2015 IEEE International Conference on Robotics and Automation (ICRA)
Washington State Convention Center
Seattle, Washington, May 26-30, 2015

Gesture Based Human - Multi-Robot Swarm Interaction and its Application to an Interactive Display

J. Alonso-Mora[1,2], S. Haegeli Lohaus[1], P. Leemann[1], R. Siegwart[1] and P. Beardsley[2]

Abstract— **A taxonomy for gesture-based interaction between a human and a group (swarm) of robots is described. Methods are classified into two categories. First, *free-form* interaction, where the robots are unconstrained in position and motion and the user can use deictic gestures to select subsets of robots and assign target goals and trajectories. Second, *shape-constrained* interaction, where the robots are in a configuration shape that can be modified by the user. In the later, the user controls a subset of meaningful degrees of freedom defining the overall shape instead of each robot directly.**

A multi-robot interactive display is described where a depth sensor is used to recognize human gesture, determining the commands sent to a group comprising tens of robots. Experimental results with a preliminary user study show the usability of the system.

I. Introduction

Robots have a growing role in society and are helping to perform varied tasks. Successful applications of large multi-robot teams include industrial automation [1] and entertainment [2]. Future applications are likely to include inspection and search-and-rescue [3]. In complex tasks, where a human must provide (in real-time) high-level control of the robot swarm, intuitive or readily-learnt interfaces are necessary. Enabling a human operator to control a robot swarm with hundreds of robots is still an open problem due to its high dimensionality in the number of degrees of freedom.

A. Contribution

We contribute a taxonomy of methods for gesture-based interaction between a human and a swarm (team) of robots.

- *Free-form.* Mostly deictic gestures are used for direct control of the robots' free-form motion, meaning unconstrained in position and motion. The user can select sub-groups of robots or individual robots in the swarm, change their positions and move them along trajectories.
- *Shape-constrained.* Provides constrained control of the robot swarm, where a configuration shape is used and only a subset of degrees of freedom are controlled.
  - *Fixed formation.* Representational gestures are used to define the formation and adjust its parameters.
  - *Constrained shape morphing.* An enclosing shape is considered, for example for optimal coverage of an area, modified by adjusting its size and position.

These methods are tested with a novel interactive display formed by twenty small robots controlled through a depth (RGB-D) sensor used to recognize human gesture.

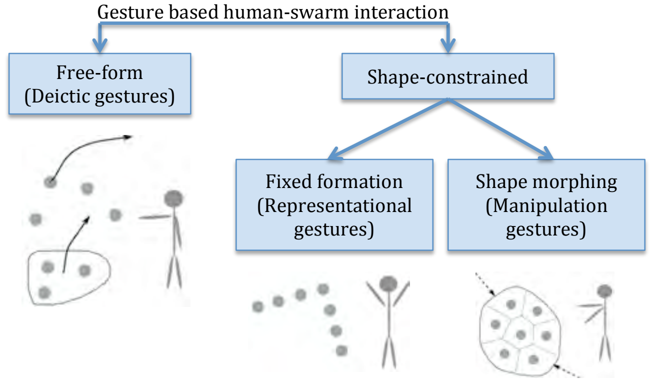

Gesture based human-swarm interaction
Free-form (Deictic gestures)
Shape-constrained
Fixed formation (Representational gestures)
Shape morphing (Manipulation gestures)

Fig. 1. Taxonomy for Human Robot Swarm interaction, with representative examples of gestures, position of the robots (circles) and motion or deformation commands. In free-form, individuals or subgroups can be selected.

B. Applicability

The methods are directly applicable in the area of entertainment robotics, which is the driving force behind this paper. In previous works [2] we developed a novel display formed by tens of small robots acting as independent movable pixels, which allowed us to display animations by combining the motion and changing color of each individual robot.

