

Autonomous Drone for Human/Object Following using PID Control and Real Time Image Recognition

Soham N. Kulkarni
Department of Electronics and
Telecommunication Engineering.
Vishwakarma Institute of
Technology
Pune, India
soham.kulkarni243@vit.edu

Madhur V. Gomase
Department of Electronics and
Telecommunication Engineering.
Vishwakarma Institute of
Technology
Pune, India
madhur.gomase24@vit.edu

Yash P. Malode
Department of Electronics and
Telecommunication Engineering.
Vishwakarma Institute of
Technology, Pune, India
yash.malode24@vit.edu

Dr. Arti V. Bang
Department of Electronics and
Telecommunication Engineering.
Vishwakarma Institute of
Technology
Pune, India
arti.bang@vit.edu

Abstract — This paper presents the design of an autonomous drone capable of tracking and following a human or object using real-time image recognition and a Proportional–Integral–Derivative (PID) control system. The drone employs a Raspberry Pi 5 as its onboard processor, enabling fast computation and efficient image analysis through OpenCV. Visual data from the Pi Camera v2 are processed using YOLO and Haar-cascade models to detect the target, and the resulting positional error is fed into a custom-tuned PID controller implemented in Python. The controller dynamically adjusts the drone's roll, pitch, and throttle for stable tracking. Communication with the MicoAir F405 Mini flight controller occurs via the MAVLink protocol, ensuring low-latency feedback. The system also supports gesture-based functions such as autonomous take-off, landing, and palm landing for intuitive control. Pre-deployment validation is carried out in Gazebo and AirSim simulations. The proposed design provides a modular, low-cost, and interactive solution for real-time human or object following, suitable for applications in education, surveillance, and rescue robotics.

Keywords —Autonomous Drone, Computer Vision, Gesture Control, MAVLink Communication, Object Tracking, PID Control, MicoAir F405 Mini, Raspberry Pi 5, Real-Time Image Processing, YOLO Detection

INTRODUCTION

Unmanned Aerial Vehicles (UAVs), in particular quadrotors, have emerged as effective platforms in various applications, including surveillance, inspection, and assistive robotics. The recent advances in computing and computer vision have allowed UAVs to perform autonomous operations such as object detection and dynamic target tracking. The benefits of using computer vision in UAVs include reduced dependency on positioning infrastructure and greater flexibility in deployment in different environments.

The tracking of humans using UAVs faces various challenges. The UAV must be able to detect the target, calculate the deviation in the target's position in real time, and adjust the flight characteristics of the UAV using robust mechanisms to ensure flight stability. These requirements impose various constraints on the computing capabilities of the system. The

quadrotor UAVs are nonlinear and underactuated machines and demand robust controllers to handle various disturbances and changes in motion.

Existing research has also focused on advanced control techniques such as optimal control techniques, non-linear trajectory optimization techniques, and visual-inertial simultaneous localization and mapping techniques. Although such techniques promise high accuracy and reliability, they require considerable computational capabilities and sophisticated sensor fusion techniques. Such techniques increase the overall cost of implementation for a low-cost embedded system. Furthermore, there are existing vision-based tracking techniques that require considerable computational capabilities and may not be optimized for real-time processing.

The proposed work focuses on designing a vision-based autonomous tracking of a human target by a quadrotor using an embedded system architecture. The proposed system includes a combination of real-time object tracking, image-based error calculation, and Proportional-Integral-Derivative (PID) based closed-loop control techniques for stabilization. The communication protocol for data exchange between the processor and flight controller is based on MAVLink protocol. The proposed system is validated through simulation before actual implementation to ensure reliable evaluation of tracking performance.

The proposed work contributes to existing research by proposing a combination of real-time object tracking and classical control techniques for autonomous tracking using a low-cost embedded system, a modular communication protocol for UAVs, and simulation-based validation of autonomous tracking of a human target by a quadrotor.

