ABSTRACT

Title of Dissertation: MULTI-DOMAIN HUMAN-ROBOT INTERFACES

Sydrak S. Abdi
Doctor of Philosophy, 2024

Dissertation Directed by: Professor Derek Paley
Department of Aerospace Engineering

As autonomous robots become more capable and integrated into daily society, it becomes crucial to consider how a user will interact with them, how a robot will perceive a user, and how a robot will comprehend a user's intentions. This challenge increases in difficulty when the user is required to interact with and control multiple robots simultaneously.

Human intervention is often required during autonomous operations, particularly in scenarios that involve complex decision-making or where safety concerns arise. Thus, the methods by which users interact with multi-agent systems is an important area of research. These interactions should be intuitive, efficient, and effective all while preserving the operator's safety. We present a novel human swarm interface (HSI) that utilizes gesture control and haptic feedback to interact with and control a swarm of quadrotors in a confined space. This human swarm interface prioritizes operator safety while reducing cognitive load during control of an aerial swarm.

Human-robot interfaces (HRIs) are mechanisms designed to facilitate communication between humans and robots, enhancing the user's ability to command and collaborate with robots

in an intuitive and user-friendly manner. One challenge is providing mobile robotic systems with the capability to localize and interact with a user in their environment. Localization involves estimating the pose (position and orientation) of the user relative to the robot, which is essential for tasks that require close interactions or navigation in shared spaces. We present a novel method for obtaining user pose as well as other anthropometric measurements useful for human-robot interactions.

Another challenge is extending these HRI and HSI paradigms to the outdoors. Unlike controlled laboratory conditions, outdoor environments involve a variety of variables such as fluctuating weather conditions as well as a mix of static and dynamic obstacles. In this dissertation, we design a portable human swarm interface that allows an operator to interact with and control a multi-agent system outdoors. The portable HSI takes the form of smart binoculars. The user uses the smart binoculars to select an outdoor location and assign a task for the multi-agent system to complete given the targeted area. This system allows for new methods of multi-agent operation, that will leverage a user's on-the-ground knowledge while utilizing autonomous vehicles for line-of-sight operations, without compromising their situational awareness.

Multi-Domain Human-Robot Interfaces

by

Sydrak Solomon Abdi

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2024

Advisory Committee:
Professor Derek A. Paley, Chair/Advisor
Professor Michael Otte
Professor Joseph Conroy
Professor Dinesh Manocha
Professor Mumu Xu

Dedication

To my family and friends who help stabilize this chaotic system.

Acknowledgments

I would like to thank my advisor, Dr. Derek Paley, for his guidance, mentorship, and support throughout my Ph.D. at the University of Maryland. Your expertise in dynamics and controls has shaped my understanding and fueled my passion for robotics. It has been an honor and a pleasure to grow as a researcher under your guidance and learn from such an extraordinary mentor.

My colleagues in the Collective Dynamics and Controls Laboratory (CDCL) have truly enriched my graduate experience. From working on problem sets and exchanging ideas at our weekly meetings to participating in competitions and conducting field tests, your presence and support will be missed. In particular, I would like to thank Wei Cui, Animesh Shastry, Srijal Poojari, Zach Bortoff, and Anthony Thompson for their advice and support throughout my research.

I am also grateful to Dr. Otte, Dr. Conroy, Dr. Minocha, and Dr. Xu for agreeing to serve on my committee and providing valuable feedback for the work presented in this dissertation. Special thanks go to Dr. Paley, Dr. Otte, and Dr. Conroy, whose expertise in control theory, path planning, and autonomous robotics has shaped me into the researcher I am today. Their guidance ignited a passion that I intend to carry forward in my future endeavors.

The research presented in this dissertation would not have been possible without the support and mentorship from Dr. Adam Fineberg and Dr. Darnell Moore at Amazon Lab126, Dr. Joseph Conroy and Nikolas Vale from the Army Research Laboratory (ARL), and the Maryland Robotics

Center (MRC).

I would like to express my deepest gratitude to my family and friends, without whom this achievement would not have been possible. To my parents, Solomon and Shilmat, your love and guidance have shaped me into the person I am today, and I will forever be grateful. To my brothers, Myshak and Abednego, your presence and encouragement have been a constant source of strength and comfort. To my friends, particularly Val, Nigel, Evan, Corbyn, Kasey, Chris, and James. From practicing presentations and seeking advice to playing video games or just helping me unwind, your friendship and support have meant the world to me. I could not imagine achieving what I have without you.

Lastly, I want to thank Sara Wengrowski. In the fall of 2020, amid a pandemic, I chose to leave my job and pursue a Ph.D. She offered her unwavering support, staying up late to work on problem sets, helping me study, commiserating during tough times, and celebrating my victories. She has been my greatest advocate, and I am here today because of her love and support.

Table of Contents

List of Tables

List of Figures

Chapter 1: Introduction

1.1 Motivation

As autonomous robots become more capable and integrated into daily society, it becomes crucial to consider how a user will interact with them, how a robot will perceive a user, and how a robot will comprehend a user's intentions. This challenge increases in difficulty when the user is required to interact with and control multiple robots simultaneously.

Command and control of an aerial swarm is a complex task. This task increases in difficulty when the flight volume is restricted, and the swarm and operator inhabit the same workspace. While autonomous systems are becoming more capable, often human intervention is still required so the methods by which users interact with these multi-agent systems is an important area of research. We present a novel human swarm interface (HSI) that utilizes gesture control and haptic feedback to interact with and control a swarm of quadrotors in a confined space. This human swarm interface prioritizes operator safety while reducing cognitive load during control of an aerial swarm. While the proposed work demonstrates that an operator can safely and intuitively control a swarm of aerial robots in the same workspace, there is an underlying reliance on a motion capture system to provide the user's pose to the swarm. Next, we address the challenge of developing the capability for an autonomous system to estimate a user's pose to facilitate more intuitive interactions.

Human-robot interfaces (HRIs) are the mechanisms by which humans and robots interact and communicate [1]. The field of human-robot interactions explores these interfaces in an effort to optimize the utility of robots, while communicating the user's intentions intuitively. These interfaces can take the form of tablets [2], AR/VR headset [3], and even adaptive or assistive exoskeleton [4]. Regardless of the system, HRIs provide methods for users to interact with robotic systems in a variety of intuitive manners. One challenge is providing mobile robotic systems with the capability to localize and interact with a user in their environment. With the introduction of systems like Astro [5], developing intuitive HRIs that leverage known surroundings and user interfaces has never been more pressing. We present a novel method for obtaining user pose as well as other anthropometric measurements useful for human-robot interactions.

While the presented systems work well in laboratory settings, another challenge is extending these HRI and HSI paradigms to the outdoors. How will these systems deal with varying environmental conditions and unstructured environments? Current systems utilize ground control stations (GCS) which rely on users interacting with computers or tablets, removing them from their current environments, and reducing their situational awareness during every interaction. These interactions often involve drop down menus and selecting desired locations on a map which can be both difficult and time-consuming under strenuous circumstances. In this dissertation, we design a portable human swarm interface that allows an operator to interact with and control a multi-agent system outdoors. The portable HSI takes the form of smart binoculars. The user uses the smart binoculars to select an outdoor location and assign a task for the multi-agent system to complete given the targeted area. This system allows for new methods of multi-agent operation, that will leverage a user's on-the-ground knowledge while utilizing autonomous vehicles for line of sight operations, without compromising their situational awareness.

1.2 Relation to Prior Work

The work in this dissertation builds on both foundational and experimental work done by others. First, we discuss prior work investigating the use of gesture and motion controls for multi-agent system, the utilization of haptic feedback in relaying information to the user, as well as the utility of potential and barrier functions in preventing collisions between autonomous systems and objects in their environment. Second, we examine monocular pose estimation systems, current image-based methods for estimating anthropometric measurements, and state augmentation for bias estimation. Lastly, we review current ground control stations, head-mounted display-based human swarm interfaces, and mobile device-based multi-agent interfaces.

1.2.1 Human Swarm Interface

Swarm robotics is an emergent field. Applications in this field range from agriculture [6] and material transport [7], to search and rescue [8] and entertainment [9]. Regardless of the task at hand, the operator is responsible for making sure that the behavior of the swarm is in accordance with the given objectives. As the tasks become increasingly complicated and operators become more involved, it becomes especially important to consider factors that affect the interaction between the operator and the swarm. By reducing the cognitive load on the operator, they may be able to make more informed decisions, leading to more effective and efficient interactions.

The field of human-swarm interactions explores the interface between human operators and robotic swarms in an effort to optimize control over the swarm while reducing the cognitive load on an operator. A cobot, or collaborative robot, is a robot intended to interact with humans within a shared environment. We describe a novel cobot human swarm interface (HSI) that reduces

cognitive load on the operator by addressing one of the largest hurdles in collaborative robotics, agent-operator collision. The safety of the operator is prioritized and encoded into each agent through the use of distance-based potentials, whereas motion and gesture controls relay desired commands and control of an aerial swarm.

Gesture and motion controls are effective methods of relaying an operator's intent to robotic systems. Not only are gestures natural and intuitive means of communication, but machine-learning algorithms have bridged the gap allowing operators to use gestures to communicate with, command, and control robotic systems.

[10], [11], and [12] describe vision-based gesture control methods in which convolutional neural networks classify gestures made by the operator. Once classified, these gestures are translated into control signals and sent to their robotic systems. [13] and [14] extend this idea to multi-agent systems by enabling agents to classify gestures onboard and perform the assigned tasks defined by those gestures autonomously. One downfall to these methods is that they require a line of sight for the operator to receive the intended instructions.

Another method of relaying desired commands is through the use of muscle and motion sensing devices such as the Mbientlab IMU bracelet, Myo armband, or OYMotion gForce-Pro+. [15] presents a method for controlling the 3D position of a quadrotor by confining the motion of the quadrotor to a 3D surface. The position of the quadrotor is determined by finding the intersection between that surface and a pointing vector generated from the arm of the operator wearing an IMU. An external button is used to iterate between predefined surfaces to achieve the desired motion in space. [16] describes an interpreter that uses the motions and static gestures of an operator wearing a Myo armband to replace the functionalities of a computer mouse, allowing an operator to control the formation of a swarm by simply drawing the desired formation with

their arm. [17] shows that static gestures may be used to interact with a virtual menu, allowing an operator to have access to a library of desired controls through which they may guide a swarm of ground robots through an environment with obstacles. Similarly, [18] develops a clustering algorithm to perform online gesture recognition and showed that an operator wearing a Myo armband may successfully navigate a drone through an obstacle course containing hoops. Others such as [19, 20], and [21] have expanded these control paradigms to develop multi-modal interfaces that include speech as well as motion and gesture control for their multi-agents systems. Motion and gesture controls are used to select the desired agents, whereas speech control is used to directly relay the desired commands to those selected agents.

While being able to control a quadrotor is crucial, the information received about the quadrotor and its states can be just as important, allowing an operator to make more informed decisions. In indoor settings, the domain in which robots move may be limited, so operators naturally rely on visual feedback as their main source of information pertaining to a robot's states. While this is generally sufficient, it becomes increasingly more difficult to estimate these parameters in multi-agent systems or environments that include obstacles.

One method to relay pertinent information quickly is through the use of haptic devices. Intensity, duration, rhythm, and tactor locations are all parameters that can be varied to develop a library of haptic patterns to relay desired information to an operator. [22] and [23] utilize an Omega 3 active force feedback device as both a joystick and tool to provide haptic feedback. As the Omega teleoperates a swarm of aerial vehicles, resistive force feedback is provided to the operator during flight when the selected direction of travel is impeded by an obstacle, aiding the operator in navigating the swarm around obstacles in the environment. [24] presents a haptic glove paired with six tactors that corresponded to the axes of motion of a teleoperated quadrotor.

The tactors on the glove vibrate with varying intensities in proportion to the quadrotor's proximity to any obstacles, providing spatial awareness even when the quadrotor moved directly out of the operator's line of sight. Others have developed libraries of vibrotactile patterns that relayed changes in robotic system's states such as the density and center of mass of a swarm of aerial robots [25] or the attitude of a virtual aircraft [26]. While these works show the benefits of using muscle and motion control sensors as pipelines to provide robotic systems with their operator's intent or desired controls, they are limited to teleportation applications, that is, they lack the ability to allow the operator to directly and safely interact with these robotic systems.

Artificial potential fields and barrier functions have a rich history of being used in path planning and collision avoidance applications for autonomous systems. [27], [28], and [29] successfully developed a steering-based collision avoidance method for vehicles by virtually attaching a variety of velocity potential functions to objects detected in the surrounding environment. The aggregate of the surrounding velocity fields is used to safely steer the car around obstacles. [30] and [31] apply potential fields to the problems of multi-agent path planning through obstacle rich complex environments. [32], [33] and [34] utilize potential fields and barrier functions, respectively, to ensure inter-agent collision avoidance during completion of tasks assigned to the multi-agent systems. The swarm-velocity controller presented here leverages these works to develop a potential based approach that achieves online inter-agent safety as well as operator collision avoidance, while continuously attempting to maintain the prescribed formation.

1.2.2 Multi-Sensor Pose and Parameter Estimation

A human-robot interface (HRI) is the mechanism by which humans and robots interact and communicate [1]. The field of human-robot interactions explores these interfaces in an effort to optimize the utility of robots, while communicating the user's intentions intuitively. Visual- and gesture-based communication methodologies are intuitive between humans, but giving a robotic system the ability to perceive and comprehend the intentions of a human is more difficult. A key challenge in the field of human-robot interactions is the estimation of the user's pose, i.e., body position and orientation. We present a sensor-fusion framework that estimates the pose of a user as well as their torso length and chest width. As an example application of this system, the pose and estimated body lengths are utilized to estimate the location of an object on the floor that the user is pointing at.

Human pose recognition and estimation is a common task in computer vision and a key step in enabling safe and intuitive human-robot interfaces. Applications include autonomous vehicles [35], action recognition [36], augmented/virtual reality [37], and sports science [38]. [39] and [40] propose model fitting-based methods by finding the joint locations of the user in an image, generating a 3D candidate pose, and recursively minimizing the error between the joints in the image and the projection of the candidate pose's joints onto the image. Similarly, [41] begins with the user's joint locations on an image, but instead reconstructs their pose by assuming their 2D joint locations are scaled orthographic projections of their 3D pose and solves for the corresponding scaling factor. While earlier approaches relied on manual joint selection or fixed appendage lengths and ratios to resolve their depth ambiguities, current methods such as SMPLify [42] have designed networks to automate this model fitting process without these

constraints. Recently, several learning-based methods have shown success in estimating 3D pose [43], [44], and [45]. Others such as [46] and [47] have expanded on these works to include multi-person pose estimation frameworks. While these works have shown success in estimating 3D poses, they rarely consider the user's actual size and, thus, their solutions are typically incorrectly scaled to match the user's anthropomorphic measurements.

While being able to estimate the pose of the user is crucial, estimating the user's anthropomorphic measurements can be just as important. The capability to estimate body and appendage lengths can provide robots with a sense of scale for the individuals they're interacting with, as well as providing them with information that could be used to develop more intuitive human-robot interfaces. [48], [49], and [50] show that the anthropometric measurements of a user can be estimated by first initializing a body model at the user's estimated depth and then estimating the measurements based on the fitted model. One downfall for these methods is that they rely on calibrating their systems per experiment or using depth sensors in addition to 2D images.

State estimation in the presence of biases has been the subject of in-depth research. [51] and [52] show that optimal estimates of a linear system's states and unknown constant biases can be obtained by augmenting the system's state vector. More recently, [53] and [54] show that state augmentation techniques can also be used to estimate states and biases in nonlinear systems on mobile robotics platforms. The extended Kalman filter (EKF) presented in this work leverages these works by adding measurement biases as well as user parameter estimate biases to the state vector. The nominally chosen values of the user's body lengths are treated as constants with their deviation from the true value as biases. Thus, the true values of the user's body lengths can be estimated in conjunction with user pose.

1.2.3 Outdoor Multi-Agent Control and Target Selection

While human swarm interfaces (HSIs) and human-robot interfaces (HRIs) have been extensively researched for indoor laboratory environments, their application in outdoor settings remain largely unexplored. Traditionally, outdoor UAV control has been managed through Ground Control Stations (GCSs), which typically leverage flight control or mission planner software packages like QGroundControl [55] or Mission Planner [56]. The packages provide numerous capabilities including system monitoring, waypoint navigation, and mission planning. While these packages are typically used for single agent operations, they maintain some basic functionalities useful for multi-agent operations. Researchers have shown that these types of GCSs can be used for coordinated missions such as multi-agent path planning [57] and optimal area coverage [58]. More recently researchers like [59] and [60] have begun writing their own custom GCSs to provide them with real-time telemetry monitoring, and advanced algorithms for swarm coordination and task allocation.

One challenge is designing these systems to be portable and easily used by a variety of operators. With the rise in popularity of augmented and virtual reality systems, many researchers have begun to integrate these mobile computational units into their human-robot interfaces. [61] utilized an Oculus Quest to develop a mixed reality interface that allows a user to visualize remote environments and supervise field robots. [62] extended this work by developing a mixed-reality system that allowed multiple operators with different roles to interact with a swarm via a configurable multi-modal human interface. Their multimodal user interface supported swarm visualization, human gesture inputs, tactile feedback, and audio feedback. Experiments showed that this system was effective in visualizing remote environments as well as controlling the po-

sition, formation, and tasks assigned to a swarm. [63] developed an augmented reality human swarm interface that allowed a user to draw a 3D lasso using their hands and provide verbal commands to interact with the selected agent in the swarm. While these systems have shown success in providing additional capabilities to users for outdoor operations, they require the user to rely on head-mounted displays which currently lack sufficient battery longevity for extended outdoor operations.

While traditional GCS have shown success in their ability to provide users with the necessary capabilities to monitor and control robotic platforms, they are often constrained by their reliance on stationary or semi-portable computational hardware. Recently, portable systems like smartphones and tablets, in conjunction with software packages like the Android Team Awareness Kit (ATAK) [64], have provided users with the capability to use touchscreens for waypoint selection and manual control. [65] used ATAK to dispatch UAVs to locations of interest to search, identify, and visualize targets. [66] utilized ATAK to allow a multi-agent team to share their locations, the locations of any objects of interest, and coordinate air-ground missions between autonomous agents during a search and rescue mission. The ATAK has also shown promise in acting as a visualizing tool. [67] showed that an ATAK could display a heat map created by a distributed network of remote sensing aerial vehicles during live operation. [68] developed a phone-based swarm controller that allowed a user to control the position, orientation, and shape of a swarm's formation outdoors using a custom written app, and the orientation of the swarm controller. While these systems are portable, they require the user to concentrate on their devices, reducing their situational awareness.

1.3 Contributions of this Dissertation

The contributions of this dissertation are in the areas of multi-agent control, pose and parameter estimation, and human-robot interactions. These contributions advance our understanding of how users can effectively engage with robots in a variety of environments. Many of these results have either been published in peer-reviewed journals or submitted and are currently under review [69].

1.3.1 Cobot Human Swarm Interface

We develop a cobot human swarm interface that prioritizes operator safety through the use of distance-based potential functions and feedback control, and a gesture-based control methodology that provides an operator with control of a swarm's position, orientation, and density in either the global frame or the operator's body frame. Experimental results validate the utility of the designed swarm velocity controller in maintaining operator safety during control of a cobot swarm while occupying the same workspace, allowing the operator to focus their efforts on completing their tasks rather than their personal safety.