In this work, the goal is to extend that novel display to an interactive display formed by tens of robots, where the user is capable of controlling their position, movement and color in an intuitive fashion. The user may individually and directly control the robots (*free-form* interaction) or have reduced control over a set of meaningful degrees of freedom (*shape-constrained* interaction).

Beyond entertainment, this work has potential in other applications where a large team of robots needs to be controlled in real-time and intuitively. For instance, taking search-and-rescue as an example - (a) one might want to direct individual robots or define teams for specific tasks, such as surveillance, which is the topic of 'free-form' interaction, or (b) a search formation or strategy might be predefined or automatically generated (for example for optimal coverage) and one would be able to globally adjust its parameters in real-time and intuitively, which is the topic of 'shape-constrained' interaction.

C. Related works

For successful human robot swarm interaction both gesture recognition and control of the swarm of robots are required. Most of the gesture recognition interfaces are organized in

[1] J. Alonso-Mora et al. are with the Autonomous Systems Lab, ETH Zurich, 8092 Zurich, Switzerland {jalonsom}@mit.edu

[2] J. Alonso-Mora and P. Beardsley are with Disney Research Zurich

978-1-4799-6923-4/15/$31.00 ©2015 IEEE

three steps: detection, tracking and recognition. Methods for detection and tracking include particle filtering and color/depth similarity [4]. Common methods for gesture recognition rely on template matching, either static (using randomized decision forests [5]) or temporal [6]. Similar to our method, the relative angles of the user's arms were employed in the latter.

From a control standpoint, [7] discussed the main features of Human Swarm Interaction and its differences to Human Robot Interaction. In that work human inputs were used to render the mission more efficient. Likewise to [8], the human was treated as a supervisor. [9] further described ten different levels of automation depending on the involvement of the human. Recent work on human-swarm interaction includes [10], which presented an investigation (in simulation) of two basic types of human-swarm interaction (a selection and a beacon control method) to enable the operator control of robot swarms. Similar to our gesture based control of a swarm of robots, [11] presented a method to control subswarms of robots, but with the goal of studying resource allocation and guidance to a goal configuration. In contrast, we propose a taxonomy for human-swarm interaction, introduce shape-based control to reduce the number of degrees of freedom and apply the methods in a novel interactive display formed by tens of robots.

D. *Organization*

Section II describes the system used for entertainment. Section III introduces the taxonomy of gestures. Sections IV and V describe free-form and shape-constrained interaction, respectively. Section VI provides experimental results including an initial user study and Section VII concludes this paper.

II. System overview - Interactive display

In this section the specific setup used for the entertainment application is described in detail.

A. *Physical setup*

The basic experimental setup, shown in Figure 2 and described in detail in [2] includes a central PC where all computations take place, an overhead camera for localization of the robots and multiple (up to fifty) small "GCtronic Elisa" differentially driven robots with a diameter of 7 cm and maximum speed of 0.5 m/s. Each robot is equipped with an RGB-LED and listens to motion/color commands broadcasted from the central PC at 10Hz. In this work, the setup is completed with an RGB-D sensor.

A Microsoft Kinect RGB-D sensor[1] is used for body tracking and to identify the gestures of the user. The OpenNI[2] library is employed, extended by the middleware component NITE[3] for body tracking. Computed from the depth and images of the sensor, the output is the estimated position of nine joints of the user, one for the head, three for each arm and two for the body. Due to the limited field of view of the sensor, the legs of the user could not be tracked. An example of the tracked joints is shown in forthcoming Figure 4(a).

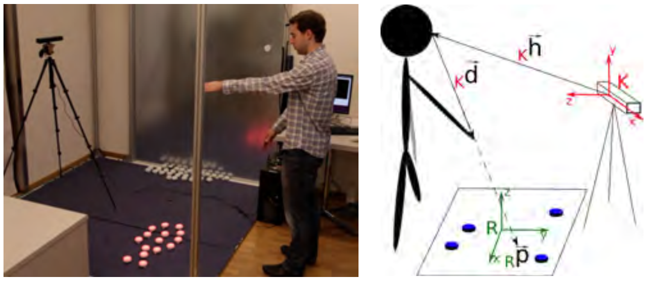


Fig. 2. *Left:* Experimental setup, with a team of robots, an RGB-D sensor, a camera and a central PC. *Right:* Robot (R) and sensor (K) coordinate systems, with pointing vector $_K\mathbf{d}$ and sensor-head vector $_K\mathbf{h}$.

