 1 kHz.

## IV. LSTM DESIGN

### A. ARCHITECTURE

Long short-term memory (LSTM) neural network is a special type of recurrent neural network (RNN) that is used in various applications with time-series data such as anomaly detection [34] and trajectory prediction [35]. They have the ability to store the previous data input in an internal state of the memory unit. The use of LSTM for interaction detection has advantages over other learning method such as RNNs or DNNs. LSTM mitigates the vanishing gradient problem faced by RNNs when training on long sequences of data, which is suitable for the detection of interaction sequences like in the CDDP case. Also, LSTMs have an advantage over DNNs and other memoryless approaches since LSTM do not require the selection of a time window. Thus LSTM is deemed suitable for the considered application. It is important to emphasize that dynamics invariance is independent of the learning method used in the TID, and could be extended to different learning methods.

The memory unit in the network keeps an internal state using the gating mechanism [36]. The gating mechanism for an LSTM cell consists of three gates: (1) Forget gate (2) Input gate (3) Output gate. A forget gate is a sigmoid activated network that decides which parts of the memory state is important, and which should be forgotten. The input

$$\frac{{}^I e_{z_1}(s)}{{}^I e_{z_2}(s)} = \frac{K_{c_2}K_{p_2} + K_{p_2}(T_{p_1}K_{c_2} + K_{d_2})s + (T_{p_1}K_{d_2}K_{p_2} + m_2)s^2 + m_2(T_{p_1} + T_{p_2})s^3 + m_2T_{p_1}T_{p_2}s^4}{K_{c_1}K_{p_1} + K_{p_1}(T_{p_2}K_{c_1} + K_{d_1})s + (T_{p_2}K_{d_1}K_{p_1} + m_1)s^2 + m_1(T_{p_2} + T_{p_1})s^3 + m_1T_{p_2}T_{p_1}s^4} \quad (20)$$



gate determines what new information should be added to the memory state and finally, the output gate determines which part of the memory state should be propagated to the next time step. In this paper, two layers of LSTM cells are adopted. This is referred to as a stacked LSTM model.

To determine the optimal neural network architecture for the problem at hand, an automated hyperparameter space searching technique was used. The search space included networks of varying depth, width, activation functions, and combinations of the features in Table 2, and was trained using two different optimizers, as listed in Table 3. A total of 55 different neural network structures were evaluated and the best-performing structure, depicted in Fig.5, was selected based on the prediction accuracy, where it achieved 98.3% accuracy on the testing dataset.

The adopted neural network structure consists of two LSTM layers with 200 and 100 units respectively. The LSTM layers implicitly estimate the force exerted by the human, as mentioned in Section III, by modelling (15, 16, 17). It is followed by a dense layer whose neurons are activated using the rectified linear unit (ReLU). This layer is applied at every time step by means of the timedistributed wrapper. The output layer consists of a single neuron activated using sigmoid function in order to perform binary classification. A classification threshold of 0.5 is used on the output neuron which is represented by a round-off function in Fig. 5. Dropout is used between the hidden layers to regularize the neural network and to avoid overfitting [37].

TABLE 3. Search space for the trained LSTM model.

Parameter	Search Space
number of layers	1,2,3
neurons per layer	100,200,300
activation function	ReLU
Optimizer	rmsprop, ADAM
features	2,3,4,6

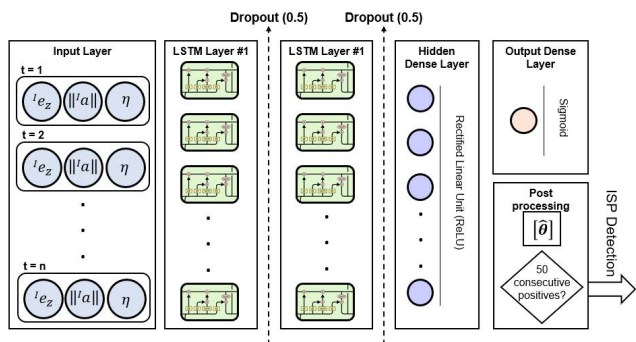


FIGURE 5. Proposed neural network architecture.

**B. DATA GENERATION AND TRAINING**

The training data for the neural network is generated from real experiments that are conducted on a UAV. This approach exposes the neural network to training data that closely resemble real-life scenarios. For further generalization, the training data is augmented by introducing noise. A bias of

0.05 and Gaussian noise with zero mean and a standard deviation of 0.0025 is introduced and appended to the original data. This provides a larger data set for training.