1.3.2 Multi-Sensor Pose and Parameter Estimation

Inspired by the introduction of consumer robots, we develop a monocular depth-estimation framework leveraging an existing keypoint detection package, a real-time pose tracking solution fusing a single camera and multiple acoustic sensors, and a method to assimilate visual and acoustic sensor data using an extended Kalman filter to estimate body dimensions. Motivated by previous work [69], this system develops the capability to estimate the pose and body lengths of a

user, providing robotic systems with a sense of scale for the individuals they're interacting with, as well as information about the user that may be used to develop more intuitive human-robot interfaces.

1.3.3 Outdoor Target Selection-based Human Swarm Interface

We develop a method for enabling users to control the position of a multi-robot system in outdoor environments. The smart binoculars, equipped with rangefinder capabilities, accurately obtain targeted coordinates and relay them to a swarm of UAVs, which then autonomously navigate to these locations to complete their assigned tasks. Experimental results show that this system can estimate the coordinates for a desired location outdoors. The captured coordinates are then relayed to a swarm of UAVs to display positional control of both individual and multiple agents simultaneously.

1.4 Outline of Dissertation

This dissertation is outlined as follows. Chapter 2 presents the mathematical and experimental preliminaries required to understand the presented work.

Chapter 3 introduces swarming formations and gesture recognition. We also derive the distance-based swarm velocity controller used to prevent inter-agent as well as operator collision avoidance and describe the haptic feedback and gesture control systems. Experimental results demonstrate the utility of the gesture-based control methodology in controlling the position, orientation, and density of the swarm, and validate the design of the swarm velocity controller.

Chapter 4 shows the derivation of the monocular pose estimation framework used to de-

velop the vision-based system's observation model as well as the sound-based system's estimate. We continue by defining the states, measurement, and observation models used by the extended Kalman filter to estimate the user's pose and body measurements.

Chapter 5 introduces the smart binoculars, a portable device designed to facilitate interactions between users and multi-agent autonomous systems in dynamic outdoor environments. We discuss the involved hardware, target localization associated functionalities, and reviews the evolution of the system. Experimental results demonstrate the utility of the smart binoculars as an outdoor human swarm interface.

The conclusion and summary of dissertation are discussed in Chapter 6.

Chapter 2: Background

This chapter provides background information for chapters 3 - 5. First, we discuss the EMG-based gesture recognition armband, aerial swarm of miniature quadrotors, and a haptic vest utilized by the human swarm interface presented in chapter 3. Second, we present the pinhole camera model in conjunction with the pose recognition and sound localization systems which are used to estimate a user's pose and anthropomorphic measurements. Third, we introduce the mathematical relationship used by the smart binoculars in chapter 5 to estimate target coordinates outdoors.

2.1 Experimental Hardware

2.1.1 Gesture Recognition

Electromyography (EMG) sensors measure and record the electrical signals generated in muscles during contraction [70]. The OYMotion gForcePro+ armband is a wearable EMG based gesture recognition device shown in Fig. 2.1. Containing an 8-channel EMG array and a 9-axis IMU, the gForcePro+ provides a real-time orientation estimation of the operators forearm as well as gesture recognition via Bluetooth BLE 4.2 up to a range of 10m. Gesture recognition is accomplished utilizing a trainable AI model onboard the armband that allows for up to 16 unique

user-defined gestures [71].

Figure 2.1: OYMotion gForcePro+

2.1.2 Indoor Aerial Swarm

The Bitcraze Loco Swarm [72] is an aerial robotic swarm consisting of homogeneous quadrotors called Crazyflies [73] utilizing the Loco Positioning system for localization. Crazyflies, shown in Fig. 2.2, are miniature quadrotors measuring 92mm $\times$ 92mm with a takeoff weight of 27g. The Loco Positioning system is an Ultra-Wide Band radio-based localization system used to find the 3D position of the Crazyflies in space [74]. Loco Positioning nodes [75] are positioned within a room. For this work, these nodes were used to define a flight volume for the swarm. Each Crazyflie is paired with a Loco Positioning deck [76]. High frequency radio messages are sent back and forth between the nodes and the decks, allowing the system to measure the distance between each node and the deck to calculate the position of the deck and therefore the Crazyflie. Position estimation is performed onboard the Crazyflie and sent to the ground station. The Loco Swarm was flown using Crazyswarm, a system architecture for controlling multiple Crazyflies

simultaneously [77].

Figure 2.2: Bitcraze Loco Swarm

2.1.3 Haptic Vest

The bHaptics TactSuit X40 is a virtual reality haptic vest, shown in Fig. 2.3. Weighing 1.7kg, this haptic vest contains 40 vibrotactile motors, 20 on both the front and back, and is connected to the base station using BLE 4.0. [78].

Figure 2.3: bHaptics TactSuit X40

2.2 Localization Algorithms

2.2.1 Pinhole Camera Model

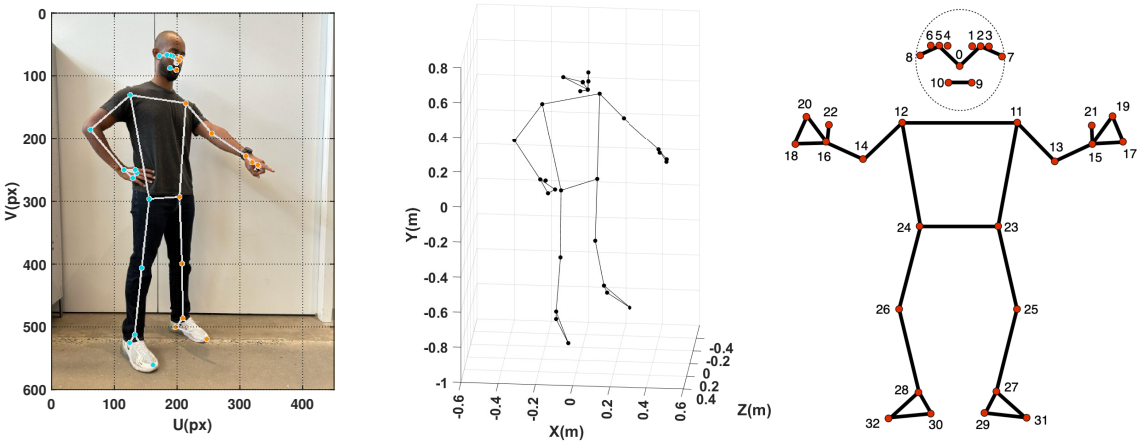
The pinhole camera model describes a mathematical relationship between a three-dimensional point and its projection onto a two-dimensional image plane. Given a camera's intrinsic properties, i.e., its focal lengths (f_x, f_y) and principal point (u_0, v_0), a point $\mathbf{P} = (X, Y, Z)_I$ can be projected onto an image plane $\mathbf{p} = (u, v)_P$ using the following relationship [79], [80], [81]:

$$u = f_x \left(\frac{X}{Z}\right) + u_0 \tag{2.1}$$

$$v = f_y \left(\frac{Y}{Z}\right) + v_0 \tag{2.2}$$

2.2.2 Pose Recognition

MediaPipe is an open-source framework for building perception pipelines [82]. MediaPipe Pose Landmarker is one of the many preconstructed solutions that utilize this framework. The MediaPipe Pose Landmarker detects, identifies, and tracks body landmarks in video feed. The system returns the estimated real-time location of the landmarks in the image frame, as well as the 3D world coordinates for a scaled skeletal model in the camera frame as shown in Fig. 2.4a. Fig. 2.4c shows a skeletal model of the 33 landmarks representing the approximate locations of the corresponding body parts [83], [84].

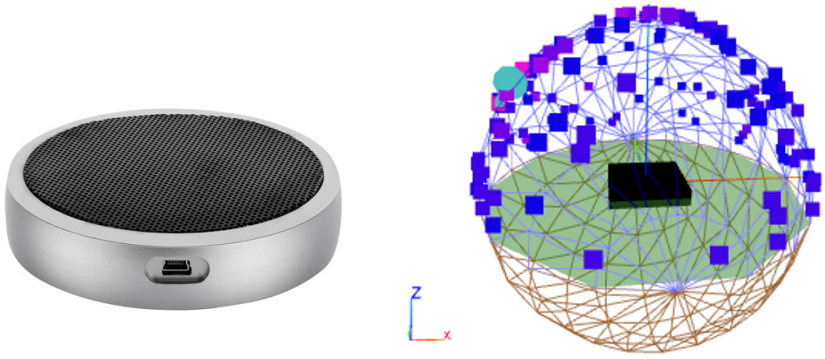


(a) (b) (c)

Figure 2.4: (a) Image annotated with landmarks from MediaPipe pose landmarker; (b) 3D world coordinates from MeidaPipe's estimated skeletal model; (c) skeletal model definition showing 33 body landmarks

2.2.3 Sound Source Localization

Direction of arrival calculations were performed on UMA-8 V2s, high-performance, low-cost multichannel USB microphone arrays with seven microphones configured in a circular arrangement [85]. These microphone arrays were utilized in conjunction with Open embeddeD Audition System (ODAS), a library dedicated to performing sound source localization, tracking, separation, and post-filtering [86]. Each microphone array provides the system with a 3D unit vector in the direction of the detected sound source.



(a) (b)

Figure 2.5: (a) UMA-8 USB mic array - V2.0, (b) Graphical representation of the auditory detections (dark blue squares) and a three-dimensional point (light blue circle) representing the unit vector in the direction of a tracked source obtained from using ODAS.

2.2.4 Terminal Geographic Location Given Range and Bearing

Given an initial location (lat_0, lon_0), range ρ, bearing β, and the radius of the earth R, the coordinates for a terminal location can be found as shown below:

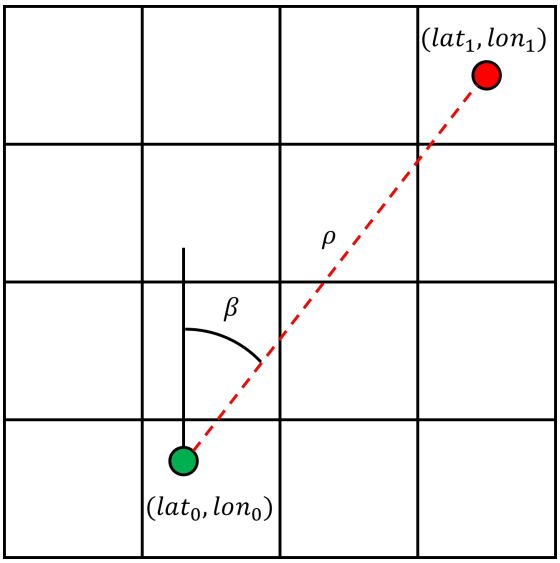


Figure 2.6: Diagram showing terminal geographic coordinates (lat_1, lon_1), given an initial position (lat_0, lon_0), range (ρ), and bearing (β)

$$\mathrm{lat}_1 = \sin^{-1}\left(\sin(\mathrm{lat}_0)\cos\left(\frac{\rho}{\mathrm{R}}\right) + \cos\left(\mathrm{lat}_0\right)\sin\left(\frac{\rho}{\mathrm{R}}\right)\cos\left(\beta\right)\right) \tag{2.3}$$

$$\mathrm{lon}_1 = \mathrm{lon}_0 + \mathrm{atan2}\left(\mathrm{a}, \mathrm{b}\right) \tag{2.4}$$

where

$$a = \sin\left(\beta\right)\sin\left(\frac{\rho}{R}\right)\cos\left(\mathrm{lat}_0\right)$$

$$b = \cos\left(\frac{\rho}{R}\right) - \sin\left(\mathrm{lat}_0\right)\sin\left(\mathrm{lat}_1\right)$$

Chapter 3: Safe Operations of an Aerial Swarm via a Cobot Human Swarm Interface

3.1 Introduction

This chapter investigates the development of a human swarm interface that facilitates the command and control of an aerial swarm while the user and swarm occupy the same workspace. We develop a gesture-based control methodology to provide an operator with control of a swarm's position, orientation, and density in either the global frame or the operator's body frame. This HSI was designed to prioritize operator safety through the use of distance-based potential functions and feedback control. This work demonstrates that an operator can safely and intuitively control a swarm of aerial robots within the same workspace.

This chapter is organized as follows. Section 3.2 introduces the control strategies employed in the developed human swarm interface. We define the swarming formations used to organize the Loco Swarm and review the trained gestures used to control the formation. This section also describes our potential-based swarm velocity controller and explains how it, in conjunction with haptic feedback, prevents collisions between the user and aerial agents. Section 3.3 reports the experimental results. The conclusions and future work are discussed in Section 3.4.

3.2 Control Strategy

In the control strategy described here, the operator utilizes gestures to command and control the position, orientation, and density of the aerial swarm. This is accomplished by dynamically modifying the formation defining the swarm, while each agent autonomously follows their assigned goal positions within the formation.

3.2.1 Swarming Formations

The swarming formations used in this work are shown in Fig. 3.1. All formations are radially symmetric, which is a property that is leveraged when controlling the density of the swarm. While the desired formation is determined by the number of desired agents and a desired radius, no agent is assigned a specific location within the formation. During assembly, the assignment problem is solved using a Munkres assignment algorithm [87], minimizing the distance each agent must travel to complete the formation.

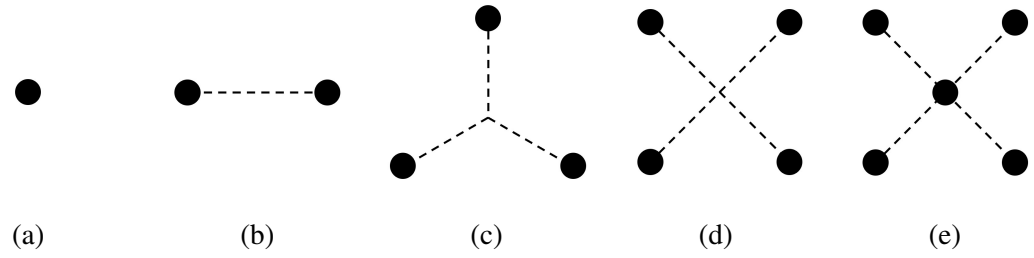


Figure 3.1: Swarming formations per number of agents, ranging from (a-e) 1-5 agents respectively.

3.2.2 Gesture Recognition

One of the objectives of this work is to provide an operator with the capability to control the position, orientation, and density of an aerial swarm in both the global frame as well as the operator's body frame. This is achieved using the trained gestures shown in Fig. 3.2.

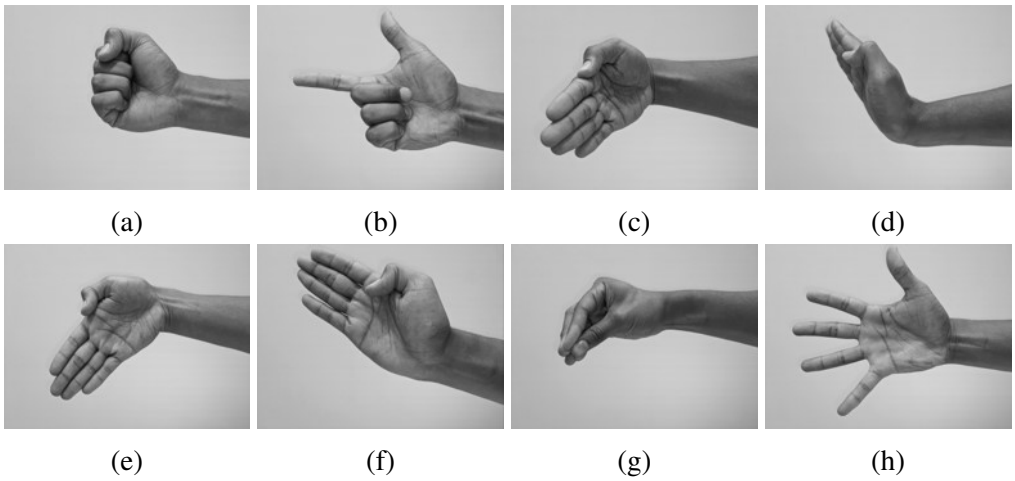


Figure 3.2: 8 custom gestures recognized by the gForcePro+ AI model after training: (a) closed fist, (b) finger pointing, (c) wrist flexion, (d) wrist extension, (e) ulnar deviation, (f) radial deviation, (g) finger pinch, and (h) finger spread.

Inspired by the work presented in [88] and [89], the selected gestures have been shown to be successfully recognized with a high degree of accuracy using a number of methods, including those employed on the OYMotion gForcePro+. The *closed fist* gesture is assigned the role of commanding the agents to takeoff from their respective locations, as well as assembling the agents into formation. The *finger pointing* gesture lands the aerial swarm. The *wrist flexion* and *wrist extension* gestures give the operator positional control over the swarm. The positional control shown in this work is limited to translation along the X-axis of the inertial frame. This could be extended with 2 additional pairs of gestures to control the translation in the remaining two axes. The *ulnar deviation* and *radial deviation* gestures provide the operator with control of

the orientation of the swarm by rotating the swarm counterclockwise and clockwise in the X-Y plane respectively. The *finger pinch* and *finger spread* gestures allow the operator to decrease and increase the density of the swarm respectively.

Gestures were selected as the control modality for this work due to the decrease in the amount of required infrastructure in comparison to tablets or hardware-based control systems, and their lack of ambiguity in comparison to speech-based control systems. The goal of this work is to develop an HSI that prioritizes operator safety, while reducing cognitive load during control of a cobot swarm. One method to decrease the cognitive load on the operator is to decrease the amount of infrastructure the operator is required to engage with.

3.2.3 Swarm Velocity Controller

Given a swarm of n identical agents, a velocity control algorithm is developed to safely navigate an agent from its current locations x_i, to their goal locations x_g, while circumventing the p obstacles in the environment. The attractive potential $U_{i_{attr}}$ and velocity controller $V_{i_{attr}}$ for agent i can thus be expressed as

$$U_{i_{attr}}(x_i) = V_0||x_i - x_{i_g}|| \tag{3.1}$$

and

$$
\begin{aligned}
V_{i_{attr}}(x_i) &= -\nabla U_{i_{attr}}(x_i) \\
&= -\nabla V_0 ||x_i - x_{i_g}|| \\
&= -V_0 \frac{(x_i - x_{i_g})}{||x_i - x_{i_g}||}
\end{aligned}
\tag{3.2}
$$

where V_o is the desired constant velocity.

The obstacle-avoidance velocity controller was designed to allow agents to safely avoid obstacles while moving towards their goal. The FIRAS function proposed by Khatib [90] is frequently used as a repulsive potential function:

$$
U_{obs}(x) = \begin{cases} \frac{1}{2}\eta \left(\frac{1}{\rho} - \frac{1}{\rho_0}\right)^2, & \rho \leq \rho_0 \\ 0, & \rho > \rho_0, \end{cases}
\tag{3.3}
$$

where ρ_0 represents the limit distance, or radius of influence, of the repulsive field and ρ represents the shortest distance to the obstacle. This function has been adapted for these applications. The obstacle-avoidance velocity controller $V_{i_{obs}}$ for agent i can thus be expressed as

$$
\begin{aligned}
V_{i_{obs}}(x_i) &= -\sum_{m=1}^{p} \nabla U_{i_{obs}}(x_i) \\
&= \begin{cases} \sum_{m=1}^{p} \eta \left(\frac{1}{\rho_m} - \frac{1}{\rho_0}\right) \frac{1}{\rho_m^2} \frac{\partial \rho_m}{\partial x}, & \rho_m \leq \rho_0 \\ 0, & \rho > \rho_0, \end{cases}
\end{aligned}
\tag{3.4}
$$

where $\rho_m = ||x_i - x_{obs_m}||$ is the distance to obstacle m.