A configuration of the sensor on a tripod in front of the user (as shown in Figure 2) was chosen after evaluating several configurations: placing the sensor on the ground or on a tripod, in front or on the side of the user and measuring the pointing direction via the head-hand or shoulder-hand vector. A user pointed at different locations and the pointing direction was inferred. Ground truth was obtained via an external VICON tracking system and standard deviation results are displayed in Table I. The data shows that tracking the direction head-hand with the camera in front of the user provides the best performance. Although lower standard deviation was observed for the sensor on the ground, that configuration was more likely to failures in the detection due to unseen joints, for instance when pointing directly to the ground at the position of the sensor.

B. *Pointing gesture*

Denote by A_{RK} the affine transformation matrix from the robot coordinate frame R to the sensor coordinate frame K (see Figure 2), which can be obtained by an auto calibration procedure[4]. The coordinates of the point $_R\mathbf{p}$ in the workspace where the user is pointing at are given by the intersection of the ground plane of normal $_R\mathbf{n} = (0,0,1)^T$ with a ray $_R\mathbf{d}$ from the user's head $_R\mathbf{h}$ to his hand. Since the later vectors

TABLE I

Zero-mean error distributions of pointing direction estimate for various configurations.

Position sensor	Pointing direction	Standard deviation [m]
Front, ground	head - hand shoulder - hand	$\sigma = 0.031903$ $\sigma = 0.080597$
Front, tripod	head - hand shoulder - hand	$\sigma = 0.039065$ $\sigma = 0.061654$
Side, ground	head - hand shoulder - hand	$\sigma = 0.56674$ $\sigma = 0.52996$
Side, tripod	head - hand shoulder - hand	$\sigma = 0.67285$ $\sigma = 0.57122$

[1] Microsoft Kinect, http://www.xbox.com/Kinect

[2] Open Natural Interaction, http://openni.org

[3] NITE, Prime Sense, http://www.primesense.com/Nite/

[4] The auto-calibration procedure is as follows: a minimum of three robots display a color sequentially. Their position in the Robot frame is given by the overhead camera. The user points at the robot with his right arm and reaches out his left arm to the left as a "click" indicator. The coordinate frame transformation A_{RK} is obtained numerically by solving the resulting system of equations (two per robot).

are given in the sensor K frame they must be transformed into the robot frame R by multiplying by A_{RK}, leading to,

$$_{R}\mathbf{r} = A_{RKK}\mathbf{h} - \frac{(A_{RKK}\mathbf{h}) \cdot {}_{R}\mathbf{n}}{(A_{RKK}\mathbf{d}) \cdot {}_{R}\mathbf{n}}(A_{RKK}\mathbf{d}).$$

C. Robot selection

To select/deselect a single robot, the user points with the right arm at the robot for one second. The robot closest to the pointed position or within a radius α around it is the target. If it is already selected it gets deselected, otherwise it gets selected. Different color LEDs on the robots are used to indicate at which robot the user is pointing at and which are selected.

To select a group of robots the user must draw a closed shape that encompasses them. A closed shape is defined by two pointed positions that are near in space but not in time and can be convex or non-convex. Standard algorithms, such as the ”crossing-number algorithm” or the ”winding number algorithm” can be used to identify the robots encompassed by the closed shape, which get selected/deselected.

D. Collision-free movement

To simplify the interaction, inter-agent collision avoidance is automatically handled in real-time. The local motion planning algorithm of [12] is employed, that computes, for each robot independently, a collision-free velocity with minimal deviation from the preferred velocity (commanded by the methods here presented) and that respects the kinematic constraints of the robots.