I. LITERATURE REVIEW

Vision-based autonomous unmanned aerial vehicles (UAVs) have been thoroughly researched in the fields of object detection and tracking, quadrotor control, and state estimation. The literature in the aforementioned fields can be broadly classified

into the following categories: (A) Vision-Based Object Detection and Tracking, (B) Quadrotor Modeling and Control Strategies, and (C) Visual-Inertial Navigation and Simulation-Based Validation.

A. Vision Based Object Detection and Tracking.

Recent advancements in deep learning have significantly improved object detection accuracy in real-time. The YOLO framework provided a unified framework in object detection by reformulating object detection as a regression problem, allowing real-time object detection speeds required in embedded systems [1]. Various object tracking methodologies have been developed on top of object detection frameworks to provide multi-object tracking capabilities. Deep SORT employs Kalman filtering with deep appearance descriptors to maintain object identity over time, while similarity learning using Siamese networks provides high-speed object tracking capabilities without requiring network retraining.

While object tracking capabilities of these methodologies have been promising, many require significant computational capabilities, especially in multi-object detection scenarios. Balancing object detection accuracy and speed is a major challenge in embedded object tracking in unmanned aerial vehicle (UAV) systems. Vision-based object tracking implementations on embedded platforms have been explored in ground robotics using Raspberry Pi-based architectures in ground robotics scenarios [7], but no similar implementations have been proposed for embedded object tracking in aerial platforms. This work focuses on real-time single-target object tracking in humans using object detection techniques optimized for embedded systems, in contrast to computationally expensive object tracking methodologies proposed in the literature.

B. Quadrotor Modelling and Control Strategies.

Quadrotor unmanned aerial vehicles (UAVs) display nonlinear and underactuated behavior, involving coupled translation and rotational dynamics. Comprehensive reviews on various aspects of quadrotor modeling and their associated control strategies can be found in reference [4]. The various difficulties associated with the stabilization, rejection of external disturbances, and attitude control of such platforms have been discussed. Earlier works on such platforms primarily focused on comparing classical Proportional-Integral-Derivative (PID) controllers with Linear Quadratic (LQ) based approaches, thereby addressing the need for a judicious balance between simplicity and optimality. Despite recent advances in various aspects of controller design for such platforms, classical PID-based approaches continue to dominate their implementation for various reasons, as discussed in reference [3]. The application of trajectory optimization techniques, such as minimum snap trajectory generation, has also been discussed for various aggressive flight maneuvers and precision trajectory tracking for such platforms in references [2] and [8]. However, such approaches assume a predefined trajectory and a good knowledge of system dynamics. For a vision-based tracking application involving a human target, such approaches may not be feasible, and hence a light-weight closed-loop controller may prove to be more suitable for such autonomous tracking platforms.

Literature surveys on various aspects of guidance, navigation, and control of unmanned rotorcraft systems highlight the recent

advancements in such platforms, involving a combination of perception and control strategies for autonomous unmanned aerial vehicles. However, such advanced approaches require high-performance processors or external computing platforms, thereby adding to system complexity and cost. In this work, a classical PID-based approach has been adapted for system stabilization.

C. Visual-Inertial Navigation and Simulation Environments

Autonomous UAV navigation systems frequently utilize visual-inertial odometry and SLAM techniques for better localization accuracy. For example, keyframe-based visual-inertial optimization techniques achieve high accuracy in state estimation for micro aerial vehicles in [15], while visual-inertial SLAM techniques achieve localization and mapping in unknown environments for micro aerial vehicles in [16], [17]. However, these techniques also increase computational overhead and complexity in sensor fusion techniques for UAV navigation systems.

Simulation environments play a vital role in validating autonomous UAV systems before real-world deployment. AirSim is a high-fidelity physics and visual simulation for autonomous vehicle research in [9], while Gazebo is a flexible and open-source multi-robot simulation in [10]. These simulation environments are useful in validating perception-control integration and stability for autonomous UAV systems.

