frontiers | Frontiers in Robotics and AI

ORIGINAL RESEARCH
published: 07 July 2022
doi: 10.3389/frobt.2022.915884

Bio-Inspired Vision and Gesture-Based Robot-Robot Interaction for Human-Cooperative Package Delivery

Kaustubh Joshi[1] *and Abhra Roy Chowdhury*[2]*

[1]*Department of Mechanical Engineering, University of Maryland, College Park, MD, United States,* [2]*Centre for Product Design and Manufacturing, Division of Mechanical Engineering, Indian Institute of Science (IISc), Bangalore, India*

This research presents a novel bio-inspired framework for two robots interacting together for a cooperative package delivery task with a human-in the-loop. It contributes to eliminating the need for network-based robot-robot interaction in constrained environments. An individual robot is instructed to move in specific shapes with a particular orientation at a certain speed for the other robot to infer using object detection (custom YOLOv4) and depth perception. The shape is identified by calculating the area occupied by the detected polygonal route. A metric for the area's extent is calculated and empirically used to assign regions for specific shapes and gives an overall accuracy of 93.3% in simulations and 90% in a physical setup. Additionally, gestures are analyzed for their accuracy of intended direction, distance, and the target coordinates in the map. The system gives an average positional RMSE of 0.349 in simulation and 0.461 in a physical experiment. A video demonstration of the problem statement along with the simulations and experiments for real world applications has been given here and in Supplementary Material.

Keywords: bio-inspired multi-robot interaction, RGB-D perception, vision based gestural interaction, human-robot cooperation, passive action recognition

OPEN ACCESS

Edited by:
Luis J. Manso,
Aston University, United Kingdom

Reviewed by:
Yong-Guk Kim,
Sejong University, South Korea
Rocco Furferi,
University of Florence, Italy

****Correspondence:***
Abhra Roy Chowdhury
abhra@iisc.ac.in

Specialty section:
This article was submitted to Robot and Machine Vision, a section of the journal Frontiers in Robotics and AI

Received: *08 April 2022*
Accepted: *09 June 2022*
Published: *07 July 2022*

Citation:
Joshi K and Roy Chowdhury A (2022) Bio-Inspired Vision and Gesture-Based Robot-Robot Interaction for Human-Cooperative Package Delivery. Front. Robot. AI 9:915884. doi: 10.3389/frobt.2022.915884

1 INTRODUCTION

Humans are adept at using audio and visual cues for communication while carrying out collaborative tasks. However, humans must rely entirely on non-verbal communication like visual gestures to coordinate in a noisy environment. This research aims towards implementing a similar ability to use gestural interaction in a networked system of robots. Traditional methods of robot communication in a multi-robot system rely heavily on network connections through communication protocols (Yan et al., 2013). What if a system of robots had to be deployed in an area where there was a lack of network resources? In such a scenario, robots will have to rely on other sensors for interaction. In this paper, two robots are used to demonstrate a vision-based gestural interaction framework to carry out a task of package handling in cooperation with a human (Wang and Schwager, 2015). This approach enables each robot to be independent of a centralized controller or server. A system of two different robots demonstrates that the framework is scalable to incorporate different types of robots in the system. The system consists of the following types of robots: (1) package handling robots capable of carrying packages and detecting both human and robot gestures; (2) messenger robots for conveying information to package handling robots and carry out supervision and can detect only human gestures. The messenger robots read a human gesture using skeletal tracking for knowing the package

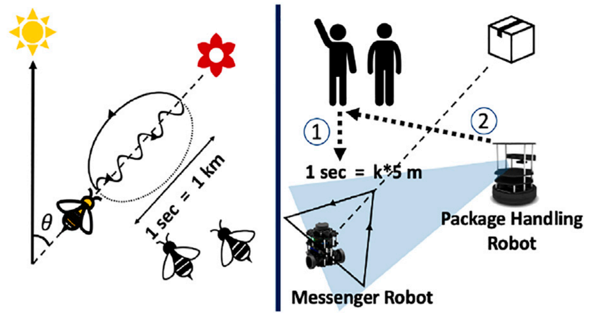


FIGURE 1 | An analogical example of how bees communicate to convey distance and direction of food source. The framework used in this paper involves a robot that moves in a triangle to convey direction and distance towards the target destination for the other robot.in cooperation with humans.

to be transported and signal a particular package handling robot to carry out the task. The package handling robot is capable of object detection and tracking by which it can infer the trajectory of a messenger robot depicting a particular action. This mode of bio-inspired passive action recognition (Das et al., 2016) is based on how bees communicate with other bees in their hive to convey the distance and direction of a food source about the hive's location and position of the Sun using a gesture-based communication called 'waggle dance' (Riley et al., 2005) (See **Figure 1**). Waggle dance has been used in for studying pattern formation and recognition (Esch and Burns, 1996). Recent studies have used it for robot interaction with live bees (Landgraf et al., 2018). In multi-robot systems, "waggle dance" inspired interaction has been used to identify commands in the form of gestures from other robots (Das et al., 2016). This paper implements a novel framework for interaction as well as determining the distance and direction through such gestures. Past work in vision-based multi robot interaction (Kuniyoshi et al., 1994) has mainly focused on limited explicit communication (Meng et al., 2006) and marker-based communication (Parker et al., 2004) or colour-based interaction (Melanie et al., 2020). This research adds a gestural interaction framework to the existing literature. Gesture based interaction involves object detection and tracking for which various algorithms have been developed in the past. The major object detection algorithms include the development of YOLO (You Only Look Once) (Redmon et al., 2016), (Shafiee et al., 2017), (Redmon and Farhadi, 2018) series as well as SSD (Single Shot Detection) (Liu et al., 2016), (Zhai et al., 2020) frameworks. Object tracking involves the usage of measuring the distance of that particular object from the camera. This has been done using a monocular camera (Crivellaro et al., 2017), (Tjaden et al., 2018) using machine learning algorithms, or a stereo camera (Lin and Wang, 2010), (Issac et al., 2016) using triangulation methods as well as other machine learning methodologies. The current paper uses a depth camera (Lukezic et al., 2019) where the IR sensor in the depth camera is used to measure the distance of the object from the camera once it has been detected. Object tracking has various methodologies being used in the past like centroid detection and tracking (Nascimento et al., 1999), average depth estimation (Chu et al., 2017) and other methods.