III. Taxonomy of gestures

Although a broader taxonomy of gestures for Human Computer Interaction exists [13], we divide gestures for Human-Swarm Interaction intro three distinct categories:

- *deictic* - pointing gestures.
- *representational* - associated to an object or idea.
- *manipulation* - designed to morph a shape.

For *free-form* interaction mostly deictic gestures are used, which are readily learnt by experimentation or by demonstration. For *shape-constrained* interaction, mostly manipulation and representational gestures appropriate to the shape are used. The color of the robots is used as feedback to the user.

All the gestures made by the user are recognized via the relative position of the joints, so that gestures are valid for users of different body sizes. From an algorithmic point of view we distinguish between three types of gestures.

- *Pose gestures.* The user must stay in the same position for a determined time and the mean relative position of the joints is computed. An example is the *selection* shown in Figure 2, where the user points for one second in order to select a robot.
- *Motion gestures.* A motion pattern is searched in consecutive frames, with a pre-defined maximum length. This is similar to the *temporal template matcher* described in the related work. An example is a waving gesture.
- *Hybrid gestures.* A pose gesture defines the start and the end. All the in-between frames are analyzed and the joint positions are used. One example is the *changing color* gesture shown in Figure 3(d) where the user holds his left hand to the front (starting signal) and the robots adopt different colors when the user moves his right hand up and down, until the left arm is moved down.

In this work, the designers chose the gestures after an iterative process of experiments and discussion. Due to the limited amount of simple and distinguishable commands, in our interactive display four different modes are created in the interface to illustrate the various concepts:

- *goal mode* (free-form) - the user is able to select robots and assign new goal positions.
- *trajectory mode* (free-form) - a trajectory to be followed can be commanded to the robots.
- *skeleton mode* (shape-constrained) - the robots form an skeleton and mimic the movements of the user.
- *smile mode* (shape-constrained) - the robots form a face that can be morphed.

The user is able to choose between modes by pointing to four distinct (and labeled) areas at the far end of the ground floor. Voice commands could be used as well.

IV. Free-form interaction

The basic functionality for Human-Swarm interaction is provided, where all degrees of freedom can be controlled by direct control of the robots’ free-form motion. The user can select sub-groups of robots or individual robots, change their colors and positions, and define trajectories to be followed.

A. Goal mode

Selection and goal assignment: one or more robots can be selected by pointing with the right hand (Figure 3(a)) as described in Section II-B. The centroid of the selected robot group is computed and stored. A new location of the formation centroid can be assigned by pointing with the left hand. If the new location is within the workspace, the robots are commanded towards new positions that maintain the formation with respect to the new centroid[5].

Scaling of the selected robot formation (Figure 3(b)) is achieved by scaling the relative position of each robot with respect to the formation’s centroid by a factor proportional to the change of distance between both elevated hands.

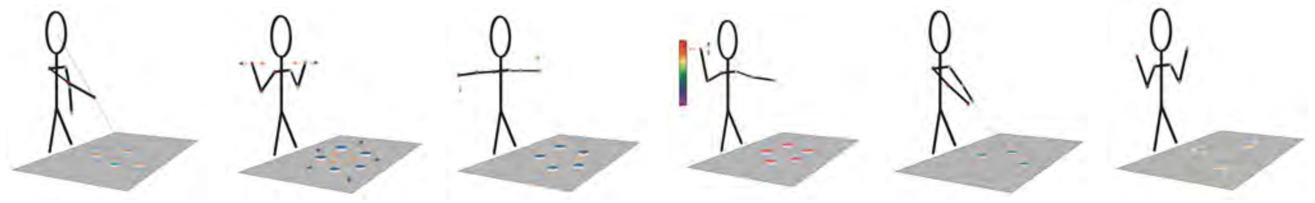
Rotation around the group’s centroid is defined proportional to the rotation of both extended arms with respect to the horizontal (Figure 3(c)).