In contrast to SLAM techniques and sensor-fusion techniques, the proposed framework focuses on relative positioning techniques for human tracking without environmental mapping. The proposed system utilizes simulation-based validation for perception-control integration while maintaining a lightweight embedded system for low-cost UAVs.

II. SYSTEM ARCHITECTURE

The system under consideration utilizes a hierarchical architecture with a high-level perception and control layer, and a low-level flight stabilization layer, as shown in Figure 1. The high-level layer is implemented using a Raspberry Pi 5, where real-time object detection is performed using the visual information provided by the monocular camera. Object detection is performed to locate the position of the human target, followed by computation of the centroid position within the image frame. The deviation between the detected target position and the center of the image frame is calculated to generate the horizontal and vertical error values.

The generated values are used to drive the closed-loop Proportional-Integral-Derivative (PID) controller, which is implemented using the Raspberry Pi. The output values generated by the controller correspond to the necessary corrections to maintain the target at the center of the image frame. The generated values are sent to the flight controller using the MAVLink communication protocol.

The flight controller, i.e., the MicoAir F405 Mini, is responsible for low-level flight stabilization using the inertial information. The generated values from the high-level controller are used to generate the necessary corrections to the motor speed. This hierarchical architecture is used to improve the modularity, efficiency, and safe flight operations of the system. The entire system operates within a continuous feedback loop, where the visual information is used to improve the control commands.

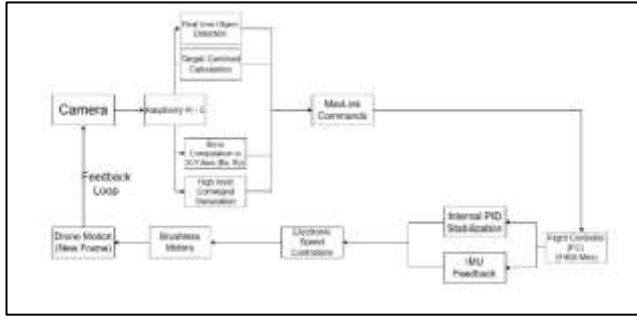


Fig 1: Block Diagram of the Drone

III. METHODOLOGY

The proposed methodology integrates real-time object detection with image-based guidance to generate motion commands for autonomous human tracking. The overall process consists of image acquisition, target localization, error computation, command generation, and flight actuation.

A. Image Acquisition and Object Detection

The video frames are continuously captured by the monocular camera installed onboard, which sends the video frames to the Raspberry Pi 5 for processing. The video frames are resized and preprocessed according to the input requirements for the YOLOv5 model.

The YOLOv5 model performs detection on each video frame, providing the coordinates of the bounding box corresponding to the detected human targets.

The bounding box is specified by:

$$(x_{min}, y_{min}, x_{max}, y_{max})$$

where:

- x_{min}, y_{min} represent the top-left corner,
- x_{max}, y_{max} represent the bottom-right corner.

The detection with the highest confidence score corresponding to the human class is selected as the target for tracking.

B. Target Centroid Acquisition

The centroid of the detected bounding box is computed to determine the target position in the image plane:

$$C_x = \frac{x_{min} + x_{max}}{2}$$

$$C_y = \frac{y_{min} + y_{max}}{2}$$

Let the image resolution be defined as $W \times H$. The center of the image frame is given by:

$$F_x = \frac{W}{2}, F_y = \frac{H}{2}$$

C. Image-Space

Error

Formulation

To maintain the human target at the center of the camera frame, horizontal and vertical errors are computed as:

$$e_x = C_x - F_x$$

$$e_y = C_y - F_y$$

Where:

e_x represents lateral deviation,

e_y represents vertical deviation.

These error values define the directional correction required for target alignment.