3.2.4 Haptic Feedback

The bHaptics TactSuit X40 is a virtual reality haptic vest shown in Fig. 2.3. In this work, these motors provide the operator with the continuous location of the center of mass of the swarm with respect to the flight volume. The flight volume was divided into 40 bins, paralleling the motors on the vest. The centroid of the swarm is calculated and localized to 1 of these 40 bins. While the swarm's center of mass is in a given bin, the corresponding tactor vibrates on the vest; as the center of mass moves, the analogous motors on the vest vibrate, providing the operator with the continuous spatial awareness of the swarm's location relative to the flight volume and its boundaries in real time. An example of this feedback can be seen in Fig. 3.3.

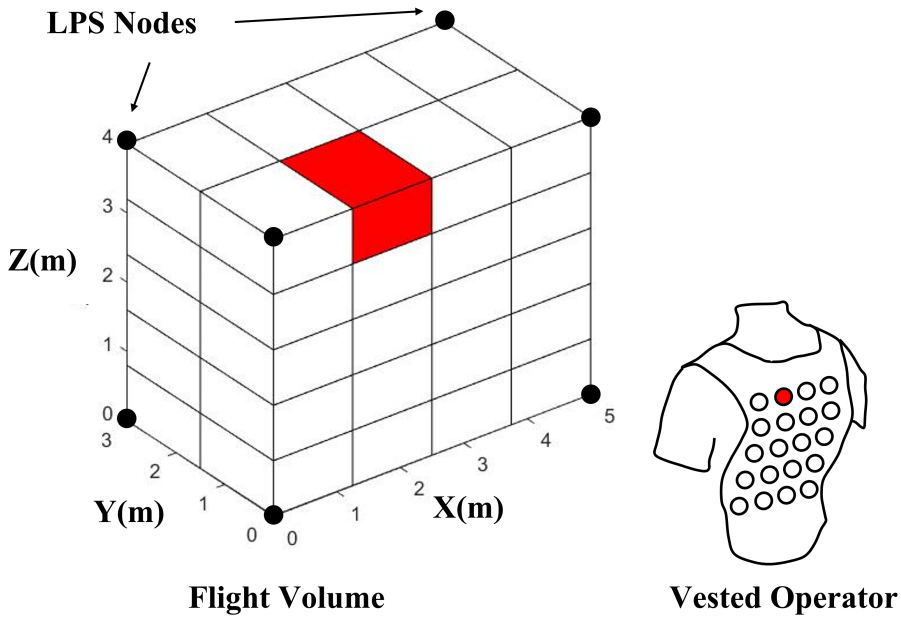


Figure 3.3: Experimental setup showing the flight volume defined by the Loco Position system (LPS) Nodes, the bin partitioning utilized to localize the swarm's centroid, and the vested operator receiving haptic feedback regarding the location of the centroid of the swarm.

3.3 Experimental Results

Experiments are conducted to demonstrate the gesture-based control methodology in controlling the position, orientation, and density of the swarm in both the inertial frame as well as the operator's body frame and validate the utility of the designed swarm velocity controller in maintaining operator safety during control of a cobot swarm while occupying the same workspace. All gesture training and experiments were conducted by the authors.

3.3.1 Experimental Setup

The system shown in these results is a Loco Swarm using Crazyswarm's *goTo()* functionality in which waypoints are sent to each agent at 4Hz and an onboard controller plans a smooth trajectory from the current state to the waypoint position. The waypoints, $x_{i_{k+1}}$, are derived as shown below:

$$x_{i_{k+1}} = \begin{cases} x_{i_k} + V_{i_{attr}}(x_{i_k})\Delta t + V_{i_{obs}}(x_{i_k})\Delta t, & ||x_{i_k} - x_{i_g}|| > V_{i_{attr}}(x_{i_k})\Delta t \\ x_{i_g} + V_{i_{obs}}(x_{i_k})\Delta t, & ||x_{i_k} - x_{i_g}|| \leq V_{i_{attr}}(x_{i_k})\Delta t \end{cases} \tag{3.5}$$

where x_{i_k} is the current position of agent i, $V_{i_{attr}}(x_{i_k})$ are the goal-bound velocities, $V_{i_{obs}}(x_{i_k})$ are the collision avoidance velocities, and Δt is the inverse of the desired waypoint frequency. All positions derived using this methodology are 2-D positions spanning the X-Y plane, with a fixed altitude of 1m in an effort to avoid the effects of the downwash interaction between the aerial agents in the swarm. The prescribed formation has a radius of 0.75m. To ensure inter-agent collision avoidance, the agents in the swarm consider their counterparts obstacles. Fig. 3.3

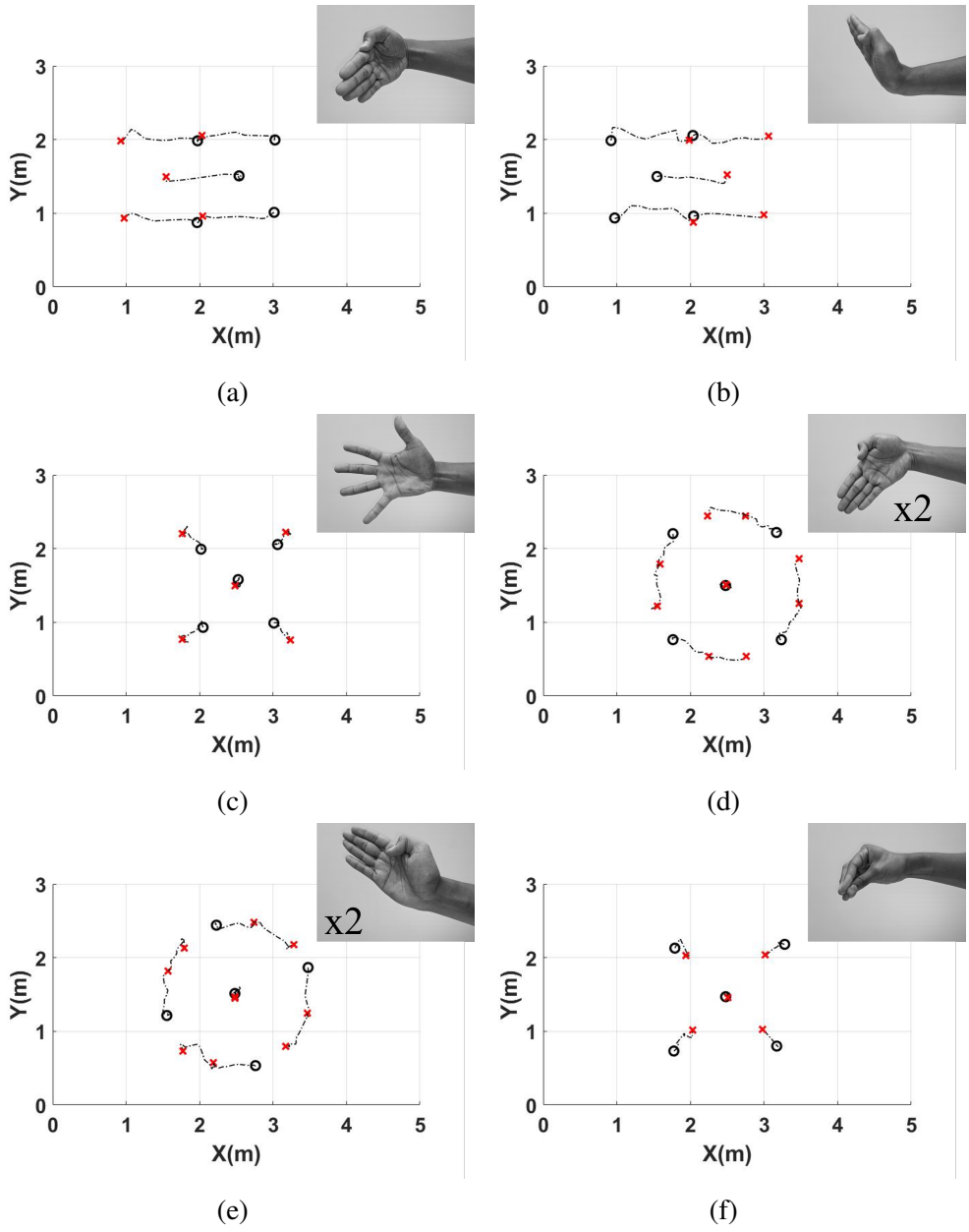
shows the experimental setup. For video results refer to supplemental materials shown here: `https://youtu.be/9kladblLRj8`

3.3.2 Position Control in Inertial Frame

In the first experiment, the ability to control the position, orientation, and density of a robotic swarm containing 5 agents through the developed HSI is illustrated. The positional control over the swarm using the *wrist flexion* and *wrist extension* gestures was mapped to a translation of 1m. The *ulnar deviation* and *radial deviation* gestures were mapped to rotations of $22.5°$. The *finger pinch* and *finger spread* gestures were mapped to radial translations of 0.25m about the centroid of the Loco Swarm's formation. The operator remained stationary throughout this experiment. Fig. 3.4 shows a time series of the Loco Swarm's trajectory, as seen from above, during a gesture-based flight demonstration. As shown, an operator can control the position, orientation, and density of a swarm in the inertial frame.

3.3.3 Position Control in Operator Frame

The second experiment continued to utilize gesture control. A VICON Vantage V8 system with 12 cameras was used to localize the operator within the environment. During this experiment, the operator was free to move throughout the environment as they desired. A robotic swarm containing 3 agents was localized using the Loco Positioning system. The OYMotion gForcePro+ provides the orientation of operator's arm. Once the pose of the operator's arm is established, a pointing ray from the operator's arm is generated, and the intersection between that ray and the floor may be calculated. The operator then utilized the previous translational control



(a) (b) (c) (d) (e) (f)

Figure 3.4: Time series (a-f) of the Loco Swarm's X-Y trajectory during a gesture based flight demonstration. The locations denoted with the ∘ and × symbols represent the initial and final positions of the swarm respectively, before and after each gesture was performed.

gestures to relay the intersection point to the Loco Swarm as the desired location for the centroid of the formation. During this control modality, the translational gestures are referred to as *call gestures*. Fig. 3.5 shows a time series of the Loco Swarm's and operator's trajectory, as seen from above, during this flight demonstration. As shown, regardless of their movement, an operator can

successfully control the position of a swarm in their body frame.

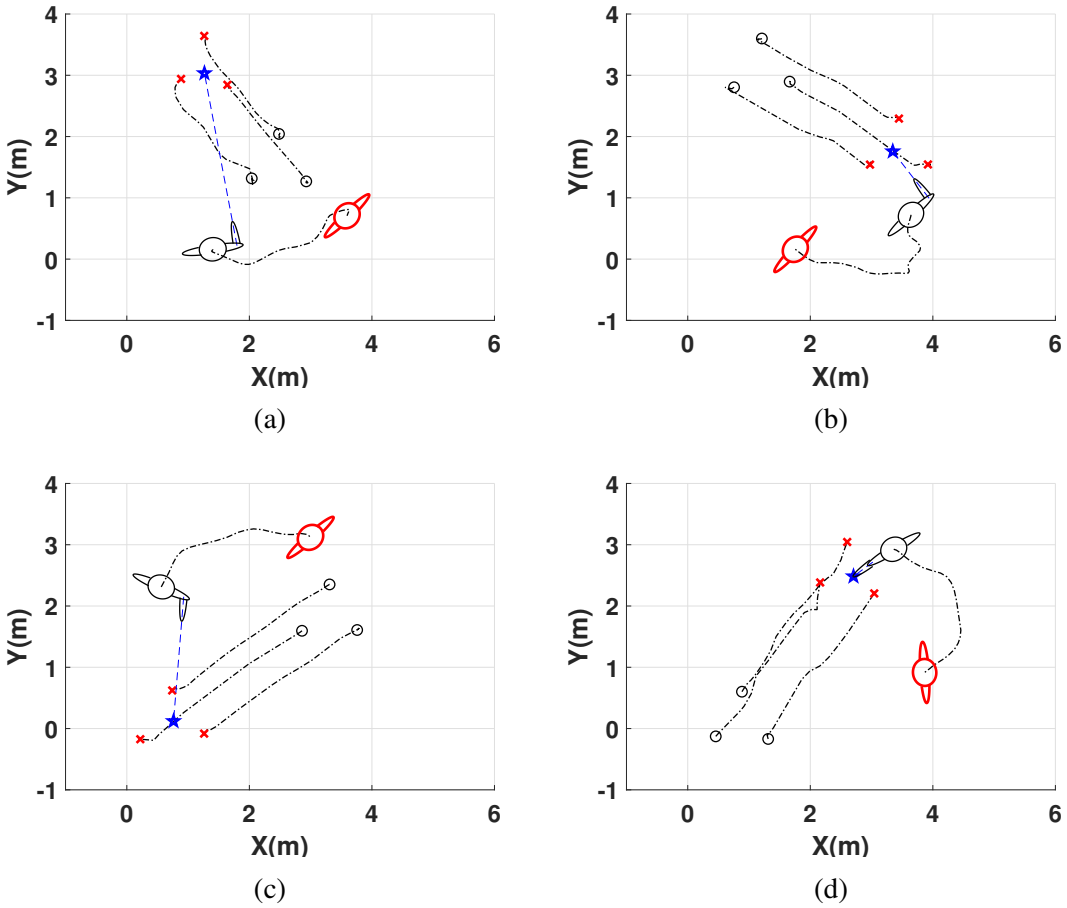


Figure 3.5: Time series (a-d) of the operator and Loco Swarm's X-Y trajectory during a flight demonstration utilizing positional control in the operator frame. The locations denoted with the ○ and × symbols represent the initial and final positions of the swarm respectively, before and after the call gesture was performed. The ⋆ symbol denotes the commanded intersection or call point.

3.3.4 Operator Collision Avoidance

The third experiment demonstrates the collision avoidance capabilities of the HSI. The operator is once again localized in the flight volume via the VICON system. Once localized, the position of the operator is added to the list of obstacles. The 3 agents were assigned an avoidance radii of 0.25m and the operator was assigned an avoidance radius of 1m. Fig. 3.6 shows the

minimum, maximum, and average inter-agent distances during this experiment, as well as the inter-agent distance prescribed by the formation. The formation is assigned a radius of 0.75m leading to a prescribed inter-agent distance of approximately 1.3m. At 115s into this experiment, the operator decreased the density of the swarm by increasing the formation radius from 0.75m to 1m, leading to a prescribed inter-agent distance of approximately 1.73m. The periods of time near 55s, 80s, and 115s show windows where the operator was not within collision range of the swarm. As shown, the lines converge, since the minimum distance, maximum distance, and average distance between agents are all equivalent for the assigned radially symmetric three-agent formation. This plot also shows that as the operator moves back and forth through the swarm, the agents at no point in time collided with each other and respect their assigned avoidance radii.

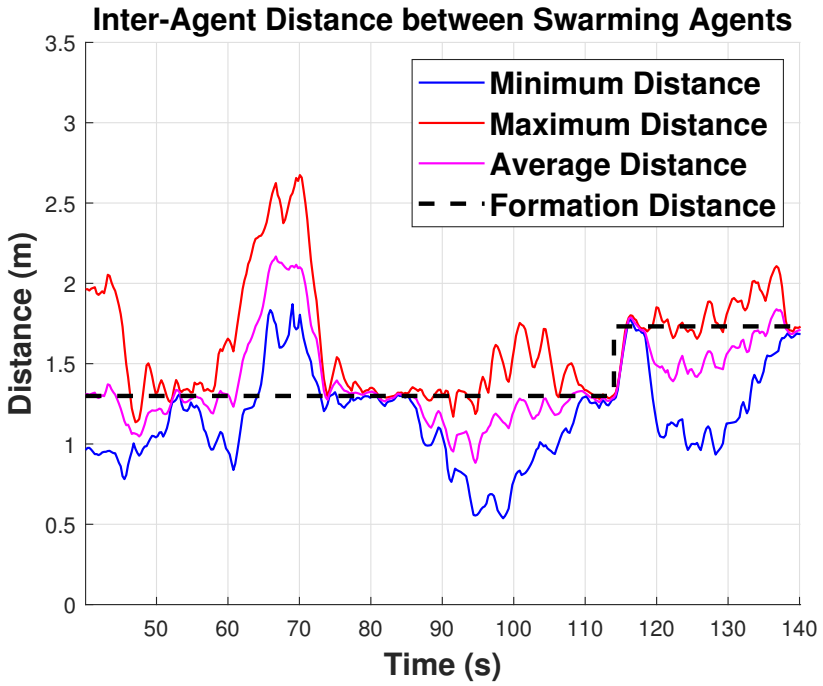


Figure 3.6: Time series showing the minimum, maximum, average, and formation based inter-agent distance during operator collision avoidance experiment.

Fig. 3.7 shows the distance between the operator and the agents within the swarm during

the same experiment. From this graph, we can see that during the experiment, the agents avoided collision with the operator, respecting the operator's designated boundary. Fig. 3.8 shows a time series of the Loco Swarm avoiding collision with the operator as the operator walks through the center of the flight volume. For a short window beginning at 60s, the operator moves forward quickly, forcing the Agent 1 to enter the operator's avoidance radius as can be seen in Fig. 3.8b and Fig. 3.8c. Agent 1 quickly compensates and exits the operator's avoidance radius in an effort to maintain desired distances between the obstacles. Thus, as the operator moves from one end of the flight volume to the other, the designed swarm velocity controller allows the Loco Swarm to actively avoid collision with all obstacles, which now include the operator, ensuring the operator's safety during control of a cobot swarm while occupying the same workspace. The operator still maintains control of the aerial swarm via the previously discussed gesture controls as can be seen at the end of the supplemental material.

3.4 Conclusion

This work presents a novel cobot human swarm interface (HSI) that prioritizes operator safety while reducing the cognitive load during control of a cobot swarm. The cognitive load required to control a single drone in the presence of a human occupying a confined space is quite high. This load is magnified significantly by increasing the number of aerial vehicles being controlled. The HSI uses EMG-based gesture control to command the position, orientation, and density of the swarm in both the inertial frame, as well as the operator's frame, removing the necessity of controlling multiple agents individually through the use of swarm formation control. The location of the centroid of the swarm is relayed to the operator via a vibrotactile haptic vest.