Reset - it is possible to command the robots to *return to the initial positions* or to cover the initial shape by waving[6]. A waving gesture is complete when the user has moved his right hand four times from side to side.

To *change the colors* of the selected robots the user holds his left hand to the front, at shoulder height (Figure 3(d)). The color hue is defined proportional to the height from the right hand to the shoulder.

[5] Relative positions are used to avoid cumulative distortion.

[6] An initial shape can be given via the real-time drawing interface described in [14].



(a) Select robots (b) Scale the formation (c) Rotate the formation (d) Change colors (e) Trace a trajectory (f) Follow the trajectory

Fig. 3. Free-form interaction. Example of six different free-form gestures. (a)-(d) in goal mode. (e)-(f) in trajectory mode.

B. Trajectory mode

Follow right hand (leader-follower): By holding the left hand up (above the left shoulder) the robots follow the point pointed by the user's right hand. This is the goal position of the global leader (pre-selected robot), which is followed in sequence by the remaining robots that maintain a predefined distance d_0 using a proportional controller. If the user stops moving his right arm the robots also stop.

Trajectory tracing: The user is able to trace a closed trajectory on the ground and command the robots to follow it. The trajectory drawing gesture starts when the user points with both hands to the same point on the ground. The user can then trace a trajectory with his right hand until the starting point is reached (Figure 3(e)). The intermediate points are stored and resampled by interpolation to maintain an approximately constant distance (about 5 cm). The trajectory can cross itself (for example it can be an '∞').

To signalize a recognized trajectory, all robots get green. To command the robots to follow it, the user can hold both hands up (above the shoulders) and the robots will follow it in a loop until the arms are lowered. To follow the trajectory, the robots are first distributed homogeneously over the closed shape and then the position of the next point of the trajectory is set as the goal position for each robot, achieving a homogeneous and constant motion.

V. SHAPE-CONSTRAINED INTERACTION

Shape-constrained interaction provides constrained control of the robot swarm, where a configuration shape is used. The human does not anymore have direct control over a robot, but over a global shape instead. This reduces the degrees of freedom to a few meaningful ones.

Two possibilities arise. First, a single shape (fixed formation) is directly controlled. Second, a given shape (for ex. for optimal coverage) is modified by changing its size, overall shape and position, thus determining the robots' positions.

In an interactive display, shape-constrained interaction facilitates interesting and fun interaction, which shall be intuitive so that the human is focused on the interaction and not on how to make a particular gesture.

A. Fixed formation: Skeleton mode

A fixed robot formation is controlled with representational gestures, where the human is able to slightly modify the formation and define its motion. Different representational gestures could be used to achieve different formations.

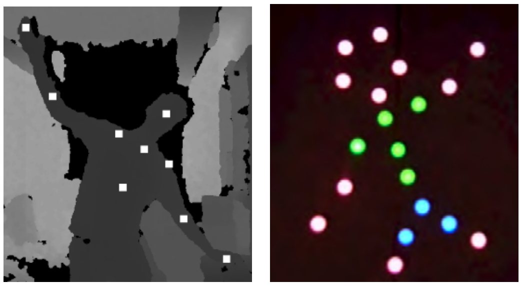
In our interactive display, the robots mimic the movements of the human. The user's joint positions are projected on a plane perpendicular to the depth coordinate, transformed to ground coordinates and each robot follows one specific joint, as shown in Figure 4. Since the sensor is not able to capture the user's legs, they are simulated following the arm movements. For aesthetic reasons, the position of the robots at the shoulders are a linear interpolations between the respective transformed shoulder and elbow joints. The head is a circle with the head joint at its center.

B. Constrained shape morphing: smile mode

For constrained control of the robots a enclosing shape is defined, which can be warped to produce a variation in the configuration. Target shapes could be computed automatically or be predesigned. Mostly manipulation gestures would be employed in this interaction, but could be combined with deictic and representational gestures. For example, the user could first draw an area where the robots are dispersed via optimal coverage [15] and then modify its size, shape and position [14]. Pre-defined shapes could also be employed via representational gestures.