Additionally, network servers are prone to damage by heat, moisture and dust. Moreover, a lack of electricity or network is possible during natural calamities, due to which a server may not be feasible. Hence, the approach ensures that the system is reliable and in cases of emergencies, can respond to human intervention and instruction to stop a certain process. To summarize, the contributions of this paper are as follows:

1) A distributed, asynchronous and scalable robot-robot interaction using vision-based gestural motion cues
2) Robot-robot interaction using a gesture-based action formation and recognition methodology:
a) Gesture formation in form of shapes using basic motion control

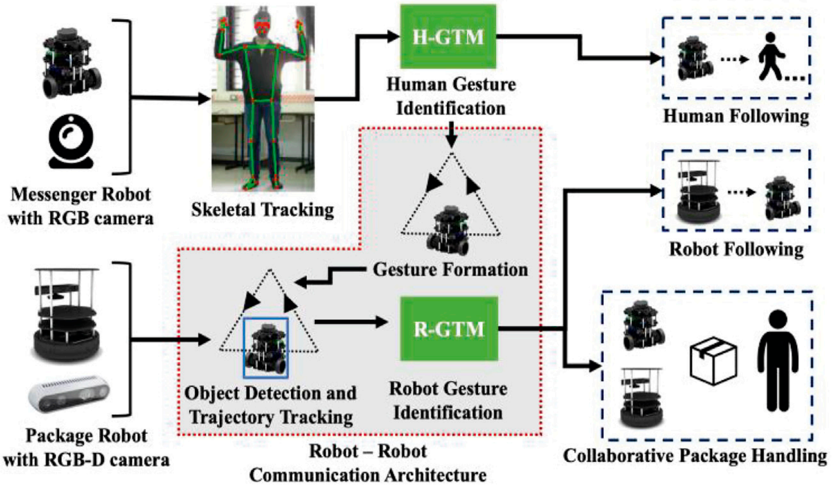

H-GTM
Human Gesture Identification
Human Following
Robot Following
Messenger Robot with RGB camera
Skeletal Tracking
Gesture Formation
R-GTM
Object Detection and Trajectory Tracking
Robot Gesture Identification
Package Robot with RGB-D camera
Robot – Robot Communication Architecture
Collaborative Package Handling

FIGURE 2 | Schematic for a collaborative vision-based gestural interaction framework. The shaded region represents the contribution of this paper.

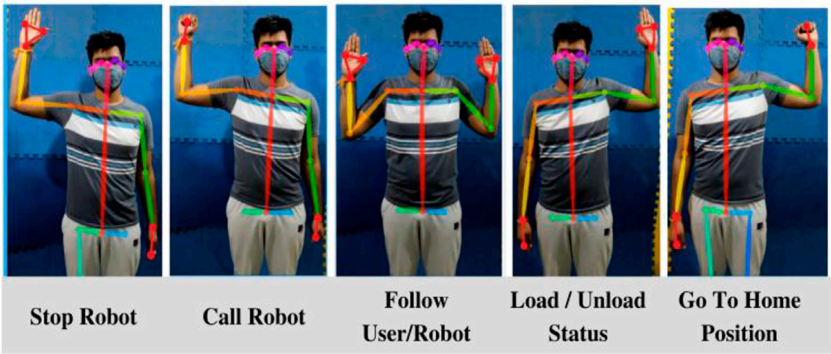


FIGURE 3 | Commands for robot correlating human gestures.

TABLE 1 | Classification of different types of shapes and their corresponding gestures.

Shape	Command	Command attributes
Triangle	Go to location	Orientation of triangle denotes the direction; Duration of gesture correlates to distance to be travelled
Circle	Follow messenger	N/A
Square	Follow a human	N/A

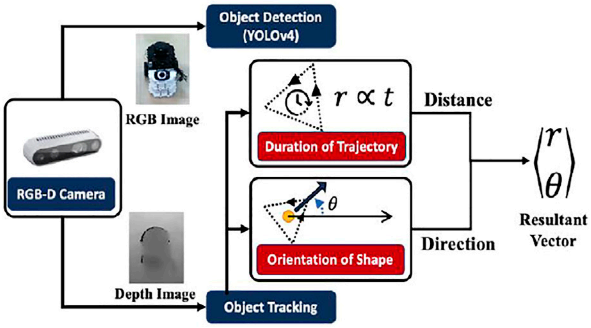


FIGURE 4 | Schematic for the methodology of Gesture Identification and Inference.

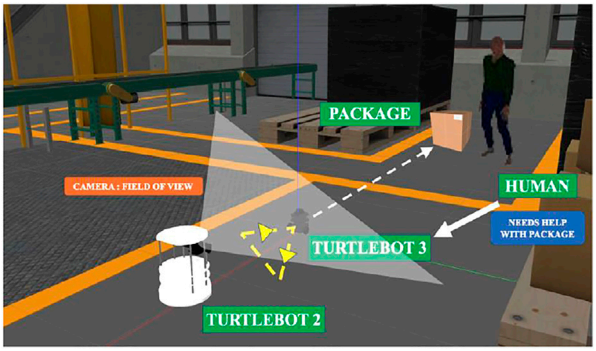


FIGURE 5 | Simulation Environment.

b) Gesture recognition of formed shapes using visual perception
c) Target Position Identification by formulating the orientation and duration of gesture
3) Experimental validation in simulation environment and a physical system of two robots with multiple humans on an experimental testbed of a manufacturing industry setup.

2 PROBLEM FORMULATION

The main focus of this paper is to demonstrate a robot-robot interaction framework using vision-based cues. Additionally, this is also done in collaboration with a human to carry out a package delivery task. The main methodology is to identify the robot and track its trajectory through which a gesture is interpreted, and the command is carried out. **Figure 2** provides an overview of the entire schematic of the proposed architecture.

2.1 Human Gesture Identification

This paper uses skeletal tracking for human body pose estimation to convey a command to the robot. Google's MediaPipe (Lugaresi et al., 2019) is used for tracking the skeletal points. A total of 18 skeletal points of the upper body are taken. Based on the coordinates of these skeletal points, certain body poses are pre-coded into the robot as certain gestures with corresponding commands, as shown in **Figure 3**. Improving skeletal tracking and identification is not the focus of this paper. Body pose estimation is merely used to introduce a human-in-the-loop for a cooperative package handling task.

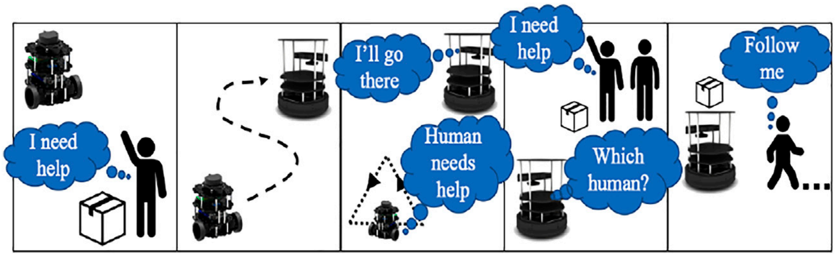


FIGURE 6 | Operational task flow.

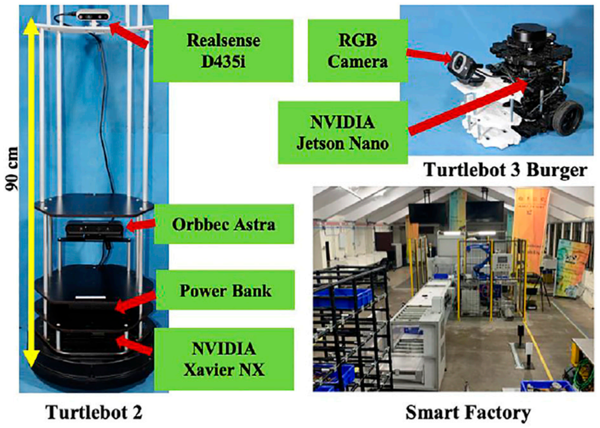


FIGURE 7 | Experimental test set-up.