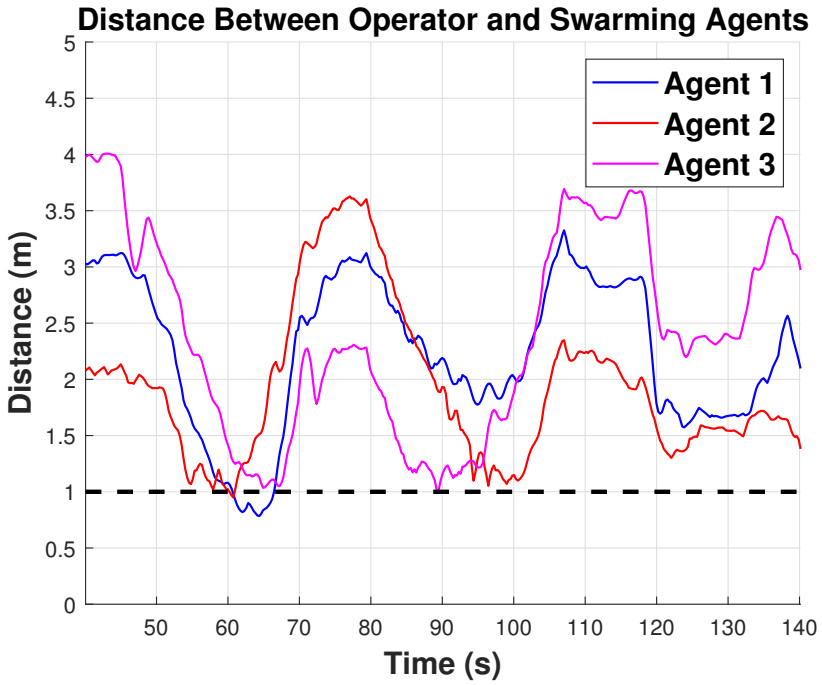
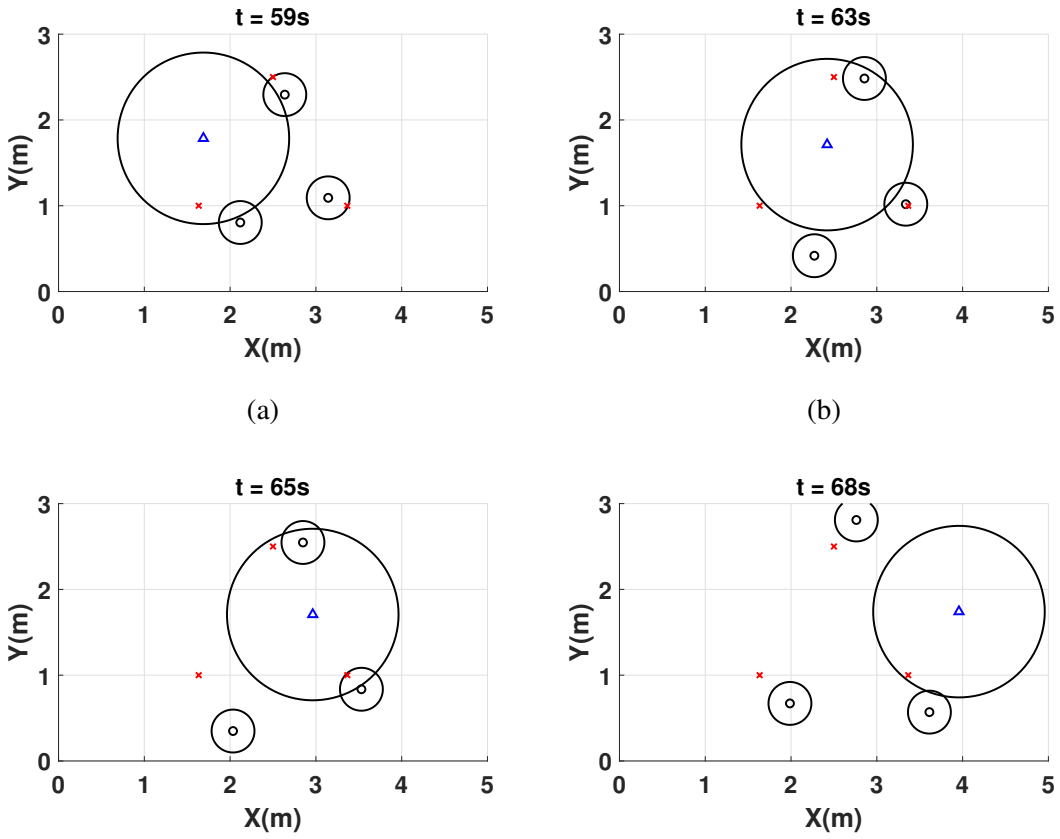


Figure 3.7: Time series of the distance between each agent and the operator during operator collision avoidance experiment.

Inter-agent as well as agent-operator collisions are prevented through a swarm velocity controller utilizing a distance-based potential function.

Experimental results demonstrate that an operator can control an aerial swarm while safely occupying and moving throughout the same workspace. Quantification of cognitive loads is a worthwhile endeavor for subsequent research. Ongoing and future work is focused on adapting this HSI to more computationally capable autonomous quadrotors, with an eye towards eliminating the requirement for infrastructure such as the motion capture system for operator localization and laptop base station.



(a) (b) (c) (d)

Figure 3.8: Time series (a-d) of the Loco Swarm avoiding collision as an operator walks through the center of the flight volume. The △ symbol denotes the position of the operator, while the ○ symbols denote the agents in the swarm. Both the operator and the agents are shown with their respective repulsive radii of influence. The × symbols represent the assigned goal locations for the agents.

Chapter 4: Multi-Sensor Pose and Parameter Estimation for Human-Robot Interactions

4.1 Introduction

This chapter investigates the development of a human-robot interface that obtains a user's pose as well as other anthropometric measurements useful for human-robot interactions. We develop a real-time pose tracking solution fusing a single camera and multiple acoustic sensors. The developed system assimilates visual and acoustic sensor data using an extended Kalman filter to estimate body dimensions. This system develops the capability to estimate the pose and body lengths of a user, providing robotic systems with a sense of scale for the individuals they're interacting with, as well as information about the user that may be used to develop more intuitive human-robot interfaces.

This chapter is organized as follows. Section 4.2 introduces the pose estimation methods employed by both the vision-based system and sound-based system. Section 4.3 describes the derivation of the state estimation system. Section 4.4 reports the experimental results. The conclusion and future work are discussed in Section 4.5.

4.2 Pose Estimation

This section presents the environmental variables as well as the depth and pose estimation methods used by both the camera and acoustic sensors.

4.2.1 Environment

This system aims to fuse both camera and acoustic sensors to estimate the pose of a user. The positions of the camera and two acoustic sensors in the inertial frame are denoted $\boldsymbol{r}_{O_C/O_I} = (x_C, y_C, z_C)_I$, $\boldsymbol{r}_{O_{M_1}/O_I} = (x_{M_1}, y_{M_1}, z_{M_1})_I$, and $\boldsymbol{r}_{O_{M_2}/O_I} = (x_{M_2}, y_{M_2}, z_{M_2})_I$, respectively. The rotation matrices from the camera and acoustic sensor frames to the inertial frame are denoted ${}^I\boldsymbol{R}^C$ and ${}^I\boldsymbol{R}^M$, respectively. The landmark locations in the image frame are $\boldsymbol{l}_N = (u_N, v_N, w_N)_P$, where $N = 0, \ldots, 32$ refers to the individual landmark, u_N and v_N are the landmark coordinates on the image plane in pixels, and w_N is the MediaPipe estimated landmark depth, which uses approximately the same scaling factor as u_N. The landmark locations in the camera frame are $\boldsymbol{L}_N = (x_N, y_N, z_N)_C$, where $N = 0, \ldots, 32$, paralleling the image plane coordinates. The direction of arrival vectors in the microphone and inertial frame are denoted $\boldsymbol{\Theta} = (\Theta_x, \Theta_y, \Theta_z)_M$ and $\boldsymbol{\vartheta} = (\vartheta_x, \vartheta_y, \vartheta_z)_I$, respectively. The mean hip position of the user in the camera frame is $\boldsymbol{r}_{U/O_C} = (x_U, y_U, z_U)_C$, whereas their position in the inertial frame is $\boldsymbol{r}_{U/O_I} = (x, y, z)_I$.

4.2.2 Monocular Position Estimation

MediaPipe provides the 3D points that define the skeletal frame of the user as shown in Fig. 2.4. Landmarks 11 and 12 ($\boldsymbol{l}_{11}, \boldsymbol{l}_{12}$) estimate the position of the user's left and right shoulders, respectively; landmarks 23 and 24 ($\boldsymbol{l}_{23}, \boldsymbol{l}_{24}$) estimate the position of the user's left and right hip,

respectively. These four landmarks define the user's torso. The pinhole camera model (2.2) is used to define rays passing through landmarks on the image plane to the landmarks on the user's body. Given the landmark location of the user's left shoulder, $\boldsymbol{l}_{11} = (u_{11}, v_{11}, w_{11})_P$, the landmark corresponding to the user's left shoulder in the camera frame $\boldsymbol{L}_{11} = (x_{11}, y_{11}, z_{11})_C$ is

$$x_{11} = \left(\frac{u_{11} - u_0}{f_x}\right) z_{11} \qquad\qquad y_{11} = \left(\frac{v_{11} - v_0}{f_y}\right) z_{11} \tag{4.1}$$

To find the depth of the user in the camera frame, which is equated to the depth z_U of the mean hip position the user, the pinhole camera model is used to generate two rays. One passing through the midpoint of the shoulder landmarks $\boldsymbol{l}_s = \frac{1}{2}(\boldsymbol{l}_{11} + \boldsymbol{l}_{12}) = (u_s, v_s, w_s)_P$, while the other passes through the midpoint of the hip landmarks $\boldsymbol{l}_h = \frac{1}{2}(\boldsymbol{l}_{23} + \boldsymbol{l}_{24}) = (u_h, v_h, w_h)_P$. These two rays are called $\overline{O_c L_s}$ and $\overline{O_c L_h}$, respectively, and are defined below and shown in Fig. 4.1:

$$\overline{O_c L_s} = \left[\left(\frac{u_s - u_0}{f_x}\right) z_u, \left(\frac{v_s - v_0}{f_y}\right) z_u, z_u\right] \tag{4.2}$$

$$\overline{O_c L_h} = \left[\left(\frac{u_h - u_0}{f_x}\right) z_u, \left(\frac{v_h - v_0}{f_y}\right) z_u, z_u\right] \tag{4.3}$$

This model assumes that the user is standing upright as opposed to bending at the hip. The distance between the two rays is calculated and set equal to a reference length, as shown in Eqn. (4.4) below. The user's torso length is chosen as the reference length since it remains

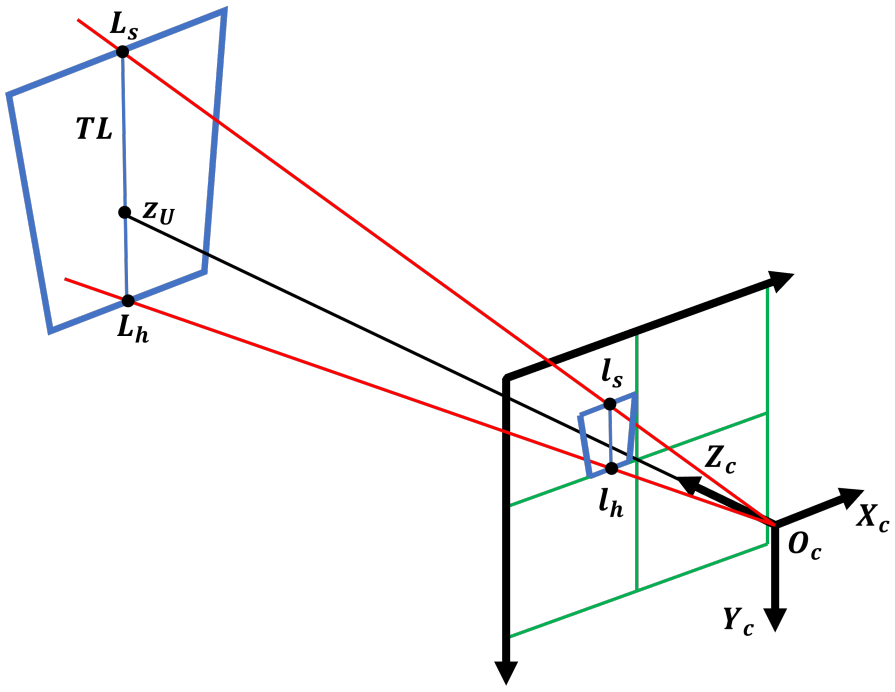


Figure 4.1: Estimating depth z_U via the geometric relationship between the pinhole camera model and rays passing through $\boldsymbol{l}_s$ and $\boldsymbol{l}_h$.

constant as the user yaws, i.e.,

$$TL = \|\overline{O_c L_s} - \overline{O_c L_h}\| \tag{4.4}$$

Given $\overline{O_c L_s}$ and $\overline{O_c L_h}$ defined in Eqn. (4.2) and (4.3), Eqn. (4.4) is then explicitly solved for z_U, resulting in the estimated depth

$$z_U = \left[\frac{TL^2}{\left[\left(\frac{u_s - u_h}{f_x} \right)^2 + \left(\frac{v_s - v_h}{f_y} \right)^2 \right]} \right]^{\frac{1}{2}} \tag{4.5}$$

The azimuth θ_U and elevation ϕ_U from the camera to the user's mean hip positon are

$$\theta_U = \tan^{-1}\left(\frac{u_h - u_0}{f_x}\right) \qquad \phi_U = \tan^{-1}\left(\frac{v_h - v_0}{f_y}\right) \tag{4.6}$$

Given z_U, $x_U = z_U \tan(\theta_U)$ and $y_U = z_U \tan(\phi_U)$ resulting in the user's position in the camera frame $\boldsymbol{r}_{U/O_C} = (x_U, y_U, z_U)_C$. Furthermore, the position in the inertial frame is

$$\boldsymbol{r}_{U/O_I} = \left({}^I\boldsymbol{R}^C\right)\boldsymbol{r}_{U/O_C} + \boldsymbol{r}_{O_C/O_I} \tag{4.7}$$

Thus, with prior knowledge of the user's torso length, the position of the user may be estimated in both the inertial and camera frames. However, this work does not assume knowledge of the user's anthropometric measurements and instead inverts this relationship to develop the vision-based system's observation model derived in Appendix A.

4.2.3 Sound-based Position Estimation

Two microphone arrays are utilized to localize the user in the environment. Given the microphone arrays' positions in the inertial frame $\boldsymbol{r}_{O_{M_1}/O_I} = (x_{M_1}, y_{M_1}, z_{M_1})_I$, $\boldsymbol{r}_{O_{M_2}/O_I} = (x_{M_2}, y_{M_2}, z_{M_2})_I$, their rotation matrices from their frames to the inertial frame ${}^I\boldsymbol{R}^{M_1}$, ${}^I\boldsymbol{R}^{M_2}$, and their calculated directions of arrival in their respective microphone array frames Θ_1, Θ_2, the 2D position (x, y) of

the user in the inertial frame may be found as shown below:

$$\vartheta_1 = \left({}^{I}R^{M_1}\right)\Theta_1, \qquad \theta_1 = \tan^{-1}\left(\frac{\vartheta_{1,y}}{\vartheta_{1,x}}\right) \tag{4.8}$$

$$\vartheta_2 = \left({}^{I}R^{M_2}\right)\Theta_2, \qquad \theta_2 = \tan^{-1}\left(\frac{\vartheta_{2,y}}{\vartheta_{2,x}}\right) \tag{4.9}$$

$$x = \frac{(y_{M_2} - y_{M_1}) + (x_{M_1}\tan(\theta_1) - x_{M_2}\tan(\theta_2))}{\tan(\theta_1) - \tan(\theta_2)} \tag{4.10}$$

$$y = (x - x_{M_1})\tan(\theta_1) + y_{M_1} \tag{4.11}$$

where ϑ_1, ϑ_2 are the direction-of-arrival vectors in the inertial frame, θ_1, θ_2 are the directions of arrival in the inertial frame, and (x, y) is the 2D position of user in the inertial frame.

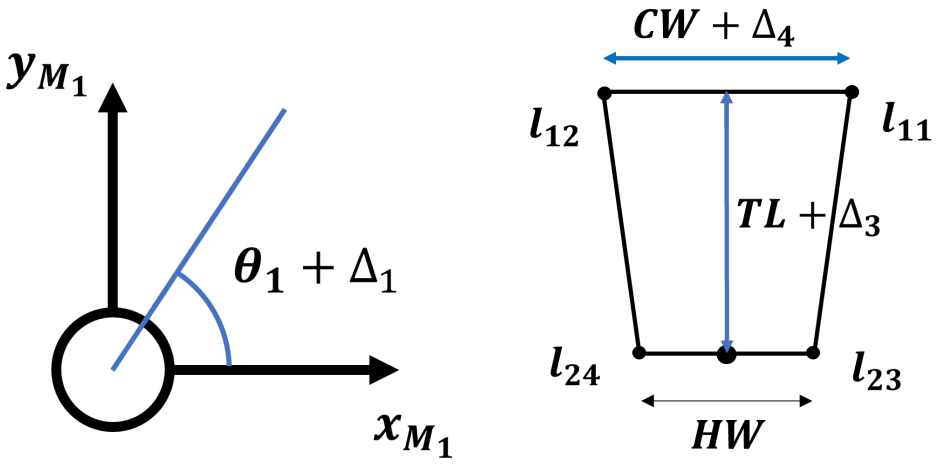
4.3 State Estimation

The extended Kalman filter (EKF) is a nonlinear extension of the Kalman filter [91]. Given a nonlinear system, an EKF linearizes the original nonlinear filter dynamics around the previous state estimates to estimate the current state of a nonlinear system given noisy measurements [92]. The EKF implemented in this work follows the standard Continuous-Discrete Extended Kalman Filter described in [91].

4.3.1 Measurement Bias and User Parameters

In this work, we augment the state vector by incorporating bias states to estimate the constant error offsets in measurements and user parameters. Let Δ_1 and Δ_2 represent the biases in direction of arrival measurements as shown in Fig. 4.2a. These biases are added to compensate for any errors in the estimated poses of the microphones. Fig. 4.2b shows the model used to de-

fine the relationship between the user's torso length TL, chest width CW, and hip width HW. To estimate the user's parameters, nominal values for the torso length TL and chest width CW are selected, and bias terms Δ_3 and Δ_4 are added to these nominal values to estimate the error in these selected values against the user's true dimensions. No bias term was added to correct errors in estimation of hip width, because the focus of this work is on developing common communication methodologies, few if any of which use hip gestures.



(a) (b)

Figure 4.2: (a) Microphone array
1 measuring direction of arrival θ_1 with additive bias term Δ_1 estimating the measurement bias; (b) model used to define user parameters for online estimation showing the relationship between the user parameters and their respective biases.

4.3.2 Measurement and State Definition

The measurements provided to this system are the directions of arrival θ_1, θ_2, the user heading ψ, and the relevant image plane MediaPipe landmarks $u_{11}, v_{11}, u_{12}, v_{12}, u_{23}, v_{23}, u_{24}, v_{24}$.

Thus, the measurement vector η for this system is as follows:

$$\eta = (\theta_1, \theta_2, \psi, u_{11}, v_{11}, u_{12}, v_{12}, u_{23}, v_{23}, u_{24}, v_{24}) \tag{4.12}$$

The state vector for this system ξ is

$$\xi = (x, y, z, \psi, \Delta_1, \Delta_2, \Delta_3, \Delta_4, s) \tag{4.13}$$

where (x, y, z) is the position of the user in the inertial frame, ψ is the heading in the environment, Δ_1 and Δ_2 are the bias estimates associated with the direction of arrival measurements for microphone arrays 1 and 2 respectively, and Δ_3 and Δ_4 are the bias estimates associated with the torso length TL and chest width CW estimates, respectively.

4.3.3 Kinematic Model

The user is modeled as a constant-speed variable-heading planar rigid body, as shown in Fig. 4.3. The states (x, y) are the position of the user, ψ is the heading, and s is the speed. Assume the velocity is aligned with the user's direction of travel.