The neural network is required to identify human interaction and reject other disturbances that it might encounter during regular operation, such as wind or collisions. Hence, the training dataset included samples of Human-UAV interactions under wind disturbances and samples of random collisions with the UAV.

Collected training data is divided into sequences, the length of which is selected based on the ISP of interest. For instance, the profile resulting from a single pull, and hence the sequence length, is shorter than the profile resulting from a double pull. This will ensure that the desired profile lies within the input sequence to the neural network which will guarantee correct prediction. The order of measurements in the sequence will remain unchanged to maintain the characteristic of the profile over time. A total of 14652 sequences were collected, 50% of which were used for training, 25% for validation, and 25% for testing. Prediction accuracy on the testing set will be used for model evaluation.

**C. REAL-TIME INFERENCE**

During Real-time inference stage, a stateful LSTM, which gives us the ability to reset the internal states according to our requirement, is used. The internal states of the LSTM are reset when the UAV has taken off and is hovering, awaiting human interaction.

We employ a two-stage discriminator for the neural network. Firstly, the output of the LSTM is a probability distribution  $\mathcal{P} \in [0, 1]$ . We chose an output  $\mathcal{P} > 0.5$  to represent a positive detection. Secondly, the LSTM model provides a detection result for every time step, however, outlier positive detections are rejected by accepting positive signals that are sustained for a predefined number of time steps, chosen to be 50 ms. This conservative approach is preferred due to the nature of the application at hand, where it is preferred not to release the payload unless a receptor human is ready to receive it. To measure the accuracy of the system, a true positive is defined when at least one positive output is present during the interaction, and a false positive when there is a positive output without an interaction.

**V. EXPERIMENTAL RESULTS**

The proposed LSTM-based ISP detection approach was extensively tested in experimentation using different UAV platforms across multiple scenarios. More specifically, the trained ISP detection model was tested on a quadcopter and a hexacopter while hovering, while moving, under wind disturbance, and using different human force signatures. This section provides, in detail, a description of the experimental setup and an analysis of the obtained results.

**A. EXPERIMENTAL SETUP**

Two different UAVs are used to verify the applicability of the proposed approach and to demonstrate its generality across



FIGURE 6. Qdrone.

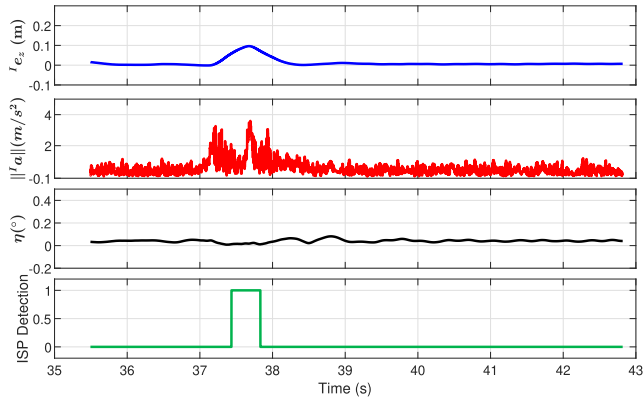


FIGURE 7. Detection of a single downward pull exerted on the base UAV.

various platforms. A quadcopter, which we refer to as the base UAV, was used for collecting training data and for initial verification, and a hexacopter, which we refer to as the testing UAV, was used for verifying the generality of the proposed approach. The specifications of the base and testing UAVs are provided below.

- **Base UAV:** The base UAV is a Quanser Qdrone quadcopter, that comes with ducted propellers (Fig. 6) which makes it suitable for physical Human-UAV interaction. The Qdrone is a small, 1000g, quadrotor, equipped with an Intel Aero board and a BMI160 onboard IMU. A proprietary *Matlab/Simulink* interface is used to communicate with the drone. A motion capture system is used to measure the position of the UAV at 120Hz.
- **Testing UAV:** The DJI F550 Hexarotor UAV is used to test all the neural networks that were trained on the base UAV. The testing UAV is a 2260g hexarotor that uses the NAVIO2 flight controller hat with a Raspberry pi3B+, where a custom flight control software is run. Xsens 610 is used as an onboard IMU. The LSTM network runs on an onboard intel NUC, since running it on the same RPi3B+ caused major delays in the system. The robot operating system (ROS) is used for communication among the flight controller, the LSTM network, and the motion capture system.