2.2 Robot-Robot Interaction

The system consists of the following types of robots: (1) package handling robots capable of carrying packages and detecting both human and robot gestures; (2) messenger robots for conveying information to package handling robots and can detect only human gestures. The messenger robots read a human gesture using skeletal tracking for knowing the package to be transported and signal a particular package handling robot to carry out the task. The package handling robot is capable of object detection and tracking by which it can infer the trajectory of a messenger robot depicting a particular action. The methodology concerned with gesture formation and identification are discussed exhaustively in the next section (see **Section 3**).

2.3 Path Planning and Control

The robots are given a map of the environment before-hand constructed using RTAB-MAP (Labbé and Michaud, 2019). Once they receive command with destination location, a global path planner is implemented using the A* algorithm, (Hart et al., 1968), and Dynamic Window Approach (Fox et al., 1997) is used for local path planning. A proportional controller is used on the robot to gradually approach the target location or follow a person or robot. The distance for following is calculated based on the size of the bounding box of objects in the camera frame. These methods are merely for demonstration, and better ones exist in the literature, but that is not the focus of this paper.

3 ROBOT-ROBOT INTERACTION

3.1 Robot Gesture Formation

Following a human command, the robot has to convey instructions to another robot. The messenger robot moves in basic shapes according to the type of command (see **Table 1**). As an extension to this gesture-based command transfer, a bio-inspired gesture attribute recognition, as shown in **Figure 1**, is demonstrated for the case of a triangle, as explained in **Table 1**.

3.1.1 Shape Formation

Before forming a gesture, the messenger robot approaches and faces the other robot head-on using pose estimation from a custom trained YOLOv4 (Bochkovskiy et al., 2020) model. Subsequently, shapes are formed by the robot discussed in **Table 1** using basic Robot Operating System (ROS) commands (Quigley et al., 2009). In the case of a triangle, the robot has to initially orient itself to angle α, which will be the initial angle the robot has to turn to before starting its motion to trace a triangle in direction θ, which is the direction the robot has to convey to the other robot in the framework. To prevent large odometry errors, an angular velocity of $\omega_i = 15^\circ$/sec is chosen for the robot. An equilateral triangle is formed with sides a = 0.4 m. Larger sides have not been chosen since it leads to higher errors in odometry. Similarly, smaller a has not been chosen since the shapes formed are not distinguishable due to a smaller area. The triangles are formed at an orientation of ϕ with a constant linear speed of v and angular speed of $\omega_\delta = 30^\circ$/sec. Here, ϕ is decided by the direction of the package handling robot from its current pose.

3.1.2 Duration Control of Gesture

The goal here is to find a v such that the triangle is traversed in time T at an orientation of ϕ. Accordingly, to cover the trajectory after orienting itself, the relation of duration T to linear speed v of the robot is:

$$T = 3*\frac{\boldsymbol{a}}{\boldsymbol{v}} + 3*\frac{\left(\frac{2\pi}{3}\right)}{\omega_\delta}\ \mathbf{secs} \qquad (1)$$

$$\therefore \boldsymbol{v} = \frac{1.2}{T-12}\,\mathbf{m/sec}$$

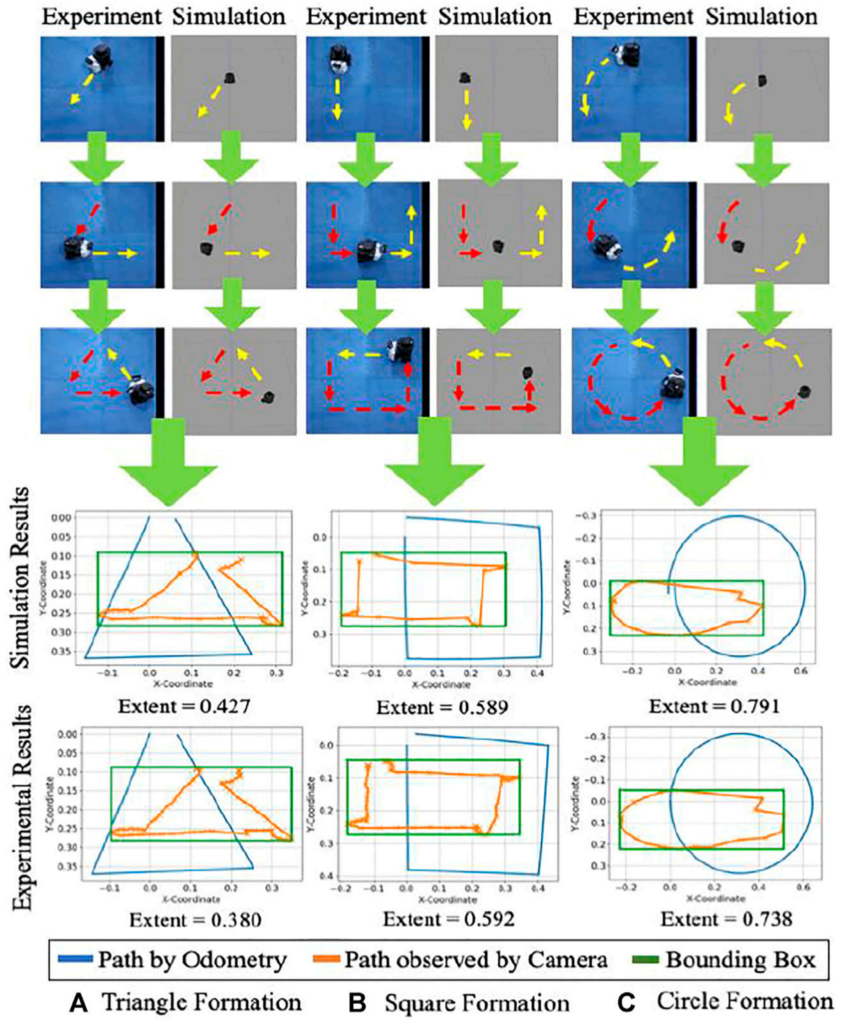


FIGURE 8 | Frame-by-frame shape formation in experiment and simulation. Red arrows indicate the path which has been traced, and yellow arrows indicate the prospective path to be followed. A comparison of odometry data with the path captured by the camera is shown for the physical system and simulation. The bounding box is represented for the observed shape, and the extent of the area is calculated.

3.2 Robot Gesture Identification

After gesture formation, the package handling robot recognizes the gestures based on shape, orientation, and motion duration. The next step after gesture formation is for the package handling robot to recognize the gesture based on the shape, orientation and time in which gesture is completed. An overall schematic of the same is depicted in **Figure 4**.