The dynamics for the user are

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\psi} \\ \dot{s} \end{bmatrix} = \begin{bmatrix} -s\sin(\psi) \\ -s\cos(\psi) \\ 0 \\ 0 \end{bmatrix} \tag{4.14}$$

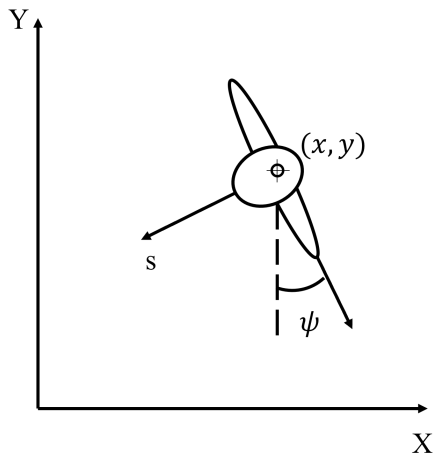


Figure 4.3: Kinematic model of the user

4.3.4 State Transition Model and Matrix

Given the user dynamics (4.14), the state transition model for the EKF is

$$\dot{\xi} = f(\xi) = \begin{Bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\psi} \\ \dot{\Delta}_1 \\ \dot{\Delta}_2 \\ \dot{\Delta}_3 \\ \dot{\Delta}_4 \\ \dot{s} \end{Bmatrix} = \begin{Bmatrix} -s\sin(\psi) \\ -s\cos(\psi) \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{Bmatrix} \tag{4.15}$$

Let $F = \frac{\partial f}{\partial \xi} \mid_{\xi=\hat{\xi}}$. We have the state transition matrix

$$F = \begin{bmatrix} 0 \;\; 0 \;\; 0 \;\; -\hat{s}\cos(\hat{\psi}) \;\; 0 \;\; 0 \;\; 0 \;\; 0 \;\; -\sin(\hat{\psi}) \\ 0 \;\; 0 \;\; 0 \;\; \hat{s}\sin(\hat{\psi}) \;\; 0 \;\; 0 \;\; 0 \;\; 0 \;\; -\cos(\hat{\psi}) \\ 0_{7\times 9} \end{bmatrix} \tag{4.16}$$

where $\hat{\psi}$ and $\hat{s}$ are the heading and speed estimates, respectively.

4.3.5 Observation Models and Matrices

Given the estimated states, the heading and observed landmarks in the image frame can be derived. The structure for the observation model from the vision-based system is shown below. The explicit definition can be found in Appendix A. We have

$$\eta = h_{MP}(\xi) = \begin{Bmatrix} 0_{2\times 1} \\ \psi \\ u_{11}(x, y, z, \psi, \Delta_4) \\ v_{11}(x, y, z, \Delta_3) \\ u_{12}(x, y, z, \psi, \Delta_4) \\ v_{12}(x, y, z, \Delta_3) \\ u_{23}(x, y, z, \psi) \\ v_{23}(x, y, z,) \\ u_{24}(x, y, z, \psi) \\ v_{24}(x, y, z) \end{Bmatrix} \tag{4.17}$$

Let $H_{MP} = \frac{\partial h_{MP}}{\partial \xi} \mid_{\xi=\hat{\xi}}$. We have

$$H_{MP} = \begin{bmatrix} & & & 0_{2\times 9} & & & & & \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ \frac{\partial u_{11}}{\partial \hat{x}} & \frac{\partial u_{11}}{\partial \hat{y}} & \frac{\partial u_{11}}{\partial \hat{z}} & \frac{\partial u_{11}}{\partial \hat{\psi}} & 0 & 0 & 0 & \frac{\partial u_{11}}{\partial \hat{\Delta}_4} & 0 \\ \frac{\partial v_{11}}{\partial \hat{x}} & \frac{\partial v_{11}}{\partial \hat{y}} & \frac{\partial v_{11}}{\partial \hat{z}} & 0 & 0 & 0 & \frac{\partial u_{11}}{\partial \hat{\Delta}_3} & 0 & 0 \\ \frac{\partial u_{12}}{\partial \hat{x}} & \frac{\partial u_{12}}{\partial \hat{y}} & \frac{\partial u_{12}}{\partial \hat{z}} & \frac{\partial u_{12}}{\partial \hat{\psi}} & 0 & 0 & 0 & \frac{\partial u_{12}}{\partial \hat{\Delta}_4} & 0 \\ \frac{\partial v_{12}}{\partial \hat{x}} & \frac{\partial v_{12}}{\partial \hat{y}} & \frac{\partial v_{12}}{\partial \hat{z}} & 0 & 0 & 0 & \frac{\partial u_{12}}{\partial \hat{\Delta}_3} & 0 & 0 \\ \frac{\partial u_{23}}{\partial \hat{x}} & \frac{\partial u_{23}}{\partial \hat{y}} & \frac{\partial u_{23}}{\partial \hat{z}} & \frac{\partial u_{23}}{\partial \hat{\psi}} & 0 & 0 & 0 & 0 & 0 \\ \frac{\partial v_{23}}{\partial \hat{x}} & \frac{\partial v_{23}}{\partial \hat{y}} & \frac{\partial v_{23}}{\partial \hat{z}} & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{\partial u_{24}}{\partial \hat{x}} & \frac{\partial u_{24}}{\partial \hat{y}} & \frac{\partial u_{24}}{\partial \hat{z}} & \frac{\partial u_{24}}{\partial \hat{\psi}} & 0 & 0 & 0 & 0 & 0 \\ \frac{\partial v_{24}}{\partial \hat{x}} & \frac{\partial v_{24}}{\partial \hat{y}} & \frac{\partial v_{24}}{\partial \hat{z}} & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{4.18}$$

Given the known positions of the microphone arrays and the estimated states, the observed directions of arrival can be found as follows. The observation model from the acoustic system is

$$\begin{aligned} \eta = h_{ODAS}(\xi) &= \begin{Bmatrix} \theta_1(x, y, \Delta_1) \\ \theta_2(x, y, \Delta_2) \\ 0_{9\times 1} \end{Bmatrix} \\ &= \begin{Bmatrix} \tan^{-1}\left(\frac{y-y_{M_1}}{x-x_{M_1}}\right) - \Delta_1 \\ \tan^{-1}\left(\frac{y-y_{M_2}}{x-x_{M_2}}\right) - \Delta_2 \\ 0_{9\times 1} \end{Bmatrix} \end{aligned} \tag{4.19}$$

Let $H_{ODAS} = \frac{\partial h_{ODAS}}{\partial \xi} \mid_{\xi=\hat{\xi}}$. We have

$$H_{ODAS} = \begin{bmatrix} \frac{\partial \theta_1}{\partial \hat{x}} & \frac{\partial \theta_1}{\partial \hat{y}} & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ \frac{\partial \theta_2}{\partial \hat{x}} & \frac{\partial \theta_2}{\partial \hat{y}} & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ & & & & 0_{9\times 9} & & & & \end{bmatrix} \tag{4.20}$$

4.4 Experimental Results

The following experiments were conducted to demonstrate the position estimation capabilities of the vision and acoustic systems individually, and the pose estimation capabilities of the sensor fusion EKF.

4.4.1 Experimental Setup

The system shown in these results is made up of two UMA-8 microphone arrays with a baseline separation of $2m$ and a webcam from a Gen8 ThinkPad X1 Carbon. Fig. 4.4 shows the relevant reference frames, estimated states, and experimental setup.

4.4.2 Monocular Position Estimation

In the first experiment, the position estimation capabilities of the monocular system described in Section 4.2.2 are demonstrated. The user walked throughout the environment using a lawnmower pattern, while recording their estimated and ground truth positions. Fig. 4.5 shows a heat map depicting the error in the estimated position versus ground truth. Throughout the experiment, the camera was rigidly mounted, thus the system is limited to the visibility within the camera's viewing angle, resulting in a conical heat map. This experiment demonstrates that

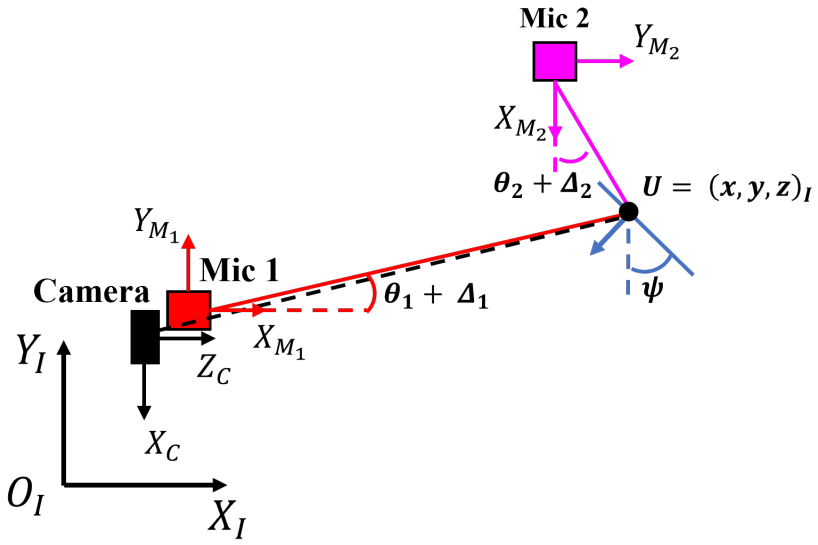


Figure 4.4: Experimental setup showing the locations and reference frames of the sensors, as well the estimated EKF states in the environment.

as long as the user is visible to the camera system, the user's position can be estimated with estimation error that increases with range.

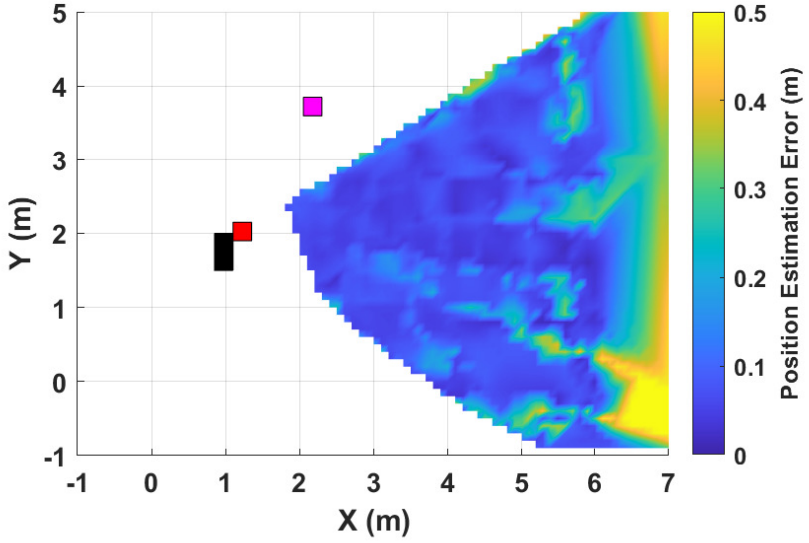


Figure 4.5: Heat map showing the error in position estimation when using the vision-based system.

4.4.3 Sound-based Position Estimation

In the second experiment, the position estimation capabilities of the sound-based system as described in Section 4.2.3 are demonstrated. The user walked throughout the environment using a lawnmower pattern and held a speaker near the center of their chest to continuously provide sound for the direction of arrival calculations, while recording their estimated and ground truth positions. Fig. 4.6 shows a heat map depicting the error in the estimated position versus ground truth. This experiment demonstrates that as long as the user can be heard by the sound-based system, the user's position can be estimated with increased errors on the line passing between the two microphone arrays.

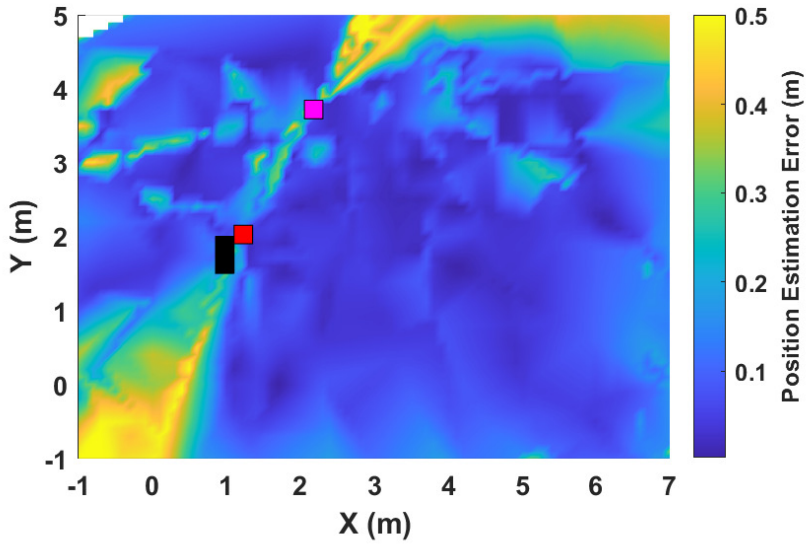


Figure 4.6: Heat map showing the error in position estimation when using the sound-based system.

4.4.4 EKF-based User Pose Estimation

In the third experiment, the pose estimation capabilities of the sensor fusion EKF described in Section 4.3 are demonstrated. Throughout this 5-minute experiment, the user performed a number of maneuvers to rigorously examine the capabilities of the system. From $0s$ to $20s$, the user was initializing the system and walking into place. From $20s$ to $115s$, the user performed a zig-zag maneuver along the X-axis while standing still for $15s$ after each movement. From $115s$ to $150s$, the user walked in a circle. From $150s$ to $210s$, the user was stationary. From $210s$ to $235s$, the user walked in a serpentine pattern along the X-axis. From $235s$ to $300s$, the user walked randomly throughout the environment. Fig. 4.7 shows the position (x, y), heading ψ, and speed s, estimates of the user throughout the 5-minute interval during experiment 3.

The average error throughout experiment 3 for the three position estimation systems are in Tab. 4.1

	$x(m)$	$y(m)$	$\psi(deg)$
Vision-based	0.5047	-0.0566	-3.9895
Sound-based	-0.4527	-0.1524	-
EKF	-0.1681	-0.0504	12.0425

Table 4.1: Position and heading error comparison between the vision-based, sound-based, and EKF systems.

Fig. 4.8 shows the estimated biases for the microphones as well as the user parameters. The user for this experiment had a measured torso length (TL) of $23in$ ($0.5842m$), a chest width (CW) of $18in$ ($0.4572m$), and a hip width (HW) of $14in$ ($0.3556m$). The EKF was provided with nominal values of $0.6m$, $0.5m$, and $0.3m$. Given mean estimated biases of -0.1176m and -0.0020m, the estimated torso length of the user was $0.7176m$ and the estimated chest width of

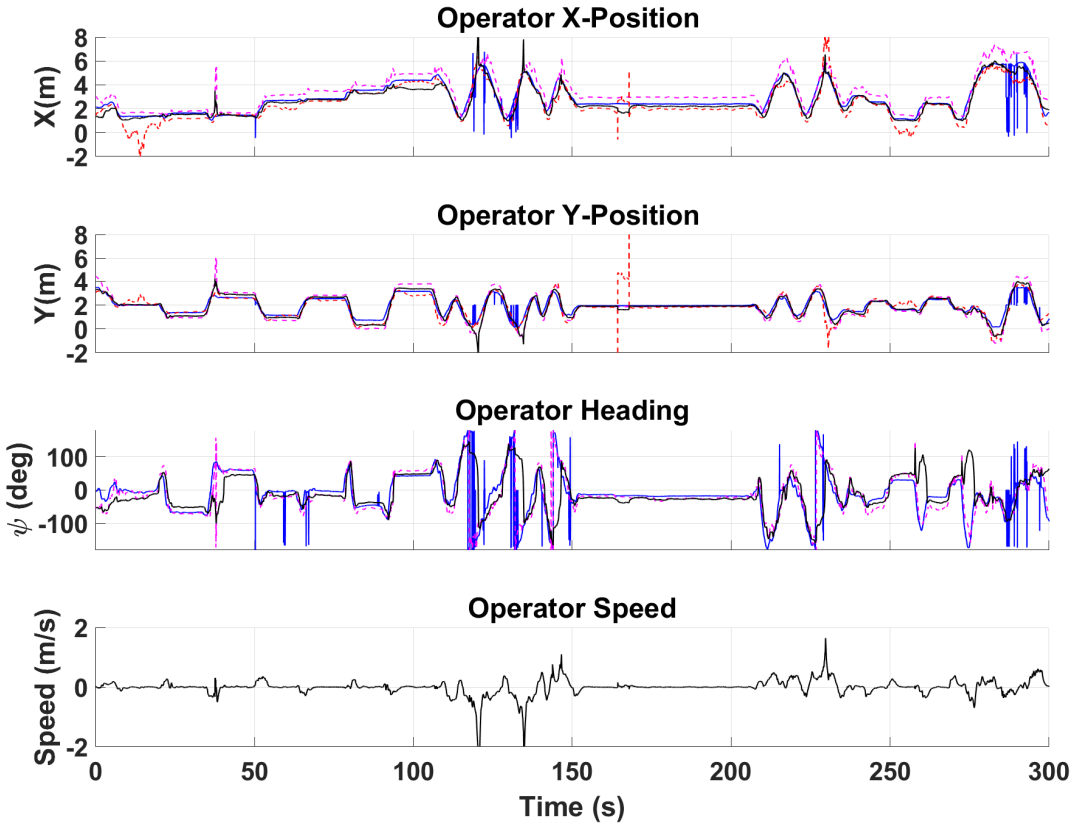


Figure 4.7: Time series showing the estimated position (x, y), heading ψ, and speed s, of the user throughout experiment 3. The red, magenta, black, and blue lines represent the estimated states from the vision-based, sound-based, EKF, and ground truth systems respectively.

the user was $0.5020m$. This experiment demonstrates that the sensor fusion EKF can be used to estimate the pose as well user parameters online.

4.4.5 Human Robot Interface

In the fourth experiment, the pose and user parameter estimation capabilities of the sensor fusion EKF are utilized in conjunction with the MediaPipe skeletal frame to demonstrate the use of this system in a human-robot interface. The user is tasked with pointing at a white bucket in the environment and the location that the user is pointing at is estimated. The mean hip position of the

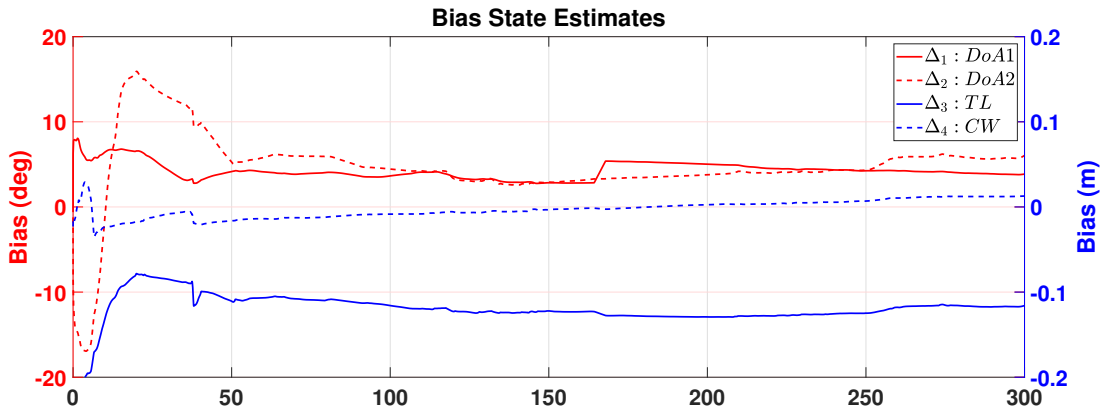


Figure 4.8: Time series showing the estimated sensor and user parameter biases throughout experiment 3. DoA1 and DoA2 refer to the sensor biases for the direction of arrival measurements, and TL and CW refer to the parameter biases for the user's torso length and chest width.