The experimentation for this research is set up as follows: firstly, experiments are conducted to show that the

TABLE 4. Identification of the base and test UAVs in the altitude loop.

UAV	$K_p$	$K_d$	$K$	$T_p$	$T_1$	$\tau$
Base UAV	75	13	0.1415	0.0224	0.2776	0.0656
Testing UAV	24.809	7.4476	0.5090	0.3	0.2	0.0128

TABLE 5. confusion matrix for base UAV.

	predicted False	predicted True
actual False	40	0
actual True	2	8

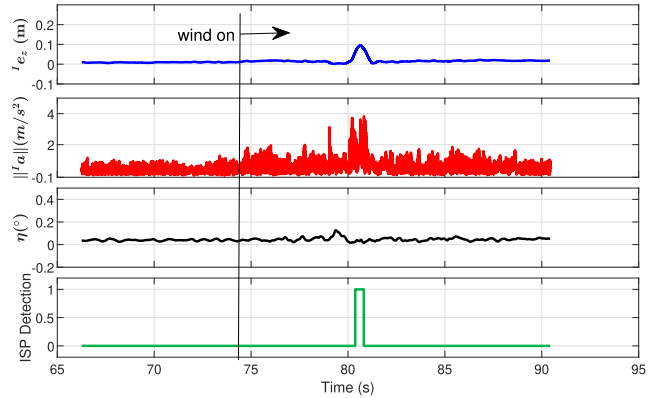


FIGURE 8. Detection of a single downward pull exerted on the base UAV under wind disturbance (at 5m/s).

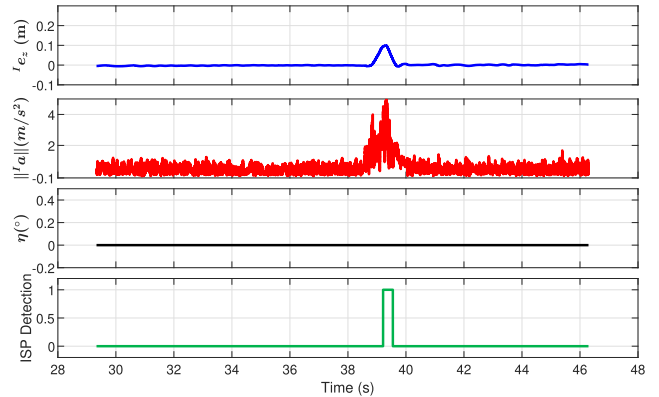
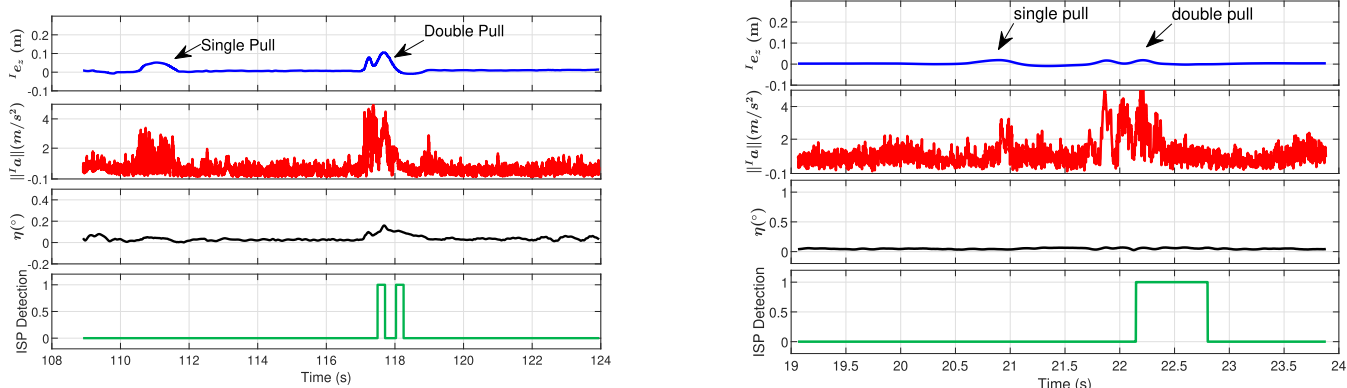


FIGURE 9. Detection of a single downward pull exerted on the base UAV while moving at 10cm/s.