3.2.1 Object Detection

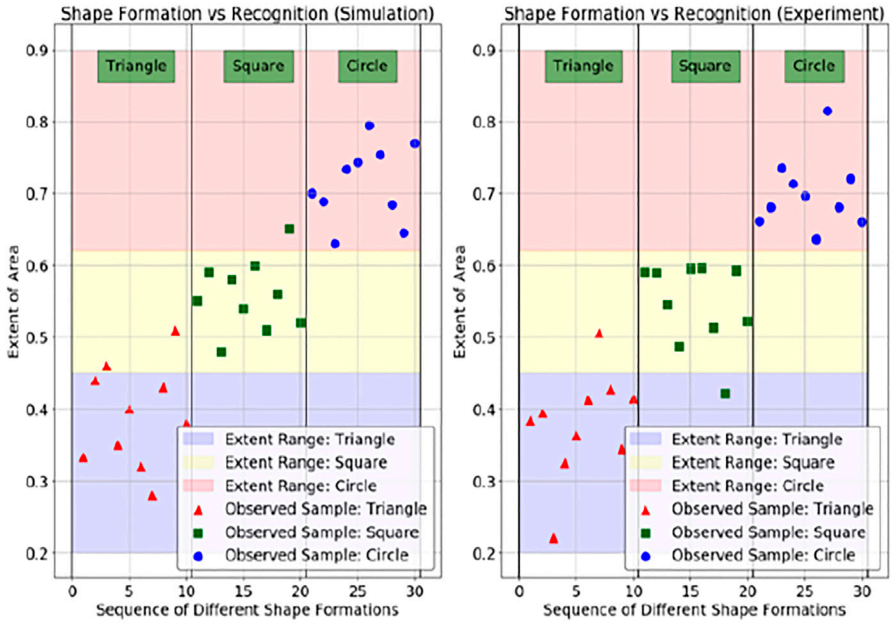
Object detection is implemented on the RGB image from the depth camera for identifying the robot. The YOLOv4 (Bochkovskiy et al., 2020) model is trained on a 500-image custom dataset. The primary aim is to identify the robot and implement a better object detection model beyond this paper's scope.

3.2.2 Object Trajectory Tracking

The depth image from the RGB-D camera is used for finding the corresponding depth of points depicted in the bounding box given by the object detection algorithm. An average of all the points is given as the distance of the robot from the camera, and the coordinates of the robots are tracked. The trajectory of the robot is tracked for its shape and time taken to complete. An important point to note in this section is that the messenger robot moves in a 2D plane but is tracked in a 3D space. This creates a discrepancy in the actual path followed by the messenger robot against the path observed by the package handling robot. However, the shape is similar when viewed from a top view.

3.2.3 Gesture Recognition

The package handling robot is trained to identify the shape of trajectory in relation to the area covered using the methodology for calculating the extent of the area implemented (Das et al., 2016). Let the shape be x,y ∊ [X,Y]. The area of the polygon has to be calculated to find the area (A).



Actual Shape		Predicted Shape		
		Triangle	Square	Circle
	Triangle	8	2	0
	Square	0	9	1
	Circle	0	0	10

Confusion Matrix for Simulation

Actual Shape		Predicted Shape		
		Triangle	Square	Circle
	Triangle	9	1	0
	Square	1	9	0
	Circle	0	0	10

Confusion Matrix for Experiment

FIGURE 9 | Sample data from simulation and experiment for the extent of area covered and ranges for the corresponding shape. The pink, yellow and purple areas are the empirical regions for the classification of shapes. Below the figure are the confusion matrices for the related scenarios.

$$\mathbf{A} = \left| \frac{\left(\sum_{i=1}^{N} x_i * y_{i+1} - x_{i+1} * y_i \right) + \left(x_N * y_1 - x_1 * y_N \right)}{2} \right| \tag{2}$$

To classify a shape, the ratio of area to its bounding box is calculated as extent of area (Das et al., 2016). The bounding box was marked using the minimum and maximum x and y coordinates tracked by the camera. Different shapes tend to have different ranges of extent of area classified empirically after conducting various test cases.

3.2.4 Target Position Identification

A gesture formation and recognition methodology is also implemented for robots to convey a specific point to visit on the map. The duration of the gesture and its orientation to another robot corresponds to the resultant vector from robot to target pose. The case of a triangle is used to depict the task of going to a particular location $\{x_T, y_T\}$ from the source location of robot $\{x_0, y_0\}$. For calculating distance, the time is taken, **T**, to cover the path corresponds to the distance, ***r***, the robot has to travel. Here, ***r*** is to be estimated by the robot, and subsequently, x_T and y_T once the direction is known

$$\boldsymbol{r} = \sqrt{(\boldsymbol{x_T} - \boldsymbol{x_0})^2 - (\boldsymbol{y_T} - \boldsymbol{y_0})^2} \tag{3}$$

This distance inferred by the robot correlates to the total time taken (T) by the messenger robot to complete the gesture. As discussed earlier, a shorter duration corresponds to a shorter distance.

$$\boldsymbol{r} \propto \boldsymbol{T} \,\&.\, \boldsymbol{r} \propto \frac{1}{\boldsymbol{v}} \tag{4}$$

Assuming that the maximum speed of the robot is v_{max} and maximum traversal distance is d_{max}

$$\boldsymbol{r}\big|_{d_{\max}} = \boldsymbol{k} * \left(\frac{3*\mathbf{a}}{\boldsymbol{v}} + \mathbf{3} * \frac{\pi}{3*\omega} \right) \Bigg|_{v_{\max}} \tag{5}$$

Where, where $a = 0.4m$ is the length of side of triangle, v is the linear speed of the robot, $\omega = \pi/12$ is the angular velocity of the

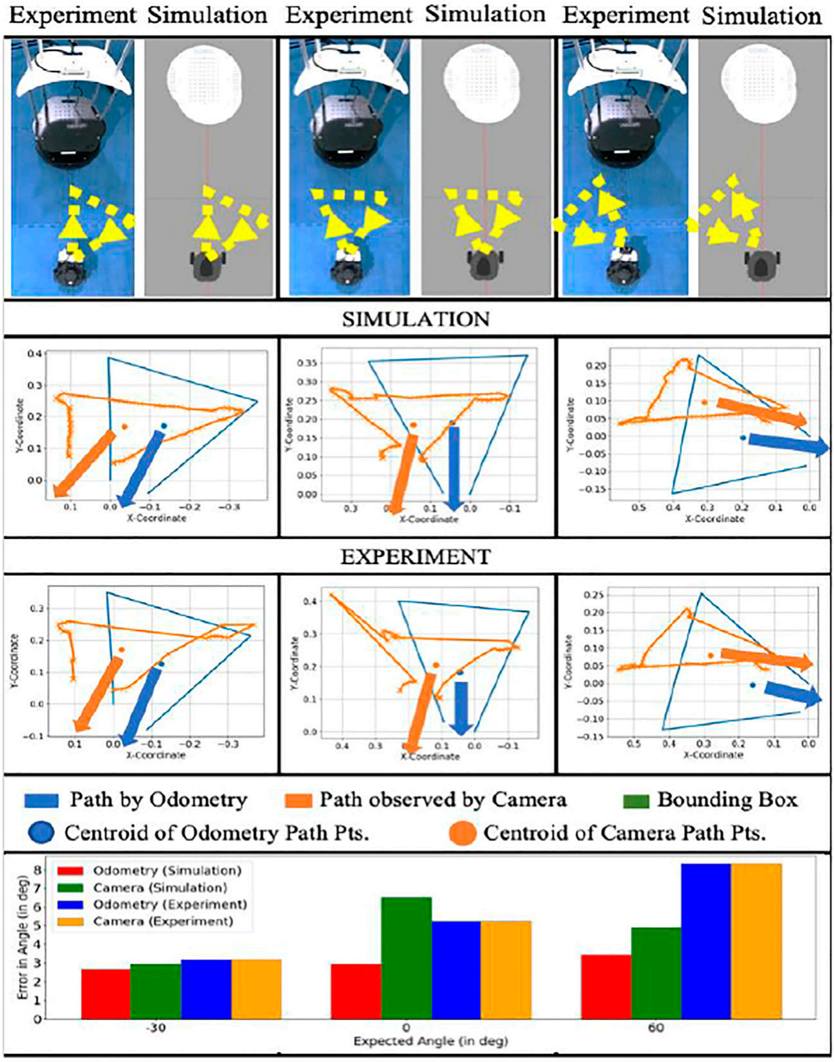


FIGURE 10 | Comparison for few cases of angle orientations and error comparison for odometry and camera perception for simulation and experimental results. The blue and orange arrows indicate the formulated angle. The bar chart depicts the error in the formulated orientation.