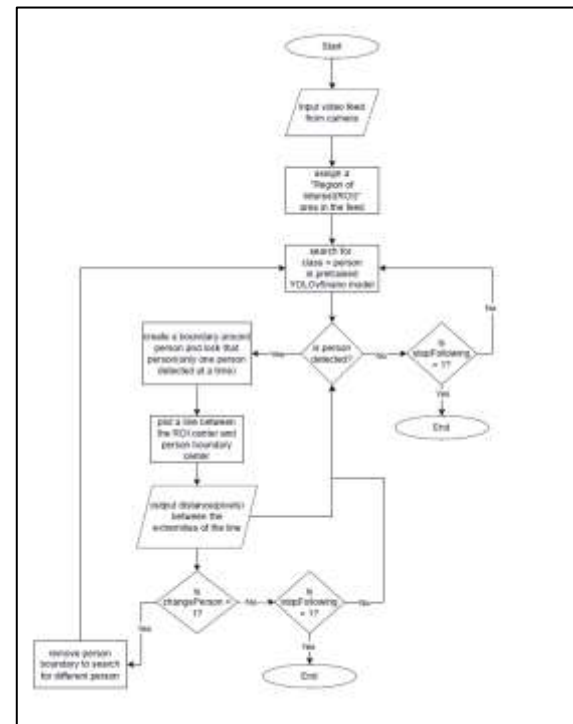


Fig 2: Image Acquisition Flowchart

D. High-Level Command Generation

The computed error values are mapped to motion commands. The sign and magnitude of e_x and e_y determine the required directional adjustment:

- If $e_x > 0$: target is right of center $\rightarrow$ drone moves right
- If $e_x < 0$: target is left of center $\rightarrow$ drone moves left
- If $e_y > 0$: target is below center $\rightarrow$ drone adjusts downward
- If $e_y < 0$: target is above center $\rightarrow$ drone adjusts upward

To avoid abrupt movements, the error values are normalized with respect to image dimensions:

$$e_x^{norm} = \frac{e_x}{W}$$

$$e_y^{norm} = \frac{e_y}{H}$$

These normalized errors are converted into high-level motion commands such as velocity or attitude setpoints before transmission to the flight controller.

E. MavLink Based Communication

These high-level motion commands are communicated from the Raspberry Pi to the flight controller through the MAVLink protocol. MAVLink is used for lightweight communication between the companion computer and the flight controller in a timely fashion.

These commands are embedded in MAVLink messages and are sent at a specific control frequency.

F. Closed-Loop Execution

The entire process operates in a continuous loop:

1. Capture frame
2. Perform object detection
3. Compute centroid
4. Calculate error
5. Generate motion command
6. Transmit command via MAVLink
7. Flight controller stabilizes and actuates motors
8. Updated frame captured

This closed-loop process enables dynamic target tracking in real time.

IV. EXPERIMENTAL SETUP

The experimental setup was designed to evaluate the performance of the proposed vision-based human-tracking quadrotor system in both simulation and hardware-integrated environments. The setup consists of onboard embedded hardware, flight control hardware, and simulation validation tools.

A. Hardware Configuration

The high-level vision processing was implemented using a Raspberry Pi 5 single-board computer with an onboard RGB monocular camera. The Raspberry Pi acts as a companion

computer used for executing object detection and command generation algorithms.

The quadrotor platform used a MicoAir F405 Mini flight controller for low-level quadrotor stabilization. It is a flight controller with an inertial measurement unit for attitude estimation. It provides onboard PID stabilization. It uses brushless DC motors with Electronic Speed Controllers.

The MAVLink protocol with a serial interface was used for communication between the Raspberry Pi 5 and flight controller.



Fig 3. CAD Model with Raspberry Pi 5

B. Software Environment

The object detection algorithm was developed using YOLOv5, which was run on a Raspberry Pi with a Linux-based operating system. The image acquisition and image processing steps were carried out using the OpenCV library.

The MAVLink communication protocol was achieved using an MAVLink API, which allowed the high-level motion commands to be sent from the computer to the flight controller.