LSTM network can reliably detect a human interaction on the base UAV it was trained on. Then the generalization of the trained LSTM to different UAVs is demonstrated with the testing UAV. Finally, an application for the system in the form of payload delivery is then demonstrated.

System identification of the two platforms is performed as mentioned in II-D; the resulting parameters are shown in Table. 4. Knowledge of the model for both platforms is necessary when using the ISP detection on a different platform as indicated in (20) and (22). The parameters of the altitude channel is tabulated in Table 4.



**FIGURE 10.** Detection of a consecutive downward double pull exerted on the base UAV (left plot), and the test UAV (right plot) using the same LSTM. The single pull was not detected in both cases, as expected.

**B. SINGLE DOWN PULL EXPERIMENTS**

In this experiment, we test the ability of our method to detect an SDP trained and tested on the base UAV. To validate our method, we perform 10 tests where the human intends to interact with the UAV (positive tests) and 40 tests where the forces are applied on the UAV without the intention to interact (negative tests). During the negative tests, random forces, with varying application time are applied on the UAV in random directions. The confusion matrix of the above tests is shown in Table. 5, with an accuracy of 96%. Fig. 7 shows one example SDP experiment. This figure shows the modification in altitude of the platform  $l_{e_z}$ , the platform acceleration norm  $\|l_a\|$ , yaw  $\eta$ , and the ISP detection flag.

**C. ISP DETECTION UNDER WIND DISTURBANCE AND UAV MOTION**

Once the ISP detection is validated on the base UAV, experiments are conducted to check the robustness of the trained detector to externally injected disturbances. A disturbance that is normally faced by UAVs is wind. To test the robustness of our approach to wind disturbances, we test the trained detector in the presence of lab generated wind, with speeds reaching 5m/s. These experiments concluded that the proposed approach is able to distinguish wind disturbances from human interactions. Fig. 8 shows the deployment of our detector in the presence of wind, and demonstrates that the detector positively identifies the human interaction only despite the wind disturbance.

In order to assess the detector’s performance while the platform is in motion, an experiment is conducted where the UAV is commanded to move in the y direction at a speed of 10cm/s; the speed was intentionally chosen small enough for the human to be able to interact with the platform in a safe manner. The detection of ISP while the UAV is moving is plotted in Fig. 9, showing the effectiveness of our detector in spite of the UAV motion.

**D. DETECTING DIFFERENT ISPs**

To show that our detector learns to find force profiles instead of simple force thresholding, an experiment is designed

where two consecutive pulls must be detected and lone pulls should be rejected, even though the direction of the pull is the same in both profiles.

In this experiment, our detector is trained to detect two consecutive pulls, and is tested on single and double pulls. Fig. 10 (left) shows the results of this experiment. In this figure we can see the detector clearly identifying the double pull, while rejecting the single pull.

Finally, the results of the more complex CSI detection profile are shown in Fig. 11. This figure shows the positive detection of the CSI interaction profile twice. This figure also shows that the LSTM-network trained to detect the CSI profile is robust against the detection of the SDP interaction profile.

Similar results were observed for the SYT ISP detection. We omitted the results of these experiments from this paper for brevity, however, these experiments are shown in the accompanying video [24].

**E. DYNAMICS TRANSFERABILITY**

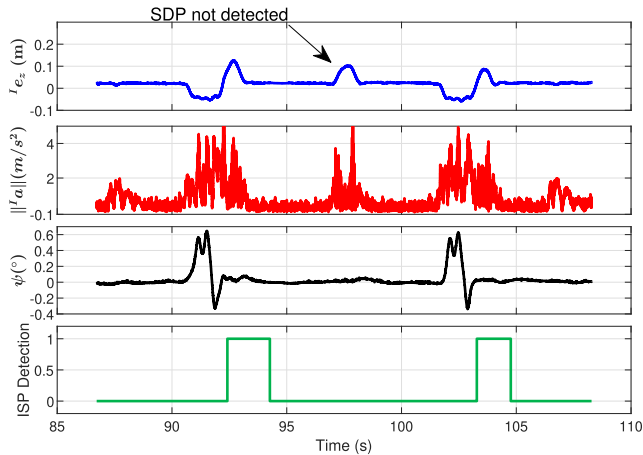
As the CDDP ISP is the most complex between the three tested ISPs, we test the dynamic transformation between two different UAVs while detecting this ISP. The testing UAV is only used to validate the dynamic transformation from the physical domain to the TID using the parameters from Table. 4.