robot and k is the proportionality constant with which the entire time period required is multiplied. Formulating **Eq. 5**, we get

$$k = \frac{\boldsymbol{d}_{\max} * \boldsymbol{v}_{\max}}{12 * (\mathbf{0.1} + \boldsymbol{v}_{\max})} \tag{6}$$

Hence, the relation of distance with time duration of gesture is

$$\boldsymbol{r} = \frac{\boldsymbol{d}_{\max} * \boldsymbol{v}_{\max}}{12 * (\mathbf{0.1} + \boldsymbol{v}_{\max})} * \boldsymbol{T} \tag{7}$$

Calculating Direction: The perpendicular to the vertex of the triangle where the gesture starts/ends, from the opposite base denotes the target direction, and the centroid of the triangle lies on its perpendicular. Let $\{x_{obs}, y_{obs}\} \in C$, where C is the set of observed coordinates of robot detected by the camera. The centroid $\{x_{ctd}, y_{ctd}\}$ is

$$\{\boldsymbol{x}_{ctd}, \boldsymbol{y}_{ctd}\} = \left\{ \frac{\sum_0^{N-1} \boldsymbol{x}_{obsi}}{\boldsymbol{N}}, \frac{\sum_0^{N-1} \boldsymbol{y}_{obsi}}{\boldsymbol{N}} \right\} \tag{8}$$

The coordinates of the vertex are taken by taking an average of the first 5, and the last 5 points in C denoted as $\{x_{vtx}, y_{vtx}\}$. Therefore, the resultant angle of direction is formulated as

$$\boldsymbol{\theta} = \boldsymbol{atan}2\left(\boldsymbol{y}_{vtx} - \boldsymbol{y}_{ctd}, \boldsymbol{x}_{vtx} - \boldsymbol{x}_{ctd}\right) \tag{9}$$

Thus, we get a resultant polar vector of $\{\vec{r}, \theta\}$ towards the target location, which is translated to cartesian coordinates on the map as the target pose for the robot.

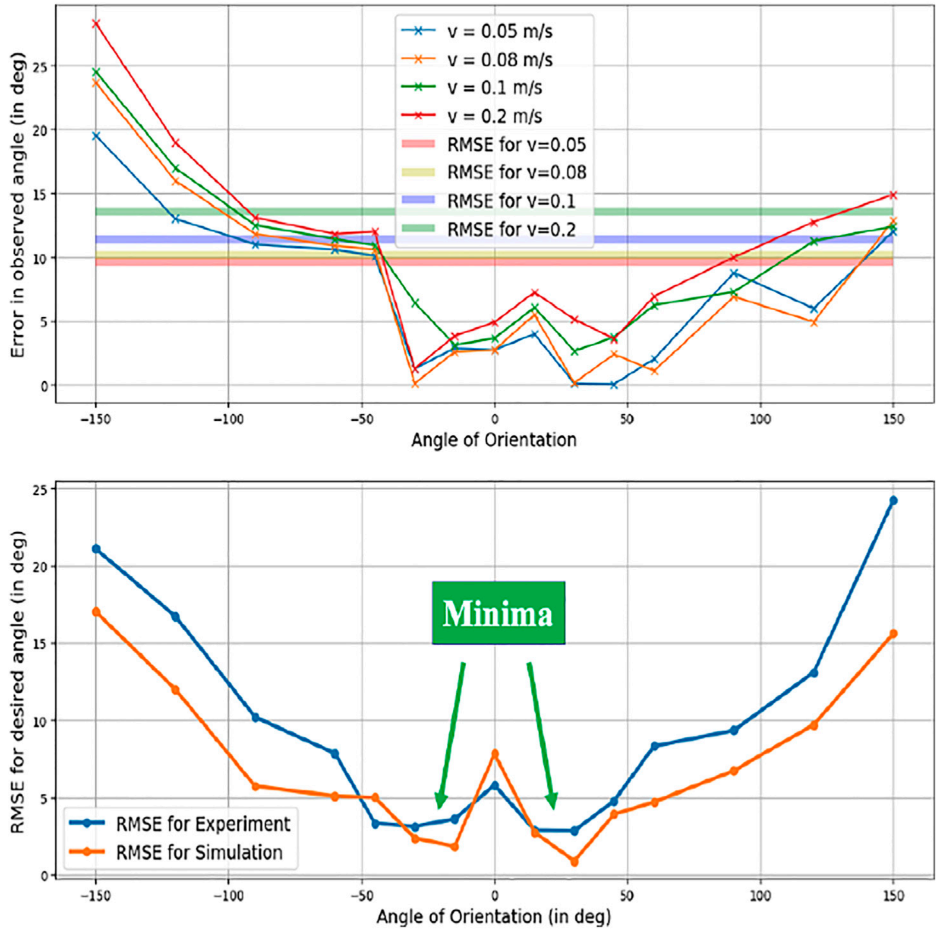


FIGURE 11 | Error plot of directional angles.

4 EXPERIMENT SET-UP

The simulation environment depicted in **Figure 5** is created in the Gazebo (Koenig and Howard, 2004) simulator. The robots used are a Turtlebot 2 equipped with a Kinect camera at the height of 90 cm and a Turtlebot 3.

The cases for forming different shapes and triangles at different velocities and orientations are carried out in Gazebo. These gestures are tracked with the simulated Kinect camera. The setup is tested on a physical system, and the experimental task flow is depicted in **Figure 6**.

The setup was tested in an industrial testbed in an area of 15 m × 6 m, consisting of a conveyer belt, a pick-to-light system, and a fixed manipulator, as shown in **Figure 7**. We test our framework using two modified robot platforms. All the robots were tested on Ubuntu 18.04 running ROS Melodic.

4.1 Package Handling Robot

A modified Turtlebot 2 has additional 3D printed plates at heights of 90 and 120 cm to accommodate space for extra sensors and packages. An NVIDIA Xavier NX was mounted on the base of the robot for a physically smaller processing system. Orbbec Astra RGB-D camera is attached at the height of 35 cm for conducting SLAM and Intel Realsense D435i RGB-D camera at the height of 90 cm with a 15° downward tilt for object detection and skeletal tracking. The Realsense cameras are accurate for object tracking and have been used in past literature for tracking position of objects due to their precise depth tracking. It possesses a depth sensor with a resolution upto 1,280 × 720 pixels at 30 frames per second. A brief description of accuracy of object tracking and precision of RealSense cameras has been described and reasoned in (Carfagni et al., 2019).