C. Simulation and Testing Environment

Before actual deployment, the proposed system has been tested using a Betaflight-based FPV simulation environment. This simulation environment provides an actual scenario to analyze the flight dynamics and control behavior of the proposed system using the flight controller firmware used by the actual system.

The Betaflight environment is similar to using hardware-in-the-loop testing, where the flight controller is allowed to function with its native PID stabilization while still allowing it to receive command inputs from outside. This is similar to actual-world conditions, where the flight controller is allowed to function with MAVLink communication while still executing commands.

The simulation environment has been used to test the actual behavior of the proposed system, especially with regard to human tracking, without risking actual damage to the system. This allowed for the actual refinement of the vision-based guidance algorithm prior to actual deployment on the actual system.

D. Operational Parameters

The vision processing loop operated at a real-time frame rate, determined by the computation limit of the Raspberry Pi. The high-level motion commands were sent at a predetermined control rate to ensure continuous guidance.

All experiments were conducted under stable illumination to ensure reliable object detection performance.

V. RESULTS AND ANALYSIS

A. Detection and Localization Performance

The performance of the vision-based detection module was evaluated by using real-time video feed. Figure 3 illustrates a sample frame of the video feed where the human target has been successfully detected by the YOLOv5 model.

The detected human target is enclosed within a bounding box, and the centroid is calculated for tracking purposes. A line representing the frame center of the image is also shown, thus facilitating the visualization of the deviation of the detected target with respect to the intended point of alignment.

The system has been successfully tested for indoor conditions, and the human target has been consistently detected. This validates the fact that the detection module is accurately providing spatial information required for the control and tracking process.

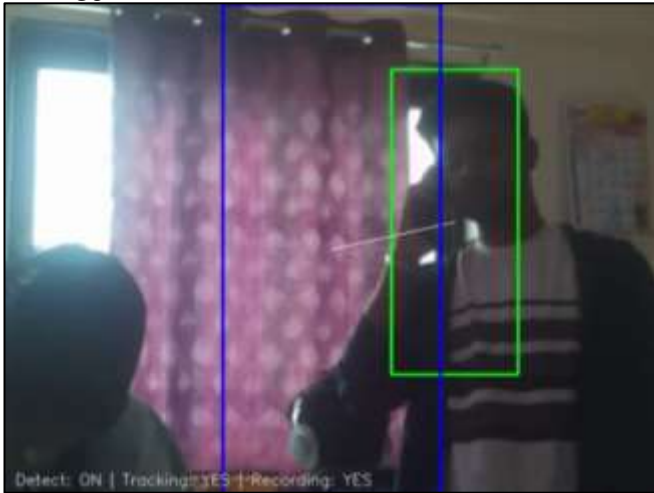


Fig 4. Real-time human detection and tracking

B. Control Response Analysis

In order to verify the dynamic performance of the control system, simulation tests were executed with sinusoidal and step input signals. Figure 5 illustrates the performance of the control system for these input conditions.

In the case of the sinusoidal input test case, the desired path is indicated by the reference input signal. The actual output of the control system follows the desired path with minor phase delay. From this test case, it is verified that the control system is capable of tracking continuously changing input conditions with satisfactory accuracy.

Considering the step input test case, the control system is capable of quickly moving towards the desired target with minor overshoot. The settling behavior of the control system verifies the stability maintained within the flight controller for handling sudden changes.

From these test case results, it is verified once again that the proposed control system is capable of handling dynamic and sudden changes in target positions.

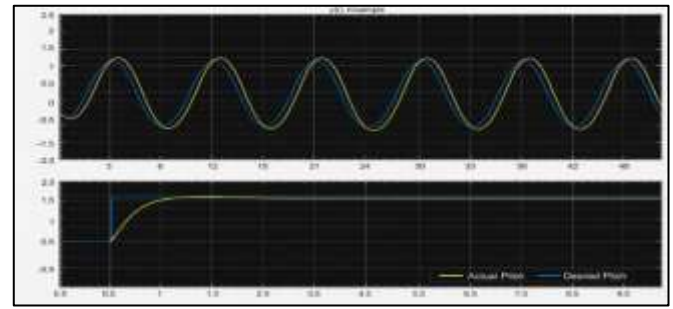


Fig 5. Simulation results showing system response to sinusoidal and step input signals.