The results of this experiment are shown in Fig. 10(right). From this figure, we can see that the human interaction is reliably detected on the testing UAV after the necessary dynamic scaling of the features.

It is worth noting that the same experiment was conducted with the testing UAV without any dynamic transformation. As expected, during this experiment, the testing UAV was not capable of detecting the human interaction.

**F. PAYLOAD DELIVERY**

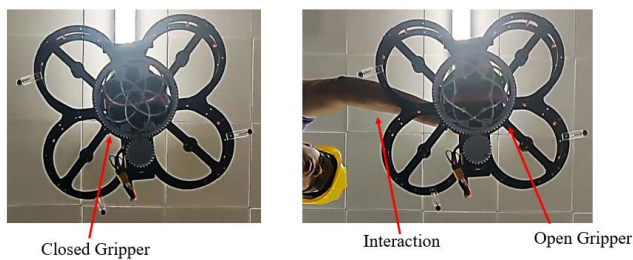
As an application to the ISP detection, we propose payload handover from a UAV to a human. The interaction detection system itself can be extended to other applications, however,



**FIGURE 11.** Detection of a Compound sequential interaction (CSI) consisting of the following sequence: push-and-hold the drone upwards, then double positive yaw twists, followed by a single negative yaw twist, and finally a single downward pull. This figure shows the detection of the considered CSI twice, and shows that the same network does not detect the simple SDP.



**FIGURE 12.** Interaction with the Test UAV in a safe and controlled setting.



**FIGURE 13.** showing the designed gripper's operation. The gripper is closed when no interaction is detected. When the human interacts with the UAV, the gripper opens to release the payload.

this application is required in real-world payload delivery problems such as the ones in [38] and [39], and can be used to validate our ISP detection method without any additional sensors or actuators that modify the platform's dynamics.

To offer a smooth user experience, once the UAV detects a human interaction, it should stop resisting the human and release it in a smooth manner. When an interaction is detected, the UAV is commanded to hold position for a set amount of time. This is done by changing the position reference to the estimated position at the start of the interaction.

We designed a gripper based on the iris gripper [40], that holds and releases the payload as required by the corresponding interaction. The gripper is built using lightweight 3D printed material to allow maximum payload capacity, and is controlled with a servo motor (Dynamixel Ax-12a). The gripper is firmly attached under the UAV and it can smoothly close and firmly grip the desired payload, and then, once an interaction is detected, it is commanded to open and release the payload. The gripper and the described operation is shown in Fig. 13

## VI. CONCLUSION

In this paper, a novel method for interaction detection between a human and a UAV is proposed. The presented method detects a state signature, referred to as the interaction states profile (ISP), exerted by the human on the UAV. The presented detection scheme is then demonstrated in a payload handover scenario, where following the ISP detection, the UAV releases the object for the human. The presented method is trained on a base UAV, with an LSTM-based neural network. The method is then transferred to other UAVs through a dynamics based transformation between the base UAV and the testing UAV, rendering the method agnostic to the training platform. The dynamics of the base and testing UAVs are exposed using the DNN-MRFT method [29]. The presented approach is validated through an extensive experimental campaign, showing the detection of different ISPs at high success rate, the robustness of the presented method to wind disturbance, and the transferability between different platforms.

In the future, this work could be extended in different directions. For example, a bidirectional human-robot handover system can be developed to consider the case where a human hands an object over to a robot. In addition, the proposed approach can be extended to distinguish between multiple classes of force signatures simultaneously. Finally, while our method requires physical interaction between the human and the UAV to detect the interaction, the intention of the human to interact prior to the contact could be added to the handover scenario as was done in [41] and [42].

In addition, the dynamics-based learning transfer method presented in this paper could be extended to new applications. While the method was demonstrated in this paper for a human-UAV interaction task, its applications are much wider, and we hope to expand this method into applications in UAV interaction with the surrounding environment like interaction with soil [43] and other similar mediums.

## REFERENCES

- [1] H. Shakhatreh, A. H. Sawalmeh, A. Al-Fuqaha, Z. Dou, E. Almaita, I. Khalil, N. S. Othman, A. Khreishah, and M. Guizani, "Unmanned aerial vehicles (UAVs): A survey on civil applications and key research challenges," *IEEE Access*, vol. 7, pp. 48572–48634, 2019.
- [2] F. Veroustraete, "The rise of the drones in agriculture," *EC Agricult.*, vol. 2, pp. 325–327, Aug. 2019.
- [3] O. Maghazei, "Drones in manufacturing: Opportunities and challenges," Ph.D. dissertation, Dept. Manag., Technol., Econ., ETH Zürich, Zürich, Switzerland, 2021.