4.2 Messenger Robot

A modified version of a Turtlebot 3 is used for the experiment. Additional 3D printed PLA plates were provided to the robot to accommodate an RGB camera. NVIDIA Jetson Nano was used as an onboard processing system for better image processing and faster computation.

The framework is tested for the following tasks:

i) Messenger robots receive and convey distance and direction of location from human to package handling robots
ii) Messenger robots specify the target package
iii) Instructs package delivery robot to follow messenger robot

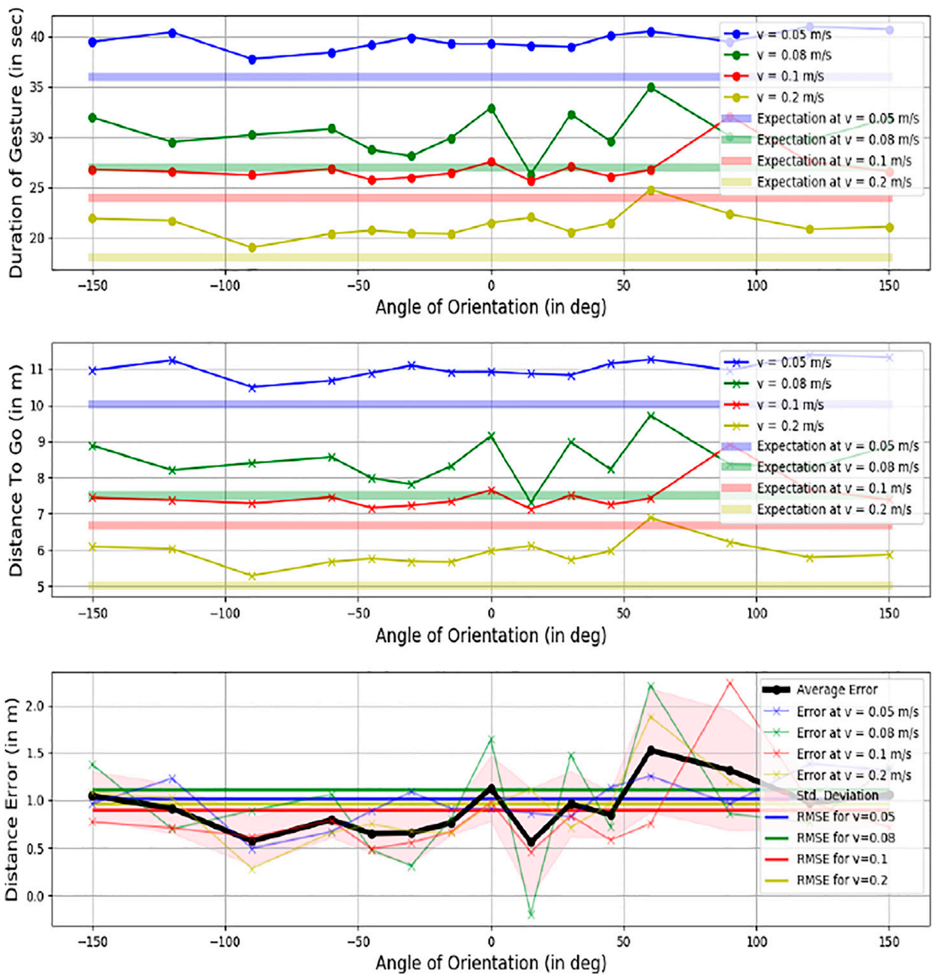


FIGURE 12 | Error plot of distance accuracy.

TABLE 2 | Summary of RMSE.

	e_θ (in Degrees)		e_r (in m)		e_d (in m)	
	Sim.	**Exp.**	**Sim.**	**Exp.**	**Sim.**	**Exp.**
0.05	5.295	7.984	~	1.024	0.391	0.528
0.08	6.378	8.058	0	1.119	0.371	0.468
0.1	7.621	9.572	~	0.908	0.343	0.432
0.2	8.542	11.915	0	0.972	0.292	0.415
Average	6.959	9.382	~0	1.006	0.349	0.461

Additionally, 3 human operators were working in the given environment in cooperation with the robots. The people were instructed with the gestures and their corresponding commands. Preliminary setup was tested for a given set of velocities and intended directions. For the current setup, the Turtlebot3 (messenger robot) is operated at a maximum velocity v_{min} = 0.05 m/s and the maximum distance in the environment to be d_{max} = 10 m. Subsequently, according to **Eq. 8**, the proportionality constant k = 0.278. We took certain angles from the range of -150° to 150° at regular intervals at either 15° or 30°. The experiments were initially carried out for the cases of v = 0.05 m/s, 0.08 m/s, 0.1 m/s and 0.2 m/s evaluating to a total number of 64 cases.

5 RESULTS

This section analyses the robot-robot interaction framework. The two primary factors to be analyzed were a) How does one robot differentiate between different gestures (shapes) drawn by the other robot? b) What is the accuracy of direction and orientation perceived by the package handling robot? The performance for human gesture identification, object detection, and path planning has not been evaluated since standard libraries and algorithms have been used. The methodologies above have been used only for demonstrating the framework as a whole.

5.1 Gesture Formation and Identification

Figure 8 describes gesture formation by the messenger robot and identification by the package handling robot. Trials of the

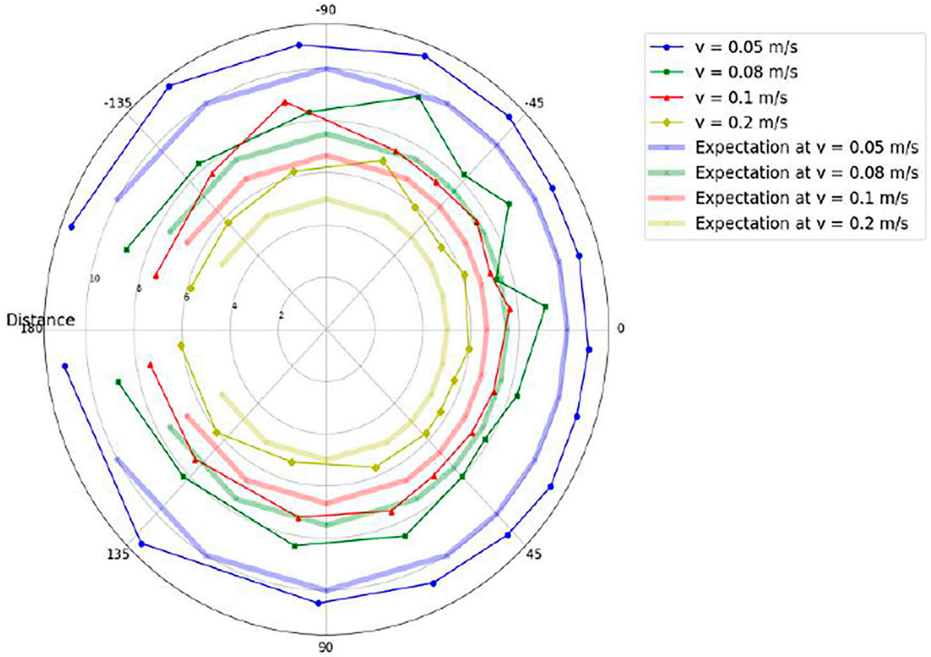


FIGURE 13 | Polar representation of Intended Destination Positions *vs*. Expected Positions.

messenger robot were conducted for tracing a triangle, square, and circle, according to **Eq. 3**, the area was calculated for each instance and compared with the given ratio to test for similarity to the shape.