In our interactive display, three shapes (mouth, eyes and eyebrows) are created that describe a face, robots are distributed for optimal coverage [2] and the human controls the parameters of each shape. Different faces can be created with this elements, as shown in Figure 5.

The shape of the mouth is defined by two circular arcs, independent for each side of the mouth and controlled by the shape and position of the user's arms as illustrated in the top-left of Figure 5. For each half of the mouth, the size is given by the distance d from the corresponding elbow to the middle line of the body. The curvature is determined by



(a) Depth image and joints. (b) Robots formation.

Fig. 4. Fixed formation. 19 robots mirror the skeleton of the user.

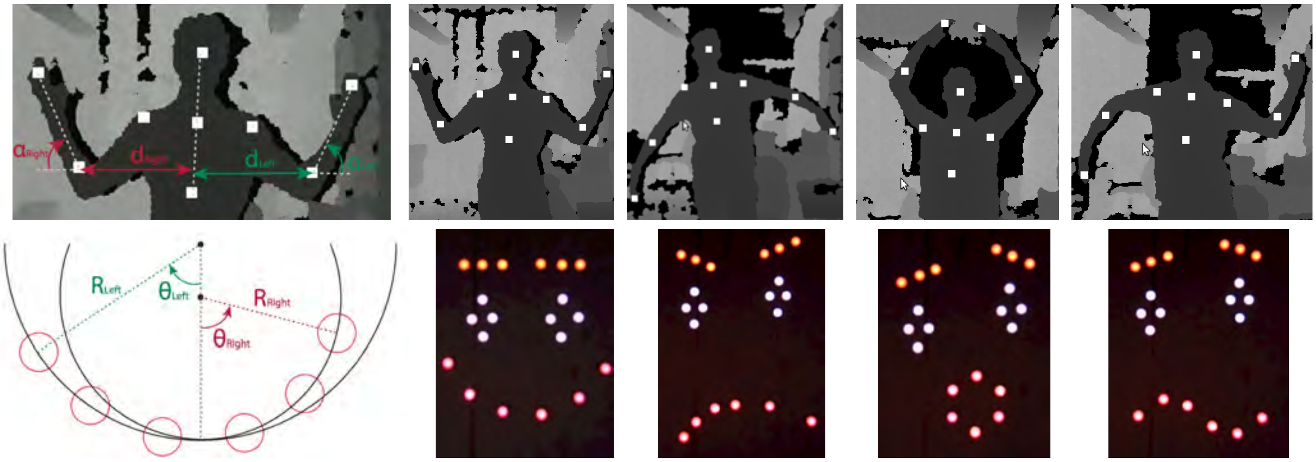


Fig. 5. Constrained shape morphing. *Left most top:* Mouth parameters taken from the depth image. *Left most bottom:* Mouth parameters, radius R (curvature) and angle θ (size). The red circles represent the robots. *Remaining images:* user's gestures (upper row) and the corresponding robots response (lower row). From left to right: smiling face, angry face, circular mouth, confused face.

the angle of the forearm α with respect to the horizontal. The radius of the circle, or the mouth, is given by $R \sim 1/\alpha$ and an angle θ which describes the size of the mouth and is $\theta \sim d/R$ (see bottom left of Figure 5).

The angle α can be negative and the radius R will therefore also be negative, interpreted by the program as a sad/angry face and the mouth is mirrored down as shown in Figure 5. Other predefined forms of the mouth, such as the circular mouth or confused face can be created. If $\alpha > \frac{\pi}{2}$ for both hands and both elbows are above the shoulders, the robots form the circular mouth. If $\alpha > 0$ for one hand and $\alpha < 0$ for the other, the robots shape the confused face.