C. System Performance Discussion

The combined results of detection and control response analysis prove the effectiveness of the system in fusing vision-based perception and flight control. The detection system produces a reliable target position, and the control system maintains stable flight of the quadrotor.

Although some delays and phase changes are observed in dynamic scenarios, stability in the system is maintained. This response is appropriate in a real-time tracking scenario where stability in flight is considered more important than alignment.

VI. CONCLUSIONS

This article outlines the design and implementation of a vision-based autonomous human tracking quadrotor using an embedded computing platform. The proposed system combines real-time object detection and image-based error computation to provide high-level motion commands to track the target. The Raspberry Pi 5 acted as a companion computer to process computer vision algorithms, and the MicoAir F405 Mini flight controller provided low-level stabilization using internal PID controllers.

To execute this design, the system architecture was designed to separate the perception and control aspects of the system to enable efficient execution of tasks on resource-constrained platforms. The MAVLink protocol provided efficient communication between the onboard computer and flight controllers to execute tasks seamlessly.

To verify the effectiveness of the proposed system, simulations using a Betaflight-based FPV simulation environment showed that the proposed system is capable of sustaining continuous tracking of the target object, which in this case is a human. The results of this experiment show that the proposed system is capable of executing tasks in real-time to detect and respond to the target in dynamic tracking scenarios.

In conclusion, the proposed system provides a cost-effective and efficient solution to implement a vision-based tracking system using a UAV, which has numerous applications in various fields.

VII. FUTURE SCOPE

The developed autonomous drone system can be further improved and expanded to increase its intelligence, efficiency, and applicability in real-world scenarios. With advancements in embedded computing and artificial intelligence, several enhancements can be introduced to make the system more adaptive and reliable.

Future enhancements include:

- **Integration of Deep Learning Models:** Using advanced algorithms like YOLOv8 or MobileNet for improved target recognition in complex environments.
 - **GPS and IMU Fusion:** Enabling autonomous navigation, waypoint tracking, and outdoor mapping for large-area missions.
 - **Obstacle Detection and Avoidance:** Implementing ultrasonic or LiDAR sensors to enhance safety during autonomous operation.
 - **Swarm Coordination:** Extending the framework to multiple drones for collaborative applications such as search and rescue or terrain mapping.
 - **Cloud-Based Data Logging:** Storing flight and image data for analysis and performance optimization.
- These developments will make the drone more versatile, scalable, and capable of performing high-level autonomous operations across various domains.