MediaPipe skeletal frame is positioned at the EKF's estimated position of the user. The skeletal frame is then scaled such that the chest width of the skeletal frame $(0.3212m)$ is equivalent to the user's estimated chest width $(0.5020m)$. The skeletal frame is then shifted vertically along the Z-axis such that the average position of the feet landmarks $(lm_{29}, lm_{30}, lm_{31}, lm_{32})$ are zero. Once the frame has been properly scaled and positioned, a pointing ray from the user's shoulder to their wrist is generated, and the intersection between that ray and the floor may be calculated. Fig. 4.9 and Fig. 4.10 show an image from the video feed provided to the system as well as the 3D reconstruction of the system estimating the ground point. The white bucket is located at $(4.58m, 3.48m)$. The estimated location that the user is pointing at is $(4.60m, 3.27m)$ resulting in an error of 0.21m. This experiment demonstrates that the proposed system can be used to estimate where the user is pointing, potentially providing contextual information to robotic system about a user intention.

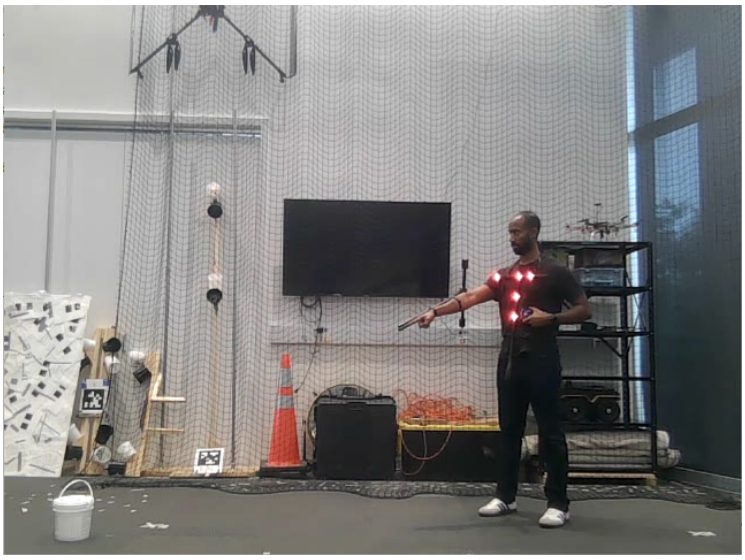


Figure 4.9: Image from the video feed provided to MediaPipe depicting the user pointing at an object (white bucket) of interest. The red LEDs on the user's chest is a VICON wand, used to collect ground truth position and orientation data.

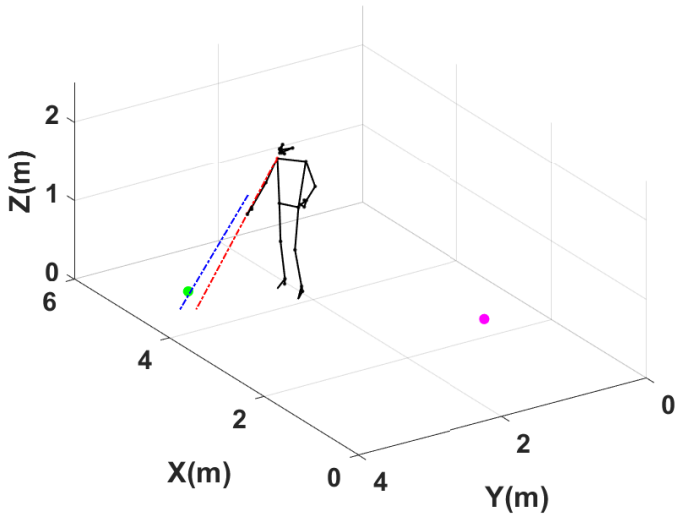


Figure 4.10: 3D reconstruction of the user's MediaPipe skeletal frame using the estimated pose and biases from the EKF to estimate the ground position the user is pointing at as shown in Fig. 4.9. The red and blue lines are the pointing rays generated from the user's pose estimation and ground truth system, respectively, whereas the magenta and green markers are the locations of the camera and ground truth location for the white bucket respectively.

4.5 Conclusion

Human pose recognition and estimation is a common task in computer vision and a key step in enabling safe and intuitive human-robot interfaces. One challenge is providing mobile robotic systems with the capabilities to localize and interact with users in their environment intuitively. This work presents a novel method for obtaining the pose as well as anthropometric measurements of a user. A camera and a keypoint detection package are used to estimate the depth and pose of a user. Multiple acoustic sensors are used to also localize the user. State augmentation within an EKF was used to fuse these estimates while also solving for the user's torso length and chest width.

Experimental results demonstrate that this system can successfully estimate the pose, measurement biases, and body lengths of a user. Additional experiments demonstrate the sensor fusion system's adaptability to a human-robot interface. Ongoing and future work is focused on conducting experiments with a diverse range of users to validate this framework's usability for estimating pose and anthropometric measurements, as well as adapting this system to mobile platforms.

Chapter 5: Command and Control of an Outdoor Swarm via a Mobile Interface

5.1 Introduction

This chapter investigates the development of a portable device, the smart binoculars, designed to facilitate interactions between users and multi-agent autonomous systems in dynamic outdoor environments. Traditional ground control stations offer users a number of useful functionalities but are often limited by the mobility of their computational hardware. The smart binoculars allow a user to select desired locations and assign tasks for robotic systems to complete at those locations, facilitating the command and control of multi-agent systems for line-of-sight operations.

This chapter is organized as follows. Section 5.2 introduces the smart binoculars, the involved hardware, target localization associated functionalities, and reviews the evolution of the system. Section 5.3 reports the experimental results. The conclusion and future work are discussed in Section 5.4.

5.2 System Design

This section outlines the hardware used in the smart binoculars, details the process for determining outdoor coordinates, reviews their task assignment capabilities, and describes the

evolution of the system throughout the design process.

5.2.1 Hardware Design

The smart binocular system is composed of six distinct pieces of hardware. The Hireed D800, a long-range laser-based distance sensor, is used to measure the distance between the user and a desired location within range. The Holybro H-RTK F9P Helical [93] provides our system with the capability to measure the magnetic heading of the smart binoculars and localize the operator in the global frame. The Adafruit ISM330DHCX + LIS3MDL Featherwing [94], a high precision 9-DoF IMU, is used to measure the orientation of the smart binoculars while a user targets a location. A Raspberry Pi 4 [95] is used to collate the lidar, GPS, and IMU data to estimate the location a user is targeting. Mechanical lever switches are used to capture the location the user is currently targeting. The system communicates to the mobile platforms as well as a ground station via a Doodle Radio Mini [96]. Fig. 5.1 shows the assembled smart binocular.

5.2.2 Target Acquisition

This system aims to estimate the location a user is targeting and provide commands for multi-agent systems to carry out at that location. The geographic position of the user and the targeted locations are denoted (lat_0, lon_0) and (lat_n, lon_n), respectively, where $n = 1 \dots N$ refers to the number of targeted locations selected by the user. Given the elevation of the smart binocular θ and the range to a target ρ_{lidar}, the geographic range to a target can be found as shown in Fig. 5.2 given the following relationship: $\rho = \rho_{lidar} * cos\,(\theta)$. Given the geographic location

Figure 5.1: Smart binoculars (SB-V4)

of the user (lat_0, lon_0), the geographic range to the target ρ_{lidar}, and the magnetic heading β, targeted locations (lat_n, lon_n) can be estimated using the relationship defined in Eqn. 2.3 and 2.4.

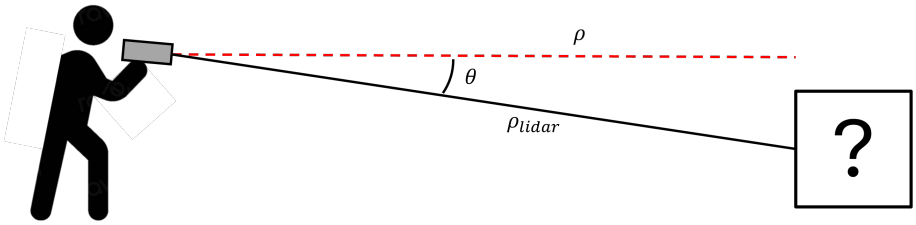
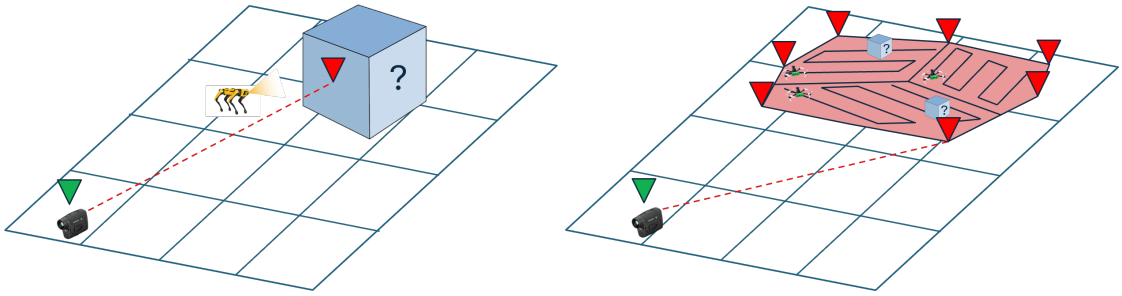


Figure 5.2: Diagram showing relationship between lidar range ρ_{lidar} and geospatial range ρ

5.2.3 Task Assignment

The targeted positions (lat_n, lon_n) are recorded on board the smart binoculars. Once the desired number of locations have been recorded, they are sent to the ground control station to either send to autonomous agents as waypoints to be visited or define a search area. Examples of both engagements can be seen in Fig. 5.3.



(a) (b)

Figure 5.3: Example engagements of the smart binoculars being used to select (a) single and (b) multiple locations for task assignment.

5.2.4 Smart Binocular Evolution

The smart binocular system has evolved in both functionality and capability. The original smart binocular (SB-V1) shown in Fig. 5.4a was designed to facilitate localizing a target. While aiming at the target, a second operator at the ground control station recorded the target's coordinates and relayed them to an Unmanned Aerial Vehicle (UAV). The UAV then launched, flew to the provided waypoint, and returned to the takeoff location. This prototype acted as a proof of concept demonstrating the usability and effectiveness of a mobile outdoor HSI. With the introduction of a mechanical levers on the SB-V2 shown in Fig. 5.4b, we introduced the capability to select and record desired waypoints. This upgrade allowed the smart binoculars to not only

control the position of a UAV but reposition the UAV dynamically during live operation. The SB-V3 shown in Fig. 5.4c builds upon the SB-V2's design and enhances overall performance. The lidar rangefinder was upgraded, increasing the targeting range from $50m$ to $100m$. The communication system transitioned from Microhard Radios to Doodle Radios, enabling the system's integration into multi-agent mesh networks. The onboard computation was shifted from a ModalAI VOXL2 [97] to a Raspberry Pi 4, facilitating unrestricted access to serial devices and streamlining development and deployment. The SB-V4 shown in Fig. 5.4d maintains the overall design and structure of the SB-V3 but utilizes higher quality sensors to improve overall performance. The GNSS module was upgraded to increase the accuracy of the binocular's GPS location and heading. The lidar was upgraded to extend the targeting range from $100m$ to $200m$. Tab. 5.1 shows the evolution of the smart binocular and compares the hardware and capabilities of each model.

(a) (b) (c) (d)

Figure 5.4: (a) SB-V1, (b) SB-V2, (c) SB-V3, (d) SB-V4

Name	Dev. Date	Range	Comm.	Comp.	Tgt. Loc.	Tgt. Sel.	Pos. Ctrl.	Task. Ass.
SB-V1	Jul 2022	50m	Microhard	VOXL2	Y	N	N	N
SB-V2	Aug 2023	50m	Microhard	VOXL2	Y	Y	Y	N
SB-V3	Mar 2024	100m	Doodle	Rasp. Pi 4	Y	Y	Y	Y
SB-V4	Aug 2024	200m	Doodle	Rasp. Pi 4	Y	Y	Y	Y

Table 5.1: Evolution of the smart binocular's functionalities and capabilities.

5.3 Experimental Results

The following experiment was conducted to demonstrate the target selection, single agent, and multi-agent reposition capabilities.

5.3.1 Experimental Setup

The system shown in these results is made up of two ModalAI M500 Drones [98] running a custom written autonomy stack, the smart binoculars (SB-V2) described in Sec. 5.2.4, and a ground control station. Fig. 5.5 shows an aerial view of the experiment site. The user targets four points of interest (A-D) and sends either an individual or multi-agent system to those points.

5.3.2 Smart Binocular Positional Control

In this experiment, the positional control capabilities of the smart binoculars are demonstrated. During the first phase of this experiment, the user walked along the tarmac, targeting specific points of interest. As the coordinates were selected and transmitted to the UAVs, the swarm autonomously navigated to the designated locations such that the center of the formation was above the targeted coordinates. The aerial swarm employed the previously defined multi-agent formation and swarm velocity controller described in Sec. 3.2. Fig. 5.6 illustrates a time

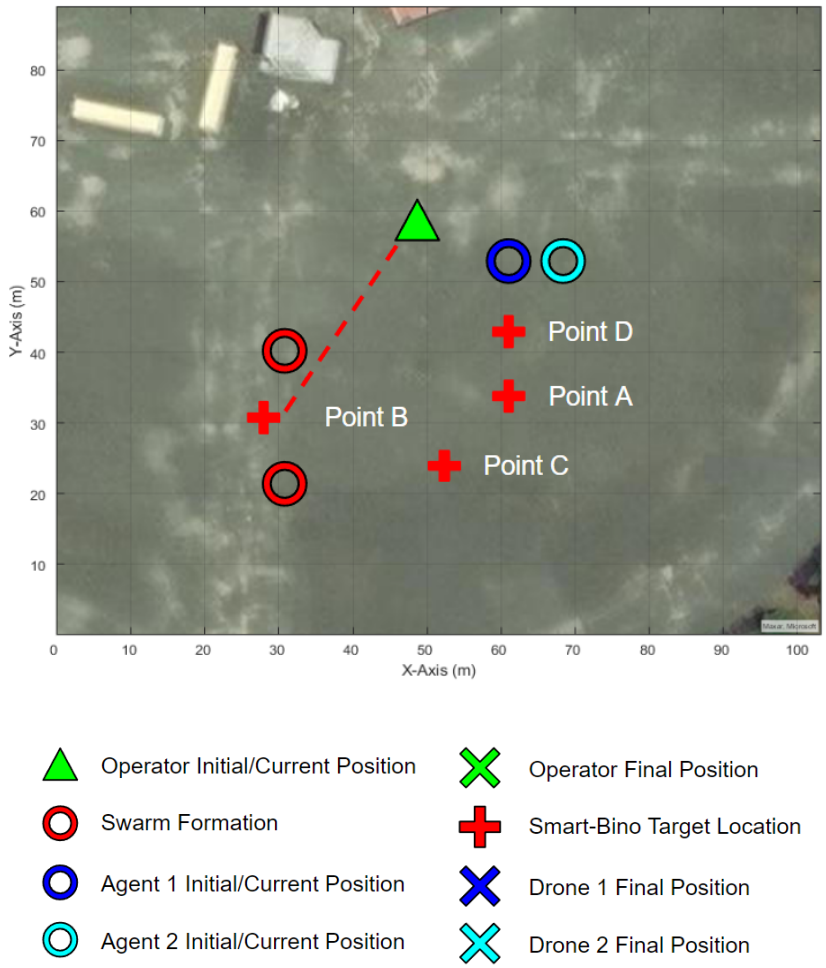


Figure 5.5: Aerial map showing outdoor experiment site, symbol definitions, and relevant points of interest

series of the UAVs trajectories, as seen from above, during the swarm reposition experiment. These results show that an operator can effectively control a swarm's position while moving outdoors using the smart binoculars.

During the second phase of the experiment, the user continued walking along the tarmac, targeting specific points of interest. As the coordinates were selected, the user also selected which agent to transmit the coordinates to, displaying positional control of an individual agent within a swarm. Next the user selected another location to relocate the entire swarm to.

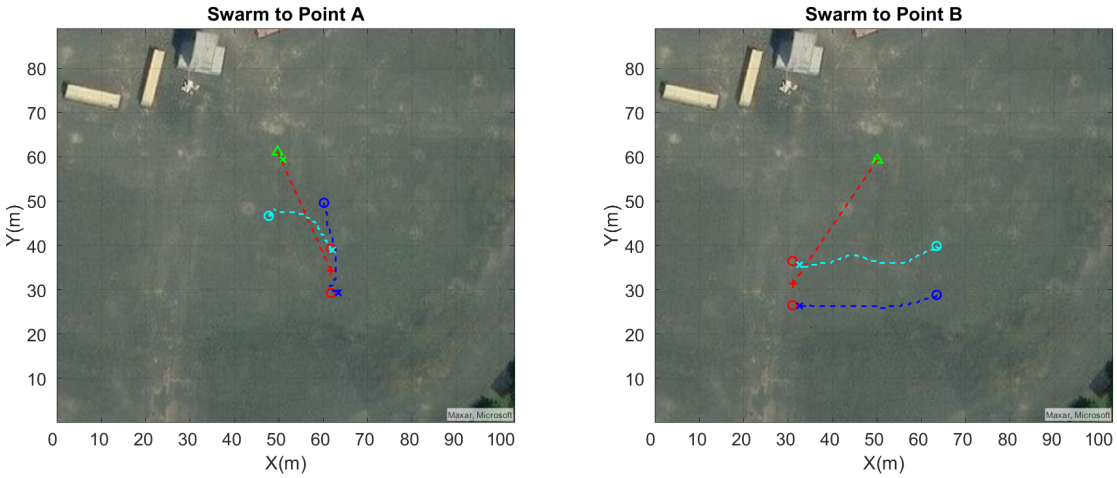


Figure 5.6: Smart binocular repositioning aerial swarm to point A and point B

Fig. 5.7 illustrates a time series of the UAVs trajectories, as seen from above, during the swarm disassembly and reassembly experiment. These results show that an operator can effectively control the position of both an individual agent as well as swarm's position while moving outdoors using the smart binoculars.