An illustration of the same is shown in **Figure 8**. The values of extent area obtained from these test cases were empirically formulated (Das et al., 2016) to create regions of gesture identification. Triangles were observed to have an extent area ratio of less than 0.45, squares from 0.45 to 0.63, and circles occupying 0.63 to 0.9 part of the bounding box. The same has been depicted for 10 trials, each for a different shape in **Figure 9** with a confusion matrix for the number of outliers and inliers. The confusion matrix for the experiments of triangle showed a better result compared to the simulations. This is attributed to a reasoning of a probable better estimate from the camera and tracking system. Moreover, the simulations were carried out in Gazebo which has close similarity to robots simulated in real world environments. Hence, a similar or better confusion matrix can be expected in either the case of simulation or experimental setup.

5.2 Gesture Formation and Detection for Different Directions

Figure 10 contains three out of the 64 cases to depict how gestures are identified for different orientations shown by the messenger robot and perceived by the package handling robot. The overall analysis for all 64 cases has been discussed in the results from **Figure 11**; **Figure 12**. The vector between the centroid and the midpoint of the average of 5 initial and final points gives the robot's orientation. The significant error in orientation is due to odometry error which adds up to the error of direction perceived by the camera.

Additionally, trajectory tracking by the camera has multiple fluctuations due to errors in depth estimation by the RGB-D camera, contributing to an added error on top of odometry error.

5.2 Performance Evaluation Metrics

5.2.1 Directional Error

Figure 11 describes the error in formulating the intended orientation of the triangle, and subsequently the direction to go in for the 64 cases. The corresponding results are depicted in **Figure 12**. A minimum error was observed at the angles of -30° and 30°.

$$e_{\theta} = \frac{1}{N}\sqrt{\sum_{i=1}^{N}\left(\theta_{obs} - \theta_{\exp}\right)^{2}} \qquad (10)$$

Where θ_o is the observed direction and θ_e, is the expected direction.

5.2.2 Distance Error

Figure 12 describes the distance accuracy error. This metric analyses the formulation's efficiency to infer the proper distance to travel for the linear velocity of the robot, i.e., how the error changes as the intended distance are increased or decreased. The duration for each case was calculated based on when the robot starts and stops moving. The corresponding predicted distances (r) were formulated for all angles from **Eq. (7)**. The error in the duration of motion is negligible for simulations since the environment is ideal. The experimental results gave an average error of 1.006 m. It was observed that the error increases as the velocity of robot increases, which can be attributed to the fact that chances of odometry errors are higher at higher speeds

$$\boldsymbol{e_r} = \frac{1}{\boldsymbol{N}}\sqrt{\sum_{i=1}^{N}\left(\boldsymbol{r_{obs}} - \boldsymbol{r_{\exp}}\right)^2} \tag{11}$$

5.2.3 Resultant Position Error

Here, the overall accuracy is tested by evaluating the final coordinates. Once both distance (r) and direction (θ) values are generated, the resultant vector $\overrightarrow{\{r,\theta\}}$ is formulated. As seen in **Table 2**, the observed error trend depicts that the error is lower at higher speeds. This shows that the absolute position error is more dependent on the distance error than the orientation error. The resultant coordinates (r, θ) for each case of given orientation and speed are depicted in **Figure 9** and the average error is represented in **Figure 10**. The overall resultant position and orientation of the robot has been represented in **Figure 13** as a polar plot. It has been observed that the distance as well as orientation shows a higher error when the speed is higher, or the required orientation is larger. This can also be shown via observations recorded in **Table 2**. The orientations at −30°and 30 °are the orientations where the messenger robot does not have to take an initial deviation before forming the gesture. Hence, these orientations show the least deviations and errors. **Table 2** also draws an interesting perspective that the resultant error is influenced more by the errors observed in the orientation rather than the ones observed in position where ***r*** is a major contributor to the $\boldsymbol{e_d}$ since error in θ is reduced to a range of [-1,1].

6 CONCLUSION AND FUTURE SCOPE

This paper proposes a novel bio-inspired framework for robots to interact with only visual cues and motion. A cooperative object manipulation task is performed to demonstrate the same by introducing a human-in-the-loop using traditional human-robot interaction methodologies. The key ideas presented in the paper are that a robot can interact with another robot by moving in a particular manner with specific kinematic constraints. Three gestures were demonstrated to signify different commands. Gesture Identification gave an accuracy of 90% for simulations and 93.33% for experiments. A framework for going to a particular location based on a triangle traced by the robot in a specific orientation and time duration was also implemented. The entire framework was tested for its accuracy in conveying the gesture and the intended coordinates in a simulation and an experimental setup. The average positional error was 0.349 m in simulations and 0.461 m for experiments. This framework utilized a team of different robots to demonstrate that the system is scalable. As a future problem, the system can also be trained to recognize the intensity of human gestures. Additionally, the robots can be trained to estimate the trajectory better by modeling a robot after detecting the robot. The applications of this framework in the real world are limitless. As shown in the experimental scenario of the paper, the framework can be used in an industrial environment, but it can also be extended to a search and rescue operation in tight environments where communication may fail. Moreover, it can collaborate with humans and carry out a rescue operation. The main advantage is that it uses only a single modality of vision which allows it to be compatible with robots of various sizes, thereby enabling it to function in a heterogeneous team of robots. The framework can be used to accommodate such multiple robots to create a completely independent and scalable team of robots.

DATA AVAILABILITY STATEMENT

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

ETHICS STATEMENT

Written informed consent was obtained from the individual(s) for the publication of any potentially identifiable images or data included in this article.

AUTHOR CONTRIBUTIONS

The paper conceptualization and ideation were done by second author AR, Project PI. Simulations and experiments were done by KJ. Paper was jointly discussed and written by both authors.

FUNDING

This project is funded by Samarth Udyog, Common Engineering Facility Centre (CEFC) CPDM, IISc, Bangalore; Department of Heavy Industries (DHI), Govt. of India Grant No. SG/MHRD-0016.

ACKNOWLEDGMENTS

We acknowledge the contributions of Mr. Pratyaksh Rao, New York University (NYU) for his valuable comments and IISc, Bangalore staff for their help during the Project Implementation.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/frobt.2022.915884/full#supplementary-material

REFERENCES

Bochkovskiy, A., Wang, C. Y., and Liao, H. Y. M. (2020). *Yolov4: Optimal Speed and Accuracy of Object Detection*. arXiv preprint arXiv:2004.10934.