The position of the eyes can be determined by pointing on the ground for one second on the desired position, each hand controls one eye. The eyebrows are automatically adjusted to each different face. If the user makes a punch gesture, moving his arm fast forwards and backwards, the eye corresponding to the arm used, turns around in a circle and all robots in the face wobble.

VI. Experimental evaluation

A video showing experimental runs of all four modes accompanies this paper. In this section preliminary results on performance and usability are presented.

A. Performance

To exemplify robot response to gestures made by the user, we present experimental results for *trajectory tracing*. Figure 6 shows two ground truth paths (red solid line), where for each one a user traces the path ten times (black dashed lines) and a team of robots follow them (grey solid lines). The average error between the user defined path and the ground truth is 0.036m, with a standard deviation of 0.025m and a maximum error of 0.143m (about 10% of the width). The robots follow the user defined path with errors of mean 0.014m, standard deviation 0.015m and maximum 0.076m. Overall, we observe very good performance, even if the user is given only 2s to draw the path and does so approximately.

B. Preliminary user study

A preliminary user study is done with twenty robots to evaluate the interface, with ten different users (university students from 18 to 30 years old) having no previous exposure to the system. The gestures were explained at the beginning of the study. In the first part, the user was asked to select four out of five robots and direct them to move in a circle. Two robots should be blue and the others should be orange. Several gestures and functions were needed to complete this task - the user had to select robots, change their colors, and change between goal and trajectory mode. To make the circle the user had either to direct the robots to follow the user's right hand and move the hand in a circle, or to define a circular trajectory and let the robots follow it.

Each user had up to four tries and 30% of the users were able to finish the task in the first try, 100% in their second try. The average time to finish the task in the first try was $2:16$ min, while the average time in the fastest try was $1:24$ min. The minimum time over all users was $0:54$ min. The users could score several statements from 0 (completely false) to 5 (completely true). The results are shown in Table II, where

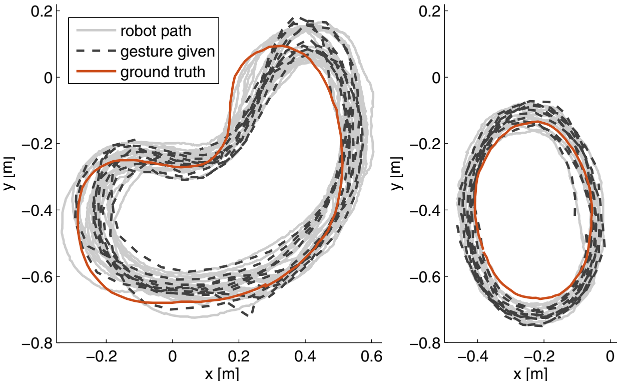


Fig. 6. Examples of robots tracking a trajectory drawn by the user. Each trajectory is drawn ten times by the user, with 2s to complete the gesture.

high is best.

Users did not encounter problems to switch between modes. Two main difficulties were observed. The first was to learn gestures for each specific function, which requires training (they all managed in their second try at most). The second was to do with pointing. People point slightly differently, and there can be an offset between the user's desired point and the system's computed point. One solution for this would be to include a projector in the setup, to provide feedback by projecting the current point computed by the system for the user's pointing direction.

Finally, in the trajectory mode, some users found it confusing to point with both hands to start a trajectory. There were two main reasons: first, some users at the end of the drawing missed the region near the starting point and their trajectory was not recognized. Second, when both hands were too close from each other (requirement to start the drawing), the errors in the hand detection by the sensor increased.

Therefore, an easier gesture was introduced to trace a trajectory. The user can trace a trajectory with his right hand and when a closed shape is detected, it is recognized as a trajectory. Although easier, this gesture has two disadvantages: a trajectory that crosses itself can no longer be recognized and the user might do the gesture unintentionally (the robots will then get green, signalizing a recognized trajectory). This will not cause a movement of the robots, since the user still has to make another gesture for the robots to follow the trajectory. Inexperienced users were much more successful defining a trajectory with this second method.