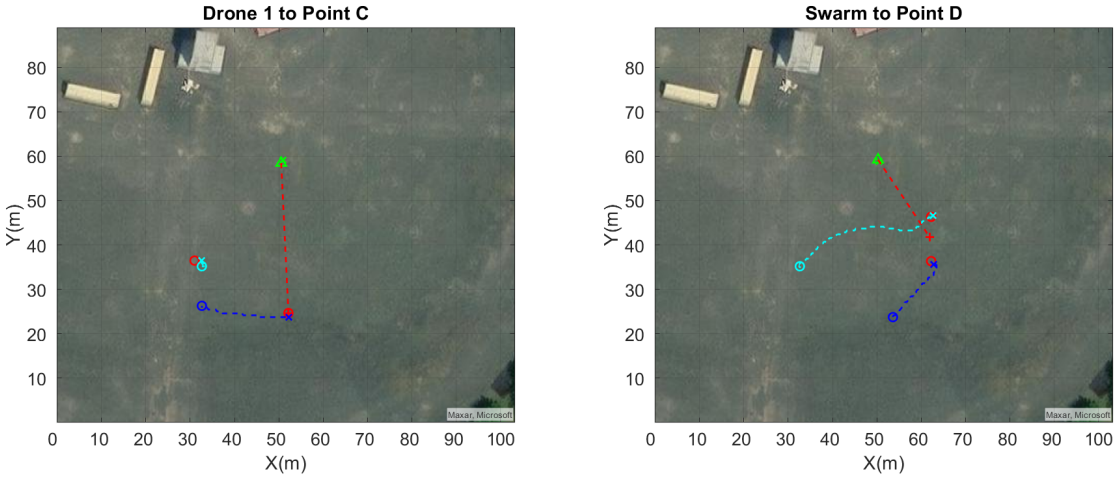


Figure 5.7: Smart binoculars repositioning agent 1 to point C and regrouping aerial swarm at point D

5.4 Conclusion

This work presents a novel method for providing a user with the capability to control the position of a multi-robot system outdoors. The rangefinder capabilities of the smart binoculars were used to obtain the targeted outdoor coordinates. The previously developed swarm velocity controller was utilized in conjunction with the task assignment system to designate tasks for the multi-agent systems to complete at their appointed locations.

Experimental results demonstrate that this system can successfully estimate the coordinates for a target location outdoors. The captured coordinates were relayed to a swarm of UAVs to display positional control of a multi-agent system. Ongoing and future work is focused on implementing voice-based control on the smart binoculars, as well as developing the ability for the smart binoculars to communicate directly with robotic systems as opposed to relying on a ground control stations to facilitate multi-agent control.

Chapter 6: Conclusion

Overall, this dissertation investigates human-robot interfaces for single and multi-agent systems. We explore positional and swarm state control as well as collision avoidance for indoor applications in confined spaces, and target localization and task assignment for outdoor multi-agent applications. We also studied pose and body parameter estimation to enable more intuitive interactions between users and robotic systems. These systems not only work to increase a user's control over robotic systems but enhance our interactions with them and expand our understanding of what these systems require for intuitive human-robot communication. This research was conducted using concepts from linear and nonlinear control theory, classical mechanics, autonomous robotics, human factors, and state estimation. Chapter 3 developed a cobot human swarm interface that prioritizes operator safety while reducing the cognitive load during control of a cobot swarm in a confined space. Experiments demonstrated that the developed interface enables position control of an aerial swarm as well as collision avoidance throughout the user's interactions with the swarm. Chapter 4 developed a method for estimating the pose as well as anthropometric measurements of a user. The estimated pose was used to estimate the location a user was pointing at, as an example application of this system's use in an intuitive human-robot interface. Chapter 5 demonstrated the use of the smart binoculars in selecting outdoor locations and assigning a multi-agent system with a task to complete at the targeted locations.

6.1 Summary of Contributions

This section summarizes the main contributions and results presented in this dissertation. First, I summarize the work presented in Chapter 3 and propose some ideas for future work in pose estimation. Second, I summarize the work in Chapter 4 and propose some future experiments with a diverse range of users to further validate the framework's usability for estimating pose and anthropometric measurements, as well as adapting this system to mobile platforms. Lastly, I summarize Chapter 5.

6.1.1 Safe Operations of an Aerial Swarm via a Cobot Human Swarm Interface

Command and control of an aerial swarm is a complex task which increases in difficulty when the flight volume is restricted, and the swarm and operator inhabit the same workspace. We present a novel human swarm interface (HSI) that utilizes gesture control and haptic feedback to interact with and control a swarm of quadrotors in a confined space. This human swarm interface prioritizes operator safety while reducing the cognitive load during control of an aerial swarm.

The presented control strategy utilized gestures to dynamically control the swarm's formation. The OYMotion gForcePro+, an electromyography (EMG) gesture recognition cuff described in Sec. 2.1.1, was trained on desired gestures shown in Fig. 3.2.2. These gestures were assigned commands that controlled the swarm's position, orientation, and density.

The swarm's formations were defined per the number of agents as shown in Fig. 3.1. While the formations were predefined, no agent was assigned a specific location within the formation. As the user performed gestures to alter the swarm's formation, the assignment problem is solved between the agents' initial (pre-gesture) and final (post-gesture) positions. The Munkres assign-

ment algorithm was used to minimize the distance each agent had to travel to complete the new formation.

A swarm velocity controller was developed to safely navigate agents from their initial to goal positions, while circumventing obstacles in their environments. The controller consists of an attractive potential enabling agents to move to their goal locations at a constant velocity, and an obstacle-avoidance potential allowing agents to safely avoid obstacles while moving towards their goal.

Haptic feedback was provided to the user through the bHaptcs TactSuit X40 shown in Sec. 2.1.3. The location of the swarm's center of mass was provided to the user by correlating its position in the environment to the motors on the haptic vest as shown in Fig. 3.3.

The results presented in Sec. 3.3.2 show that our HSI can successfully control the position, orientation, and density of a swarm in the global frame, or a frame fixed to the environment. Unfortunately, this methodology constrained our movements to a predefined discretized grid. One way to avoid that constraint was to base the control of the swarm on the position of the operator as opposed to a global frame.

The results presented in Sec. 3.3.3 demonstrate our system's reposition capabilities in the operator-frame. The user walks throughout the environment, points to a desired location, performs the required gesture, and the swarm successfully navigates to that location.

One major concern for cobot systems in a confined space is the safety of the operator during their interactions. That safety was enforced through our experiments through the developed swarm velocity controller. During the third experiment presented in Sec. 3.3.4, the user walked around and through an aerial swarm in flight. Fig. 3.6 and Fig. 3.7 show inter-agent and agent-operator distances. Throughout the experiment, the agents maintained their formation

while avoiding collision with the user.

While the proposed work demonstrates that an operator can safely and intuitively control a swarm of aerial robots in the same workspace, there is an underlying reliance on a motion capture system to provide the user's pose to the swarm. We next approached the problem of developing the capability for an autonomous system to estimate a user's pose to enable more intuitive interactions.

6.1.2 Multi-Sensor Pose and Parameter Estimation

A human-robot interface (HRI) is the mechanism by which humans and robots interact and communicate. The field of human-robot interactions explores these interfaces in an effort to optimize the utility of robots, while communicating the user's intentions intuitively. A key challenge in the field of human-robot interactions is the estimation of the user's pose, i.e., body position and orientation, and size. Motivated by the work presented in [69] and Chapter 3 which relies on a motion capture system to estimate user pose, this work describes a sensor-fusion framework that estimates the pose of a user as well as their torso length and chest width.

The presented estimation framework utilized visual sensors as well as acoustic sensors. A camera paired with MediaPipe Pose Landmarker, a computer vision software package described in Sec. 2.2.2, detected, identified, and tracked the user's body landmarks given a video feed. These landmarks, in conjunction with the pinhole camera model were used to develop a monocular depth estimation framework, assuming the system had prior knowledge of the user's torso length. Sec. 4.2.2 shows this derivation as well as how the depth was leveraged to estimate the position of the user. Two acoustic sensors were utilized to localize the user in the environment.

Each microphone array provided the system with a 3D unit vector in the direction of the detected sound source. Given the microphone arrays' positions, orientations, and their calculated directions of arrival, the 2D position (x,y) of the user was found as shown in Sec. 4.2.3.

Given a nonlinear system, an EKF linearizes the original nonlinear filter dynamics around the previous state estimates to estimate the current state of a nonlinear system given noisy measurements. In our work we added four bias states to the state vector to estimate the constant error offset in measurements as well as user parameters; two for the direction of arrival measurements and two to estimate the user's body lengths. Bias terms are often used to estimate measurement deviations from expected values. The novelty in this work lies in the definition of the bias terms themselves. By setting nominally chosen values of the torso length and chest width as the expected values, and the measurement deviations of the user's body lengths as the biases, the user's body lengths can be approximated by subtracting the estimated bias from the expected values. Thus, the EKF provides a method to estimate the user chest width and torso length, in conjunction with user pose. The dynamics and filter equations of the proposed EKF are shown in Sec. 4.3.

Results from vision-only and sound-only position estimation experiments show that both systems accurately estimate the position of the user independently. One drawback to the vision-only system is that the position estimation capabilities are limited to the camera's viewing angle as shown in Fig. 4.5. While the sound-only system is not limited to any viewing angle as shown in Fig. 4.5, since it estimates the 2D position (x, y), it does not provide the system with an orientation ψ estimate like the vision-based system does. By fusing these two systems, we can accurately track the pose of a user in our described environment. Fig. 4.7 presents a 5-minute experiment where the position (x, y), orientation ψ, and speed s of a user was tracked as they

walk throughout the environment performing various maneuvers. The estimated biases for the microphones and user parameters are shown in Fig. 4.8. Given the mean estimated biases, we show that this system successfully estimates the torso length and chest width of the user. This experiment demonstrates that the sensor fusion EKF can be used to estimate the pose as well user parameters online.

Next, we demonstrate how the estimated user parameters could be used to develop an intuitive human-robot interface. The user is tasked with pointing at a white bucket in the environment and the location the user is pointing at is estimated. Fig. 4.9 and Fig. 4.10 show an image feed provided to the system as well as the 3D reconstruction of the system estimating the ground point. This experiment demonstrates that the proposed system can be used to estimate where the user is pointing, potentially providing contextual information to robotic system about a user's intention.

6.1.3 Outdoor Target Selection-based Human Swarm Interface

While human swarm interfaces (HSIs) and human-robot interfaces (HRIs) have been extensively researched for indoor laboratory environments, their application in outdoor settings remain largely unexplored. Unstructured terrain and varying environmental conditions introduce numerous external factors that impact the usability and effectiveness of these systems. Another challenge is designing these systems to be mobile and easily used by a single operator. We present the smart binoculars, a portable device designed to facilitate interactions between users and multi-agent autonomous systems, particularly in outdoor and dynamic environments. Given a location of interest, the smart binoculars will allow a user to send autonomous systems to those locations, within line of sight, to complete desired tasks.

The presented system utilizes an onboard GPS module to localize the user, and a rangefinder to measure the distance to a location the user is targeting. Once the user's coordinates and range to target are captured, an onboard 9-DoF IMU is used to estimate the orientation and heading of the smart binoculars. Given the range, user's coordinates, and magnetic heading, the terminal coordinates for a target are estimated as described in Sec. 5.2.2. The smart binoculars use mechanical switches to provide the user with a method to select desired locations and a Doodle Mini to communicate with the autonomous systems and ground control station.

The smart binoculars will provide an interface for users in the field to interact with mobile robotic systems for line-of-sight operations. Given a location of interest, users will target the location and record the coordinates. Once the desired number of coordinates are recorded, they are sent to the ground control station to either send to autonomous agents as waypoints or define a search area. Sec. 5.2.3 provides examples of these engagements.

The smart binoculars have evolved in both functionality and capability as discussed in Sec. 5.2.4. Initially, the SB-V1 provided target localization capabilities and demonstrated the feasibility of a mobile outdoor human-swarm interface (HSI). The SB-V2 introduced mechanical levers that allowed users to select and record waypoints, enabling dynamic UAV repositioning during operation. The SB-V3 further enhanced performance with an upgraded rangefinder and transitioning to Doodle Radios for improved communication and integration into multi-agent mesh networks. The SB-V4 maintained the overall design and structure of the SB-V3 but utilized higher quality sensors to improve overall performance.

The results presented in Sec. 5.3 demonstrate the smart binoculars' ability to target desired locations and control the positions of a multi-agent systems. Fig. 5.6 shows that an operator can effectively control a swarm's position while moving outdoors using the smart binoculars. Fig. 5.7

shows that an operator can effectively control the position of both an individual agent as well as a swarm's position, giving the user the ability to split and regroup multi-agent systems for a variety of tasks.

6.2 Ongoing and Future Work

As autonomous systems become more engrained in our society, studying how users interact with these systems and how these systems perceive and comprehend a user's intentions has become a field of study in its own right. The Cobot HSI enabled an operator to safely and intuitively control a swarm of aerial robots using gestures control, while occupying the same workspace. Future work aims to adapt this HSI to more computationally capable autonomous systems, enabling applications beyond controlled laboratory environments.

The current dependence on a motion capture system also motivated the development of an HRI framework that does not rely on experimental infrastructure to facilitate human-robot interactions. We developed an HRI that estimated both the pose and anthropometric measurements of a user. While the proposed framework's usability was demonstrated through experimentation, future work is focused on conducting experiments with a diverse range of users to validate this framework's utility for estimating pose and anthropometric measurements for a wide array of individuals. Another area of interest is in adapting this framework to mobile platforms, allowing autonomous systems to utilize this capability to enable new forms of interactions, furthering the field of HRI.

The smart binoculars provide users the ability to command and control outdoor multi-agent systems, without compromising their situational awareness, empowering them to leverage their

on-the-ground knowledge for improved decision-making. Future efforts will focus on developing more fluid methods for the user to intact with the smart binoculars, streamlining overall operational efficiency. One method is using the operator's voice to capture a point or designate a task to the system the user is interacting with. Another area of interest is eliminating the need for a ground control station all together and developing more intuitive methods for the user and autonomous systems to interact in these diverse outdoor environment.

Appendix A: Vision-based System's Observation Model

A.1 Vision-based System's Observation Model

Observed Heading: Given the 3D landmarks of the shoulders and hips ($\boldsymbol{l}_{11}, \boldsymbol{l}_{12}, \boldsymbol{l}_{23}, \boldsymbol{l}_{24}$) from MediaPipe, the heading is calculated directly as shown below:

$$\boldsymbol{l}_s = \frac{1}{2}\left(\boldsymbol{l}_{11} + \boldsymbol{l}_{12}\right), \quad \boldsymbol{l}_h = \frac{1}{2}\left(\boldsymbol{l}_{23} + \boldsymbol{l}_{24}\right) \tag{A.1}$$

$$\psi_{chest} = \tan^{-}1\left(\frac{w_{11} - w_s}{u_{11} - u_s}\right) \tag{A.2}$$

$$\psi_{hips} = \tan^{-}1\left(\frac{w_{23} - w_h}{u_{23} - u_h}\right) \tag{A.3}$$

$$\psi = \frac{1}{2}\left(\psi_{chest} + \psi_{hips}\right) \tag{A.4}$$

Transforming Position Estimate From Inertial to Camera Frame: Given a position estimate of the user in the inertial frame $\boldsymbol{r}_{U/O_I} = (x, y, z)$, the position estimate in the camera frame $\boldsymbol{r}_{U/O_C} = (x_U, y_U, z_U)$ is:

$$\boldsymbol{r}_{U/O_C} = {}^{C}\boldsymbol{R}^{I}\left(\boldsymbol{r}_{U/O_I} - \boldsymbol{r}_{O_C/O_I}\right) \tag{A.5}$$

$$x_U = {}^C\boldsymbol{R}^I{}_{(1,1:3)} \left(\boldsymbol{r}_{U/O_I} - \boldsymbol{r}_{O_C/O_I}\right) \tag{A.6}$$

$$y_U = {}^C\boldsymbol{R}^I{}_{(2,1:3)} \left(\boldsymbol{r}_{U/O_I} - \boldsymbol{r}_{O_C/O_I}\right) \tag{A.7}$$

$$z_U = {}^C\boldsymbol{R}^I{}_{(3,1:3)} \left(\boldsymbol{r}_{U/O_I} - \boldsymbol{r}_{O_C/O_I}\right) \tag{A.8}$$

$$\begin{aligned} x_U = & {}^C\boldsymbol{R}^I{}_{(1,1)} \left(x - x_{O_C/O_I}\right) + \\ & {}^C\boldsymbol{R}^I{}_{(1,2)} \left(y - y_{O_C/O_I}\right) + \\ & {}^C\boldsymbol{R}^I{}_{(1,3)} \left(z - z_{O_C/O_I}\right) \end{aligned} \tag{A.9}$$

y_U and z_U can be solved for explicitly in a similar fashion. This position is the position of the user's hips in the camera frame.

Finding Landmarker World Position Estimate in Camera Frame: Given the position of the user's hips in the camera frame $\boldsymbol{r}_{U/O_C} = (x_U, y_U, z_U)$, the position of the hip and shoulder landmarks in the camera frame ($\boldsymbol{L}_{11}, \boldsymbol{L}_{12}, \boldsymbol{L}_{23}, \boldsymbol{L}_{24}$) can be solved for as shown below:

$$x_n = x_U \pm \frac{1}{2} BL \cos\left(\psi\right) \tag{A.10}$$

$$y_n = y_U - BL \tag{A.11}$$

$$z_n = z_U \tag{A.12}$$

where Tab. A.1 shows the relevant body lengths (BL) used to solve for the two shoulder landmarks (11 and 12), and the two hip landmarks (23 and 24).

Finding Landmark Position Estimate in Image Plane: Once all landmarks have been

n	u_n	v_n
11	$CW + \Delta_4$	$-(TL + \Delta_3)$
12	$-(CW + \Delta_4)$	$-(TL + \Delta_3)$
23	HW	0
24	$-(HW)$	0

Table A.1: Body lengths used to solve for the landmark positions in the camera frame.

solved for, the Pinhole Camera Model (Eqn. 2.2) is used to explicitly solve for the estimated locations of the landmarks in the image frame as shown below:

$$u_{11} = f_x \left(\frac{x_U + \frac{1}{2}(CW + \Delta_4)\cos(\psi)}{z_U} \right) + u_0 \tag{A.13}$$

$$v_{11} = f_y \left(\frac{y_U - (TL + \Delta_3)}{z_U} \right) + v_0 \tag{A.14}$$

$$u_{12} = f_x \left(\frac{x_U - \frac{1}{2}(CW + \Delta_4)\cos(\psi)}{z_U} \right) + u_0 \tag{A.15}$$

$$v_{12} = f_y \left(\frac{y_U - (TL + \Delta_3)}{z_U} \right) + v_0 \tag{A.16}$$

$$u_{23} = f_x \left(\frac{x_U + \frac{1}{2}HW}{z_U} \right) + u_0 \tag{A.17}$$

$$v_{23} = f_y \left(\frac{y_U}{z_U} \right) + v_0 \tag{A.18}$$

$$u_{24} = f_x \left(\frac{x_U - \frac{1}{2}HW}{z_U} \right) + u_0 \tag{A.19}$$

$$v_{24} = f_y \left(\frac{y_U}{z_U} \right) + v_0 \tag{A.20}$$

Bibliography

[1] Julia Berg and Shuang Lu. Review of interfaces for industrial human-robot interaction. *Current Robotics Reports*, 1:27–34, 2020.

[2] Peter Birkenkampf, Daniel Leidner, and Christoph Borst. A knowledge-driven shared autonomy human-robot interface for tablet computers. In *2014 IEEE-RAS International Conference on Humanoid Robots*, pages 152–159. IEEE, 2014.

[3] Shuwen Qiu, Hangxin Liu, Zeyu Zhang, Yixin Zhu, and Song-Chun Zhu. Human-robot interaction in a shared augmented reality workspace. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 11413–11418. IEEE, 2020.

[4] Christian Fleischer and Günter Hommel. A human–exoskeleton interface utilizing electromyography. *IEEE Transactions on Robotics*, 24(4):872–882, 2008.

[5] Paul Dempsey. The teardown: Amazon astro consumer robot. *Engineering & Technology*, 18(2):70–71, 2023.

[6] Carlos Carbone, Oscar Garibaldi, and Zohre Kurt. Swarm robotics as a solution to crops inspection for precision agriculture. *KnE Engineering*, 3:552–562, 2 2018.