- [4] M. Allenspach, Y. Vyas, M. Rubio, R. Siegwart, and M. Togno, “Human-state-aware controller for a tethered aerial robot guiding a human by physical interaction,” *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 2827–2834, Apr. 2022.
- [5] A. Afifi, M. van Holland, and A. Franchi, “Toward physical human–robot interaction control with aerial manipulators: Compliance, redundancy resolution, and input limits,” in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2022, pp. 4855–4861.
- [6] E. Matheson, R. Minto, E. G. Zampieri, M. Faccio, and G. Rosati, “Human–robot collaboration in manufacturing applications: A review,” *Robotics*, vol. 8, no. 4, p. 100, Dec. 2019.
- [7] J. Mainprice, M. Gharbi, T. Simeon, and R. Alami, “Sharing effort in planning human–robot handover tasks,” in *Proc. IEEE RO-MAN, 21st IEEE Int. Symp. Robot Hum. Interact. Commun.*, Sep. 2012, pp. 764–770.
- [8] E. Cuniato, J. Cacace, M. Selvaggio, F. Ruggiero, and V. Lippiello, “A hardware-in-the-loop simulator for physical human–aerial manipulator cooperation,” in *Proc. 20th Int. Conf. Adv. Robot. (ICAR)*, Dec. 2021, pp. 830–835.
- [9] H. Khambhaita and R. Alami, “Assessing the social criteria for human–robot collaborative navigation: A comparison of human-aware navigation planners,” in *Proc. 26th IEEE Int. Symp. Robot Hum. Interact. Commun. (RO-MAN)*, Aug. 2017, pp. 1140–1145.
- [10] M. Hamandi, M. D’Arcy, and P. Fazli, “DeepMoTion: Learning to navigate like humans,” in *Proc. 28th IEEE Int. Conf. Robot Hum. Interact. Commun. (RO-MAN)*, Oct. 2019, pp. 1–7.
- [11] K. Strabala, M. K. Lee, A. Dragan, J. Forlizzi, S. S. Srinivasa, M. Cakmak, and V. Micelli, “Toward seamless human–robot handovers,” *J. Hum.-Robot Interact.*, vol. 2, no. 1, pp. 112–132, Feb. 2013, doi: 10.5898/JHRI.2.1.Strabala.
- [12] M. Hamandi, E. Hatay, and P. Fazli, “Predicting the target in human–robot manipulation tasks,” in *Proc. Int. Conf. Social Robot. Cham, Switzerland: Springer*, 2018, pp. 580–587.
- [13] R. Zhang, L. Wu, Y. Yang, W. Wu, Y. Chen, and M. Xu, “Multi-camera multi-player tracking with deep player identification in sports video,” *Pattern Recognit.*, vol. 102, Jun. 2020, Art. no. 107260.
- [14] J. R. Medina, F. Duvallet, M. Karnam, and A. Billard, “A human-inspired controller for fluid human–robot handovers,” in *Proc. IEEE-RAS 16th Int. Conf. Humanoid Robots (Humanoids)*, Nov. 2016, pp. 324–331.
- [15] A. Gómez Eguíluz, I. Rañó, S. A. Coleman, and T. M. McGinnity, “Reliable robotic handovers through tactile sensing,” *Auto. Robots*, vol. 43, no. 7, pp. 1623–1637, Jan. 2019, doi: 10.1007/s10514-018-09823-2.
- [16] N. Hendrich, H. Bistry, J. Liebrecht, and J. Zhang, “Natural robot–human handover combining force and tactile sensors,” in *Proc. IEEE/RSJ Intl. Conf. Intell. Robots Syst.*, 2014, pp. 1–7.
- [17] L. Roveda and D. Piga, “Sensorless environment stiffness and interaction force estimation for impedance control tuning in robotized interaction tasks,” *Auto. Robots*, vol. 45, no. 3, pp. 371–388, 2021, doi: 10.1007/s10514-021-09970-z.
- [18] M. Hanafusa and J. Ishikawa, “External force estimation for nonlinear systems using recurrent neural network,” in *Proc. IEEE/ASME Int. Conf. Adv. Intell. Mechatronics (AIM)*, Jul. 2019, pp. 1055–1061.
- [19] B. Yüksel, C. Secchi, H. H. Bühlhoff, and A. Franchi, “Aerial physical interaction via IDA-PBC,” *Int. J. Robot. Res.*, vol. 38, no. 4, pp. 403–421, Apr. 2019, doi: 10.1177/0278364919835605.
- [20] S. Rajappa, H. Bulthoff, and P. Stegagno, “Design and implementation of a novel architecture for physical human–UAV interaction,” *Int. J. Robot. Res.*, vol. 36, nos. 5–7, pp. 800–819, May 2017, doi: 10.1177/0278364917708038.
- [21] C. D. McKinnon and A. P. Schoellig, “Unscented external force and torque estimation for quadrotors,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2016, doi: 10.1109/IROS.2016.7759831.
- [22] T. Tomic and S. Haddadin, “A unified framework for external wrench estimation, interaction control and collision reflexes for flying robots,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Sep. 2014, pp. 4197–4204, doi: 10.1109/IROS.2014.6943154.