Carfagni, M., Furferi, R., Governi, L., Santarelli, C., Servi, M., Uccheddu, F., et al. (2019). Metrological and Critical Characterization of the Intel D415 Stereo Depth Camera. *Sensors* 19, 489. doi:10.3390/s19030489

Chu, Q., Ouyang, W., Li, H., Wang, X., Liu, B., and Yu, N. (2017). "Online Multi-Object Tracking Using CNN-Based Single Object Tracker with Spatial-

Temporal Attention Mechanism," in *Proceedings of the IEEE International Conference on Computer Vision*, 4836–4845. doi:10.1109/iccv.2017.518

Crivellaro, A., Rad, M., Verdie, Y., Yi, K. M., Fua, P., and Lepetit, V. (2017). Robust 3D Object Tracking from Monocular Images Using Stable Parts. *IEEE Trans. Pattern Anal. Mach. Intell.* 40 (6), 1465–1479. doi:10.1109/TPAMI.2017.2708711

Das, B., Couceiro, M. S., and Vargas, P. A. (2016). MRoCS: A New Multi-Robot Communication System Based on Passive Action Recognition. *Robotics Aut. Syst.* 82, 46–60. ISSN 0921-8890. doi:10.1016/j.robot.2016.04.002

Esch, H., and Burns, J. (1996). Distance Estimation by Foraging Honeybees. *J. Exp. Biol.* 199 (1), 155–162. doi:10.1242/jeb.199.1.155

Fox, D., Burgard, W., and Thrun, S. (1997). The Dynamic Window Approach to Collision Avoidance. *IEEE Robot. Autom. Mag.* 4 (1), 23–33. doi:10.1109/100.580977

Hart, P., Nilsson, N., and Raphael, B. (1968). A Formal Basis for the Heuristic Determination of Minimum Cost Paths. *IEEE Trans. Syst. Sci. Cyber.* 4 (2), 100–107. doi:10.1109/tssc.1968.300136

Issac, J., Wuthrich, M., Cifuentes, C. G., Bohg, J., Trimpe, S., and Schaal, S. (2016). "Depth-based Object Tracking Using a Robust Gaussian Filter," in Proceeding of the 2016 IEEE International Conference on Robotics and Automation (ICRA), 608–615. doi:10.1109/icra.2016.7487184

Koenig, N., and Howard, A. (2004). "Design and Use Paradigms for Gazebo, an Open-Source Multi-Robot Simulator," in Proceeding of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2149–2154.3

Kuniyoshi, Y., Rickki, J., Ishii, M., Rougeaux, S., Kita, N., Sakane, S., et al. (1994). "Vision-based Behaviors for Multi-Robot Cooperation," in Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'94), 925–932.

Labbé, M., and Michaud, F. (2019). RTAB-map as an Open-Source Lidar and Visual Simultaneous Localization and Mapping Library for Large-Scale and Long-Term Online Operation. *J. Field Robot.* 36, 416–446. doi:10.1002/rob.21831

Landgraf, T., Bierbach, D., Kirbach, A., Cusing, R., Oertel, M., Lehmann, K., et al. (2018). *Dancing Honey Bee Robot Elicits Dance-Following and Recruits Foragers*. arXiv preprint arXiv:1803.07126.

Lin, K. H., and Wang, C. C. (2010). "Stereo-based Simultaneous Localization, Mapping and Moving Object Tracking," in Proceeding of the 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IEEE), 3975–3980. doi:10.1109/iros.2010.5649653

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., et al. (2016). "Ssd: Single Shot Multibox Detector," in *European Conference on Computer Vision* (Springer), 21–37. doi:10.1007/978-3-319-46448-0_2

Lugaresi, C., Tang, J., Nash, H., McClanahan, C., Uboweja, E., Hays, M., et al. (2019). *MediaPipe: A Framework for Building Perception Pipelines*. ArXiv, abs/1906.08172.

Lukezic, A., Kart, U., Kapyla, J., Durmush, A., Kamarainen, J. K., Matas, J., et al. (2019). "Cdtb: A Color and Depth Visual Object Tracking Dataset and Benchmark," in Proceedings of the IEEE/CVF International Conference on Computer Vision, 10013–10022. doi:10.1109/iccv.2019.01011

Melanie, S., Martina, U., Micha, S., and Wil-fried, E. (2020). Swarm Robotic Behaviors and Current Applications. *Front. Robotics AI* 7, 36. doi:10.3389/frobt.2020.00036

Meng, Y., Nickerson, J. V., and Gan, J. (2006). "Multi-robot Aggregation Strategies with Limited Communication," in Proceeding of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2691–2696. IROS 2006. doi:10.1109/iros.2006.281991

Nascimento, J. C., Abrantes, A. J., and Marques, J. S. (1999). "An Algorithm for Centroid-Based Tracking of Moving Objects," in Proceedings of the 1999 IEEE International Conference on Acoustics, Speech, and Signal Processing, 3305–3308. Proceedings. ICASSP99 (Cat. No. 99CH36258). doi:10.1109/icassp.1999.7575486

Parker, L. E., Kannan, B., Tang, F., and Bailey, M. (2004). "Tightly-coupled Navigation Assistance in Heterogeneous Multi-Robot Teams," in Proceeding of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 1016–1022. (IEEE Cat. No.04CH37566).

Quigley, M., Gerkey, B., Conley, K., Faust, J., Foote, T., Leibs, J., et al. (2009). "ROS: An Open-Source Robot Operating System," in Proc. Open-Source Software Workshop Int. Conf. Robotics and Automation.

Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2016). "You Only Look once: Unified, Real-Time Object Detection," in Proceedings of the IEEE conference on computer vision and pattern recognition, 779–788. doi:10.1109/cvpr.2016.91

Redmon, J., and Farhadi, A. (2018). *Yolov3: An Incremental Improve-Ment*. arXiv preprint arXiv:1804.02767.

Riley, J. R., Greggers, U., Smith, A. D., Reynolds, D. R., and Menzel, R. (2005). The Flight Paths of Honeybees Recruited by the Waggle Dance. *Nature* 435, 205–207. doi:10.1038/nature03526

Shafiee, M. J., Chywl, B., Li, F., and Wong, A. (2017). *Fast YOLO: A Fast You Only Look once System for Real-Time Embedded Object Detection in Video*. arXiv preprint arXiv:1709.05943.

Tjaden, H., Schwanecke, U., Schomer, E., and Cremers, D. (2018). A Region-Based Gauss-Newton Approach to Real-Time Monocular Multiple Object Tracking. *IEEE Trans. Pattern Anal. Mach. Intell.* 41 (8), 1797–1812. doi:10.1109/TPAMI.2018.2884990

Wang, Z., and Schwager, M. (2015). "Multi-robot Manipulation with No Communication Using Only Local Measurements," in Proceeding of the 2015 54th IEEE Conference on Decision and Control (CDC), 380–385.

Yan, Z., Jouandeau, N., and Cherif, A. A. (2013). A Survey and Analysis of Multi-Robot Coordination. *Int. J. Adv. Robotic Syst.* 10 (12). doi:10.5772/57313

Zhai, S., Shang, D., Wang, S., and Dong, S. (2020). DF-SSD: An Improved SSD Object Detection Algorithm Based on DenseNet and Feature Fusion. *IEEE Access* 8, 24344–24357. doi:10.1109/access.2020.2971026

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.