VIII. REFERENCES

- [1] J. REDMON AND A. FARHADI, "YOLOV3: AN INCREMENTAL IMPROVEMENT," *ARXIV PREPRINT ARXIV:1804.02767*, 2018.
- [2] D. MELLINGER AND V. KUMAR, "MINIMUM SNAP TRAJECTORY GENERATION AND CONTROL FOR QUADROTORs," IN *PROC. IEEE INT. CONF. ROBOTICS AND AUTOMATION (ICRA)*, 2011, pp. 2520–2525.
- [3] M. J. MAHMOOD, S. A. WANI, AND H. N. SHEIKH, "DESIGN AND IMPLEMENTATION OF A PID CONTROLLER FOR QUADCOPTER TRAJECTORY TRACKING," IN *PROC. IEEE INT. CONF. INTELLIGENT COMPUTING, INSTRUMENTATION AND CONTROL TECHNOLOGIES*, 2019, pp. 1491–1496.
- [4] P. CORKE, R. MAHONY, AND V. KUMAR, "MULTIROTOR AERIAL VEHICLES: MODELING, ESTIMATION, AND CONTROL OF QUADROTOR," *IEEE ROBOTICS & AUTOMATION MAGAZINE*, VOL. 19, NO. 3, PP. 20–32, 2012.
- [5] S. BOUABDALLAH, A. NOTH, AND R. SIEGWART, "PID VS LQ CONTROL TECHNIQUES APPLIED TO AN INDOOR MICRO QUADROTOR," IN *PROC. IEEE/RSJ INT. CONF. INTELLIGENT ROBOTS AND SYSTEMS (IROS)*, 2004, pp. 2451–2456.
- [6] T. BRESCIANI, "MODELLING, IDENTIFICATION AND CONTROL OF A QUADROTOR HELICOPTER," M.S. THESIS, LUND UNIVERSITY, 2008.
- [7] A. KAMATH AND P. THOMAS, "VISION-BASED AUTONOMOUS HUMAN TRACKING AND FOLLOWING ROBOT USING RASPBERRY PI," IN *PROC. IEEE INT. CONF. COMMUNICATION AND SIGNAL PROCESSING*, 2018, pp. 1337–1341.
- [8] M. MUELLER, M. HEHN, AND R. D'ANDREA, "A COMPUTATIONALLY EFFICIENT MOTION PRIMITIVE FOR QUADROPTER TRAJECTORY GENERATION," *IEEE TRANSACTIONS ON ROBOTICS*, VOL. 31, NO. 6, PP. 1294–1310, 2015.
- [9] S. SHAH ET AL., "AIRSIM: HIGH-FIDELITY VISUAL AND PHYSICAL SIMULATION FOR AUTONOMOUS VEHICLES," IN *FIELD AND SERVICE ROBOTICS*, 2018.
- [10] N. KOENIG AND A. HOWARD, "DESIGN AND USE PARADIGMS FOR GAZEBO, AN OPEN-SOURCE MULTI-ROBOT SIMULATOR," IN *PROC. IEEE/RSJ IROS*, 2004, pp. 2149–2154.
- [11] R. MAHONY, V. KUMAR, AND P. CORKE, "MULTIROTOR AERIAL VEHICLES: MODELING AND CONTROL," *IEEE ROBOTICS & AUTOMATION MAGAZINE*, 2012.
- [12] A. BEWLEY ET AL., "SIMPLE ONLINE AND REALTIME TRACKING WITH A DEEP ASSOCIATION METRIC," IN *PROC. IEEE INT. CONF. IMAGE PROCESSING (ICIP)*, 2016. (DEEP SORT)
- [13] L. BERTINETTO ET AL., "FULLY-CONVOLUTIONAL SIAMESE NETWORKS FOR OBJECT TRACKING," IN *EUROPEAN CONF. COMPUTER VISION (ECCV)*, 2016.
- [14] F. KENDOU, "SURVEY OF ADVANCES IN GUIDANCE, NAVIGATION, AND CONTROL OF UNMANNED ROTORCRAFT SYSTEMS," *JOURNAL OF FIELD ROBOTICS*, VOL. 29, NO. 2, PP. 315–378, 2012.
- [15] S. LEUTENEGGER ET AL., "KEYFRAME-BASED VISUAL-INERTIAL ODOMETRY USING NONLINEAR OPTIMIZATION," *THE INTERNATIONAL JOURNAL OF ROBOTICS RESEARCH*, VOL. 34, NO. 3, PP. 314–334, 2015.
- [16] M. ACHELNIK ET AL., "VISUAL-INERTIAL SLAM FOR MAVs," IN *PROC. IEEE IROS*, 2012.
- [17] M. FAESSLER ET AL., "AUTONOMOUS, VISION-BASED FLIGHT AND LIVE DENSE 3D MAPPING WITH A QUADROTOR MICRO AERIAL VEHICLE," *JOURNAL OF FIELD ROBOTICS*, 2016.