- [23] A. Ayyad, M. Chehadeh, P. H. Silva, M. Wahbah, O. A. Hay, I. Boiko, and Y. Zweiri, “Multirotors from takeoff to real-time full identification using the modified relay feedback test and deep neural networks,” *IEEE Trans. Control Syst. Technol.*, vol. 30, no. 4, pp. 1561–1577, Jul. 2022.
- [24] A. Peringal. (2022). *Experiment Video*. [Online]. Available: <https://youtu.be/WcXa1H558UA>
- [25] M. Hamandi, F. Usai, Q. Sablé, N. Staub, M. Togno, and A. Franchi, “Design of multirotor aerial vehicles: A taxonomy based on input allocation,” *Int. J. Robot. Res.*, vol. 40, nos. 8–9, pp. 1015–1044, Aug. 2021.
- [26] C. Cheron, A. Dennis, V. Semerjyan, and Y. Chen, “A multifunctional HIL testbed for multirotor VTOL UAV actuator,” in *Proc. IEEE/ASME Int. Conf. Mech. Embedded Syst. Appl.*, Jul. 2010, pp. 44–48, doi: 10.1109/MESA.2010.5552032.
- [27] C. Yan, L. Fu, J. Zhang, and J. Wang, “A comprehensive survey on UAV communication channel modeling,” *IEEE Access*, vol. 7, pp. 107769–107792, 2019.
- [28] Z. Fang, J. Wang, Y. Ren, Z. Han, H. V. Poor, and L. Hanzo, “Age of information in energy harvesting aided massive multiple access networks,” *IEEE J. Sel. Areas Commun.*, vol. 40, no. 5, pp. 1441–1456, May 2022.
- [29] A. Ayyad, M. Chehadeh, M. I. Awad, and Y. Zweiri, “Real-time system identification using deep learning for linear processes with application to unmanned aerial vehicles,” *IEEE Access*, vol. 8, pp. 122539–122553, 2020, doi: 10.1109/ACCESS.2020.3006277.
- [30] I. M. Boiko, “Loop tuning with specification on gain and phase margins via modified second-order sliding mode control algorithm,” *Int. J. Syst. Sci.*, vol. 43, no. 1, pp. 97–104, Jan. 2012, doi: 10.1080/00207721003790344.
- [31] M. S. Chehadeh and I. Boiko, “Design of rules for in-flight non-parametric tuning of PID controllers for unmanned aerial vehicles,” *J. Franklin Inst.*, vol. 356, no. 1, pp. 474–491, Jan. 2019, doi: 10.1016/j.jfranklin.2018.10.015.
- [32] I. Boiko, “Design of non-parametric process-specific optimal tuning rules for PID control of flow loops,” *J. Franklin Inst.*, vol. 351, no. 2, pp. 964–985, Feb. 2014, doi: 10.1016/j.jfranklin.2013.09.011.
- [33] R. Rohrer and M. Sobral, “Sensitivity considerations in optimal system design,” *IEEE Trans. Autom. Control*, vol. AC-10, no. 1, pp. 43–48, Jan. 1965.
- [34] P. Malhotra, L. Vig, G. Shroff, and P. Agarwal, “Long short term memory networks for anomaly detection in time series,” in *Proc. ESANN*, vol. 89, pp. 89–94.
- [35] S. Dai, L. Li, and Z. Li, “Modeling vehicle interactions via modified LSTM models for trajectory prediction,” *IEEE Access*, vol. 7, pp. 38287–38296, 2019.
- [36] F. A. Gers, J. Schmidhuber, and F. Cummins, “Learning to forget: Continual prediction with LSTM,” *Neural Comput.*, vol. 12, no. 10, pp. 2451–2471, 2000.
- [37] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: A simple way to prevent neural networks from overfitting,” *J. Mach. Learn. Res.*, vol. 15, pp. 1929–1958, Sep. 2014.
- [38] Amazon. (2016). Amazon.com. [Online]. Available: <https://www.amazon.com/Amazon-Prime-Air/b/?node=8037720011>
- [39] P. Wing. *Google Drones Will Deliver Chipotle Burritos At Virginia Tech*. Accessed: Apr. 7, 2022. [Online]. Available: <https://money.cnn.com/2016/09/08/technology/google-drone-chipotle-burrito/index.html>
- [40] Flexsys. (2019). *Adaptive Compliant Grippers*. Accessed: Apr. 7, 2022. [Online]. Available: <https://www.flxsys.com/grippers>
- [41] M. Hamandi and P. Fazli, “Online learning of human navigational intentions,” in *Social Robotics*. Cham, Switzerland: Springer, 2018, pp. 1–10.
- [42] M. Hamandi, E. Hatay, and P. Fazli, “Predicting the target in human–robot manipulation tasks,” in *Social Robotics*. Cham, Switzerland: Springer, 2018, pp. 580–587.
- [43] S. Hutangkabodee, Y. H. Zweiri, L. D. Seneviratne, and K. Althoefer, “Performance prediction of a wheeled vehicle on unknown terrain using identified soil parameters,” in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2006, pp. 3356–3361.