[7] Jianing Chen, Melvin Gauci, Wei Li, Andreas Kolling, and Roderich Groß. Occlusion-based cooperative transport with a swarm of mobile robots. *IEEE Transactions on Robotics*, 31:307–321, 4 2015.

[8] Gustavo A. Cardona, Juan Ramirez-Rugeles, Eduardo Mojica-Nava, and Juan M. Calderon. Visual victim detection and quadrotor-swarm coordination control in search and rescue environment. *International Journal of Electrical and Computer Engineering*, 11:2079–2089, 6 2021.

[9] Gabriel Quiroz and Si Jung Kim. A confetti drone: Exploring drone entertainment. In *2017 IEEE International Conference on Consumer Electronics (ICCE)*, pages 378–381. IEEE, 3 2017.

[10] Markos Sigalas, Haris Baltzakis, and P Trahanias. Gesture recognition based on arm tracking for human-robot interaction. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5424–5429. IEEE, 12 2010.

[11] Mohamed A. Kassab, Mostafa Ahmed, Ali Maher, and Baochang Zhang. Real-time human-uav interaction: New dataset and two novel gesture-based interacting systems. *IEEE Access*, 8:195030–195045, 10 2020.

[12] Jawad Nagi, Frederick Ducatelle, Gianni A. Di Caro, Dan Cireşan, Ueli Meier, Alessandro Giusti, Farrukh Nagi, Jurgen Schmidhuber, and Luca Maria Gambardella. Max-pooling convolutional neural networks for vision-based hand gesture recognition. In *2011 IEEE International Conference on Signal and Image Processing Applications (ICSIP)*, pages 342–347. IEEE, 11 2011.

[13] Jawad Nagi, Alessandro Giusti, Luca M. Gambardella, and Gianni A. Di Caro. Human-swarm interaction using spatial gestures. In *2014 IEEE International Conference on Intelligent Robots and Systems (IROS)*, pages 3834–3841. IEEE, 9 2014.

[14] Akash Chaudhary, Tiago Nascimento, and Martin Saska. Controlling a swarm of unmanned aerial vehicles using full-body k-nearest neighbor based action classifier. In *2022 International Conference on Unmanned Aircraft Systems, (ICUAS)*, pages 544–551. IEEE, 6 2022.

[15] Boris Gromov, Jerome Guzzi, Luca M. Gambardella, and Alessandro Giusti. Intuitive 3d control of a quadrotor in user proximity with pointing gestures. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5964–5971. IEEE, 5 2020.

[16] Aamodh Suresh and Sonia Martínez. Human-swarm interactions for formation control using interpreters. *International Journal of Control, Automation and Systems*, 18:2131–2144, 8 2020.

[17] Sasanka Nagavalli, Meghan Chandarana, Katia Sycara, and Michael Lewis. Multi-operator gesture control of robotic swarms using wearable devices. In *Proceedings of the Tenth International Conference on Advances in Computer-Human Interactions*, pages 25–33. IEEE, 2017.

[18] Joseph DelPreto and Daniela Rus. Plug-and-play gesture control using muscle and motion sensors. In *Proceedings of the 2020 ACM/IEEE International Conference on Human-Robot Interaction*, pages 439–448, 3 2020.

[19] Boris Gromov, Luca M. Gambardella, and Gianni A. Di Caro. Wearable multi-modal interface for human multi-robot interaction. In *2016 IEEE International Symposium on Safety, Security and Rescue Robotics (SSRR)*, pages 240–245. IEEE, 10 2016.

[20] Ludger Overmeyer, Florian Podszus, and Lars Dohrmann. Multimodal speech and gesture control of agvs, including eeg-based measurements of cognitive workload. *CIRP Annals - Manufacturing Technology*, 65:425–428, 1 2016.

[21] Mingxuan Chen, Ping Zhang, Zebo Wu, and Xiaodan Chen. A multichannel human-swarm robot interaction system in augmented reality. *Virtual Reality and Intelligent Hardware*, 2:518–533, 12 2020.

[22] Dongjun Lee, Antonio Franchi, Hyoung Il Son, Changsu Ha, Heinrich H. Bulthoff, and Paolo Robuffo Giordano. Semiautonomous haptic teleoperation control architecture of multiple unmanned aerial vehicles. *IEEE/ASME Transactions on Mechatronics*, 18:1334–1345, 5 2013.

[23] Hyoung Il Son, Lewis L Chuang, Junsuk Kim, and Heinrich H Bülthoff. Haptic feedback cues can improve human perceptual awareness in multi-robots teleoperation. In *2011 International Conference on Control, Automation and Systems (ICCAS)*, pages 1323–1328. IEEE, 10 2011.

[24] Matteo MacChini, Thomas Havy, Antoine Weber, Fabrizio Schiano, and Dario Floreano. Hand-worn haptic interface for drone teleoperation. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 10212–10218, 5 2020.

[25] Evgeny Tsykunov, Ruslan Agishev, Roman Ibrahimov, Luiza Labazanova, Akerke Tleugazy, and Dzmitry Tsetserukou. Swarmtouch: Guiding a swarm of micro-quadrotors with impedance control using a wearable tactile interface. *IEEE Transactions on Haptics*, 12:363–374, 7 2019.

[26] Qiangqiang Ouyang, Juan Wu, and Miao Wu. Vibrotactile display of flight attitude with combination of multiple coding parameters. *Applied Sciences*, 7, 12 2017.

[27] Naoki Shibata, Seiji Sugiyama, and Takahiro Wada. Collision avoidance control with steering using velocity potential field. In *Proceedings of the 2014 IEEE Intelligent Vehicles Symposium*, pages 438–443. IEEE, 6 2014.

[28] Michael T. Wolf and Joel W. Burdick. Artificial potential functions for highway driving with collision avoidance. *Proceedings - IEEE International Conference on Robotics and Automation*, pages 3731–3736, 2008.

[29] Hu Hongyu, Zhang Chi, Sheng Yuhuan, Zhou Bin, and Gao Fei. An improved artificial potential field model considering vehicle velocity for autonomous driving. *IFAC-PapersOnLine*, 51(31):863–867, 1 2018.

[30] Yongjie Yan and Yan Zhang. Collision avoidance planning in multi-robot based on improved artificial potential field and rules. In *Proceedings of the 2008 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pages 1026–1031. IEEE, 2 2009.

[31] Jiayi Sun, Jun Tang, and Songyang Lao. Collision avoidance for cooperative uavs with optimized artificial potential field algorithm. *IEEE Access*, 5:18382–18390, 8 2017.

[32] Derek J. Bennet and Colin R. McInnes. Distributed control of multi-robot systems using bifurcating potential fields. *Robotics and Autonomous Systems*, 58:256–264, 3 2010.

[33] Heba Gaber, Safaa Amin, and Abdel-Badeeh M Salem. A combined coordination technique for multi-agent path planning. In *2010 International Conference on Intelligent Systems Design and Applications*, pages 563–568. IEEE, 11 2010.

[34] Yousef Emam, Paul Glotfelter, and Magnus Egerstedt. Robust barrier functions for a fully autonomous, remotely accessible swarm-robotics testbed. In *Proceedings of the 2019 IEEE Conference on Decision and Control (CDC)*, pages 3984–3990. IEEE, 12 2019.

[35] Andrei Zanfir, Mihai Zanfir, Alex Gorban, Jingwei Ji, Yin Zhou, Dragomir Anguelov, and Cristian Sminchisescu. Hum3dil: Semi-supervised multi-modal 3D humanpose estimation for autonomous driving. In *Conference on Robot Learning*, pages 1114–1124, 2023.

[36] Diogo C Luvizon, David Picard, and Hedi Tabia. Multi-task deep learning for real-time 3D human pose estimation and action recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(8):2752–2764, 2020.

[37] Dushyant Mehta, Srinath Sridhar, Oleksandr Sotnychenko, Helge Rhodin, Mohammad Shafiei, Hans-Peter Seidel, Weipeng Xu, Dan Casas, and Christian Theobalt. Vnect: Real-time 3D human pose estimation with a single RGB camera. *ACM transactions on graphics*, 36(4):1–14, 2017.

[38] Pouya Jafarzadeh, Petra Virjonen, Paavo Nevalainen, Fahimeh Farahnakian, and Jukka Heikkonen. Pose estimation of hurdles athletes using OpenPose. In *2021 International Conference on Electrical, Computer, Communications and Mechatronics Engineering*, pages 1–6, 2021.

[39] Chunyu Wang, Yizhou Wang, Zhouchen Lin, Alan L Yuille, and Wen Gao. Robust estimation of 3D human poses from a single image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2361–2368, 2014.

[40] Carlos Barron and Ioannis A Kakadiaris. Estimating anthropometry and pose from a single image. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition.*, volume 1, pages 669–676, 2000.

[41] Camillo J Taylor. Reconstruction of articulated objects from point correspondences in a single uncalibrated image. *Computer Vision and Image Understanding*, 80(3):349–363, 2000.

[42] Federica Bogo, Angjoo Kanazawa, Christoph Lassner, Peter Gehler, Javier Romero, and Michael J Black. Keep it SMPL: Automatic estimation of 3D human pose and shape from a single image. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part V 14*, pages 561–578, 2016.

[43] Hashim Yasin, Umar Iqbal, Bjorn Kruger, Andreas Weber, and Juergen Gall. A dual-source approach for 3D pose estimation from a single image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4948–4956, 2016.

[44] Dylan Drover, Rohith MV, Ching-Hang Chen, Amit Agrawal, Ambrish Tyagi, and Cong Phuoc Huynh. Can 3D pose be learned from 2D projections alone? In *Proceedings of the European Conference on Computer Vision Workshops*, pages 0–0, 2018.

[45] Ching-Hang Chen and Deva Ramanan. 3D human pose estimation = 2D pose estimation + matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern recognition*, pages 7035–7043, 2017.

[46] Gines Hidalgo, Yaadhav Raaj, Haroon Idrees, Donglai Xiang, Hanbyul Joo, Tomas Simon, and Yaser Sheikh. Single-network whole-body pose estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6982–6991, 2019.

[47] Sven Kreiss, Lorenzo Bertoni, and Alexandre Alahi. PifPaf: Composite fields for human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11977–11986, 2019.

[48] Alexander Weiss, David Hirshberg, and Michael J Black. Home 3D body scans from noisy image and range data. In *2011 International Conference on Computer Vision*, pages 1951–1958. IEEE, 2011.

[49] Huanghao Xu, Yao Yu, Yu Zhou, Yang Li, and Sidan Du. Measuring accurate body parameters of dressed humans with large-scale motion using a Kinect sensor. *Sensors*, 13(9):11362–11384, 2013.

[50] Xu Yang Gan, Haidi Ibrahim, and Dzati Athiar Ramli. A simple vision based anthropometric estimation system using webcam. In *Journal of Physics: Conference Series*, volume 1529, page 022067, 2020.

[51] Bernard Friedland. Treatment of bias in recursive filtering. *IEEE Transactions on Automatic Control*, 14(4):359–367, 1969.

[52] Andrew H Jazwinski. *Stochastic Processes and Filtering Theory*. Courier Corporation, 2007.

[53] Ross Hartley, Maani Ghaffari, Ryan M Eustice, and Jessy W Grizzle. Contact-aided invariant extended Kalman filtering for robot state estimation. *The International Journal of Robotics Research*, 39(4):402–430, 2020.

[54] David M Bevly and Bradford Parkinson. Cascaded Kalman filters for accurate estimation of multiple biases, dead-reckoning navigation, and full state feedback control of ground vehicles. *IEEE Transactions on Control Systems Technology*, 15(2):199–208, 2007.

[55] Qgroundcontrol. https://qgroundcontrol.com/, 2023.

[56] Mission planner. https://ardupilot.org/planner/, 2023.

[57] CH Wu, SH Tu, SW Tu, LH Wang, and WH Chen. Realization of remote monitoring and navigation system for multiple uav swarm missions: Using 4g/wifi-mesh communications and rtk gps positioning technology. In *2022 International Automatic Control Conference (CACS)*, pages 1–6. IEEE, 2022.

[58] Youkyung Hong, Sunggoo Jung, Suseong Kim, and Jihun Cha. Autonomous mission of multi-uav for optimal area coverage. *Sensors*, 21(7):2482, 2021.

[59] Nitesh Kumar and Mangal Kothari. Development of a custom ground control station for unmanned aerial vehicle swarming. In *AIAA SCITECH 2024 Forum*, page 0748, 2024.

[60] Rune Hylsberg Jacobsen, Lea Matlekovic, Liping Shi, Nicolaj Malle, Naeem Ayoub, Kaspar Hageman, Simon Hansen, Frederik Falk Nyboe, and Emad Ebeid. Design of an autonomous cooperative drone swarm for inspections of safety critical infrastructure. *Applied Sciences*, 13(3):1256, 2023.

[61] Michael Walker, Zhaozhong Chen, Matthew Whitlock, David Blair, Danielle Albers Szafir, Christoffer Heckman, and Daniel Szafir. A mixed reality supervision and telepresence interface for outdoor field robotics. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2345–2352. IEEE, 2021.

[62] Chen Zhao, Chuanqi Zheng, Leah Roldan, Thomas Shkurti, Ammar Nahari, Wyatt S Newman, Dustin J Tyler, Kiju Lee, and Michael J Fu. Adaptable mixed-reality sensorimotor interface for human-swarm teaming: Person with limb loss case study and field experiments. *Field Robotics*, 3(1):243–265, 2023.

[63] Mingxuan Chen, Ping Zhang, Zebo Wu, and Xiaodan Chen. A multichannel human-swarm robot interaction system in augmented reality. *Virtual Reality & Intelligent Hardware*, 2(6):518–533, 2020.

[64] Civtak/atak. https://www.civtak.org/home/, 2023.

[65] Jinho Kim, Tim Gregory, Jade Freeman, and Christopher M Korpela. System-of-systems for remote situational awareness: Integrating unattended ground sensor systems with autonomous unmanned aerial system and android team awareness kit. In *2022 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 1418–1423. IEEE, 2022.

[66] Dominic Larkin, Michael Novitzky, Jinho Kim, and Christopher M Korpela. Atak integration through ros for autonomous air-ground team. In *2021 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 1116–1122. IEEE, 2021.

[67] Andrew Kopeikin, Conner Russell, Hayden Trainor, Ashley Rivera, Tyrus Jones, Benjamin Baumgartner, Pratheek Manjunath, Samuel Heider, Thomas Surdu, and Matthew Galea. Designing and flight-testing a swarm of small uas to assist post-nuclear blast forensics. In *2020 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 466–472. IEEE, 2020.

[68] Michael Lichtenstern, Michael Angermann, and Martin Frassl. Imu-and gnss-assisted single-user control of a mav-swarm for multiple perspective observation of outdoor activities. In *Proceedings of the 2011 International Technical Meeting of The Institute of Navigation*, pages 1062–1069, 2011.

[69] Sydrak S Abdi and Derek A Paley. Safe Operations of an Aerial Swarm via a Cobot Human Swarm Interface. In *2023 IEEE International Conference on Robotics and Automation*, pages 1701–1707. IEEE, 2023.

[70] Xu Zhang, Xiang Chen, Yun Li, Vuokko Lantz, Kongqiao Wang, and Jihai Yang. A framework for hand gesture recognition based on accelerometer and emg sensors. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 41:1064–1076, 3 2011.

[71] gforcepro+ emg armband. http://www.oymotion.com/en/product32/149, 2021.

[72] Loco swarm bundle. https://store.bitcraze.io/products/the-swarm-bundle, 2021.

[73] Crazyflie 2.1. https://store.bitcraze.io/products/crazyflie-2-1, 2021.

[74] Loco positioning system. https://www.bitcraze.io/documentation/system/positioning/loco-positioning-system/, 2021.

[75] Loco positioning node. https://store.bitcraze.io/products/loco-positioning-node, 2021.

[76] Loco positioning deck. https://store.bitcraze.io/products/loco-positioning-deck, 2021.

[77] James A. Preiss, Wolfgang Honig, Gaurav S. Sukhatme, and Nora Ayanian. Crazyswarm: A large nano-quadcopter swarm. In *Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3299–3304. IEEE, 7 2017.

[78] https://www.bhaptics.com/tactsuit/tactsuit-x40, 2021.

[79] Richard Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge university press, 2003.

[80] Peter I Corke, Witold Jachimczyk, and Remo Pillat. *Robotics, Vision and Control: Fundamental Algorithms in MATLAB*, volume 73. Springer, 2011.

[81] Alexander Hornberg. *Handbook of Machine Vision*. John Wiley & Sons, 2006.

[82] Mediapipe solutions guide. https://developers.google.com/mediapipe/solutions/guide, 2022.

[83] MediaPipe. Mediapipe overview talk - google berlin 11 dec 2019. https://www.youtube.com/watch?v=YoVPBLgE3lQ, Jan. 2020.

[84] Pose landmark detection guide. https://developers.google.com/mediapipe/solutions/vision/pose-landmarker, 2022.

[85] UMA-8 USB mic array - V2.0. https://www.minidsp.com/products/usb-audio-interface/uma-8-microphone-array, 2022.

[86] François Grondin, Dominic Létourneau, Cédric Godin, Jean-Samuel Lauzon, Jonathan Vincent, Simon Michaud, Samuel Faucher, and François Michaud. ODAS: Open embeddeD Audition System. *Frontiers in Robotics and AI*, 9, 2022.

[87] Daniel Hernández, José M Cecília, Carlos T Calafate, Juan-Carlos Cano, and Pietro Manzoni. The kuhn-munkres algorithm for efficient vertical takeoff of uav swarms. In *2021 IEEE Vehicular Technology Conference (VTC2021-Spring)*, pages 1–5. IEEE, 4 2021.

[88] Zhen Zhang, Kuo Yang, Jinwu Qian, and Lunwei Zhang. Real-time surface emg pattern recognition for hand gestures based on an artificial neural network. *Sensors*, 19:1–16, 7 2019.

[89] Peter B. Shull, Shuo Jiang, Yuhui Zhu, and Xiangyang Zhu. Hand gesture recognition and finger angle estimation via wrist-worn modified barometric pressure sensing. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 27:724–732, 4 2019.

[90] Oussama Khatib. Real-time obstacle avoidance for manipulators and mobile robots. *The International Journal of Robotics Research*, 5:90–98, 3 1986.

[91] J.L. Crassidis and J.L. Junkins. *Optimal Estimation of Dynamic Systems, Second Edition*. Chapman & Hall/CRC Applied Mathematics & Nonlinear Science. Taylor & Francis, 2011.

[92] Maria Isabel Ribeiro. Kalman and Extended Kalman Filters: Concept, Derivation and Properties. *Institute for Systems and Robotics*, 43(46):3736–3741, 2004.

[93] Holybro h-rtk f9p helical. https://holybro.com/products/h-rtk-f9p-gnss-series?variant=41466787168445, 2023.

[94] Adafruit ism330dhcx + lis3mdl featherwing - high precision 9-dof imu. https://www.adafruit.com/product/4569, 2023.

[95] Raspberry pi 4. https://www.raspberrypi.com/products/raspberry-pi-4-model-b/, 2023.

[96] Mini mesh rider radio. https://doodlelabs.com/products/mesh-rider-radios/mini/, 2023.

[97] Voxl2. https://www.modalai.com/products/voxl-2?variant=39914779836467, 2023.

[98] M500. https://docs.modalai.com/m500/, 2023.