**ANEES PERINGAL** received the B.Sc. degree in aerospace engineering from Khalifa University, Abu Dhabi, where he is currently pursuing the M.Sc. degree in aerospace engineering. His research interests include control of dynamic systems and autonomous robotics.



**MOHAMAD CHEHADEH** (Member, IEEE) received the M.Sc. degree in electrical engineering from Khalifa University, Abu Dhabi, UAE, in 2017. He is currently with the Khalifa University Center for Autonomous Robotic Systems (KUCARS). His research interests include identification, perception, and control of complex dynamical systems utilizing the recent advancements in the field of AI.



**RANA AZZAM** received the B.Sc. degree in computer engineering and the M.Sc. degree by research in electrical and computer engineering from Khalifa University, in 2014 and 2016, respectively, and the Ph.D. degree in engineering with a focus on robotics, in 2020. She is currently a Postdoctoral Fellow with the Department of Aerospace Engineering. Her research interests include machine learning, reinforcement learning, navigation, and simultaneous localization and mapping.



**MAHMOUD HAMANDI** received the M.Eng. degree in mechanical engineering from the American University of Beirut, Beirut, Lebanon, in 2017, and the Ph.D. degree in automation and robotics from the National Institute for Applied Sciences, Toulouse, France, in 2021. He is currently a Postdoctoral Fellow with the Khalifa University Center for Autonomous Robotic Systems (KUCARS). His research interests include perception, design, and control of robotic systems, with applications to aerial robots.



**IGOR BOIKO** (Senior Member, IEEE) received the M.Sc., Ph.D., and D.Sc. degrees from Tula State University and Higher Attestation Commission, Russia. Currently, he is a Professor with Khalifa University, Abu Dhabi, UAE. His research interests include frequency-domain methods of analysis and design of nonlinear systems, discontinuous and sliding mode control systems, PID control, process control theory, and applications.



**YAHYA ZWEIRI** (Member, IEEE) received the Ph.D. degree from the King's College London, in 2003. He is currently an Associate Professor with the Department of Aerospace Engineering and the Deputy Director of Advanced Research and Innovation Center, Khalifa University, United Arab Emirates. He was involved in defense and security research projects in the last 20 years at the Defense Science and Technology Laboratory, King's College London, and the King Abdullah II Design and Development Bureau, Jordan. He has published over 130 refereed journals and conference papers and filed ten patents in USA and U.K., in the unmanned systems field. His main research interests include robotic systems for extreme conditions with particular emphasis on applied artificial intelligence (AI) aspects and neuromorphic vision systems.

...