The second part of the user study was to test the *shape-constrained* interaction. Scored feedback is given in Table II. The skeleton mode was the second most liked feature by inexperienced users (after the trajectory mode), since it is fun and has an easy interaction. To accurately represent the skeleton, the maximum speed the user can move is 0.3 m/s in the current setup, twice the typical speed of the robots.

Overall, users found shape-constrained interaction more intuitive and easier to use than free-form - thanks to the better mapping between commands and robot movement and the lower number of degrees of freedom directly controlled.

VII. Conclusion

This paper describes a taxonomy for real-time interaction between a human and a swarm of robots. Methods are divided into free-form and shape-constrained interaction. *Free-form* provides direct control over to robots, while *shape-constrained* provides constrained control via a reduced set of meaningful degrees of freedom for a target shape configuration. An application of an interactive multi-robot display is described and tested with a team of small robots where gestures are recognized via a depth sensor. All methods were quickly learnt by unexperienced users, who found shape-constrained interaction more intuitive and easier to use.

Statements	Score	
Free-form modes	mean	deviation
It is fun to command the robots	4.8	0.16
The interface is easy to use	3.8	0.56
It is not hard to know individual gestures	2.4	0.44
It is not confusing to change between modes	3.4	0.84
Shape-based modes	mean	deviation
These features are fun	5.0	0
The interface is easy to use	4.2	0.36
The gestures/commands are intuitive	4.0	0.4

TABLE II

Mean scores for interaction modes, maximum score of 5.

One question for future research is whether it is possible to define a standard vocabulary for swarm interaction, say for robot games. In addition, for robots deployed on a projectable surface, we believe that projection could be used both to project user-controllable widgets for the robots and as feedback for a robot's current state and operation. Finally, automated learning to adjust to an individual user's gestures is a route to improve usability, for example with approaches based on Hidden Markov Models [16].

References

[1] P. R. Wurman, R. D'Andrea, and M. Mountz, "Coordinating hundreds of cooperative, autonomous vehicles in warehouses," *AI Magazine*, vol. 29, no. 1, p. 9, 2008.

[2] J. Alonso-Mora, A. Breitenmoser, M. Rufli, R. Siegwart, and P. Beardsley, "Image and Animation Display with Multiple Robots," *Int. Journal of Robotics Research*, vol. 31, pp. 753–773, May 2012.

[3] R. R. Murphy, "Human-robot interaction in rescue robotics," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, vol. 34, no. 2, pp. 138–153, 2004.

[4] J. Suarez and R. R. Murphy, "Hand gesture recognition with depth images: A review," *RO-MAN, IEEE*, pp. 411–417, 2012.

[5] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake, "Real-time human pose recognition in parts from single depth images," in *Computer Vision and Pattern Recognition, 2011 IEEE Conf. on*, 2011.

[6] S. Waldherr, R. Romero, and S. Thrun, "A gesture based interface for human-robot interaction," *Autonomous Robots*, 2000.

[7] S. Bashyal and G. K. Venayagamoorthy, "Human swarm interaction for radiation source search and localization," *IEEE Swarm Intelligence Symp.*, 2008.

[8] M. L. Cummings, "Human supervisory control of swarming networks," *Swarming: Autonomous Intelligent Networked Systems Conf.*, 2004.

[9] R. Parasuraman, T. B. Sheridan, and C. D. Wickens, "A model for types and levels of human interaction with automation," *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, vol. 30, no. 3, pp. 286–297, 2000.

[10] A. Kolling, K. Sycara, S. Nunnally, and M. Lewis, "Human Swarm Interaction: An Experimental Study of Two Types of Interaction with Foraging Swarms," *Journal of Human-Robot Interaction*, June 2013.

[11] G. Podevijn, R. O'Grady, Y. Nashed, and M. Dorigo, "Gesturing at Subswarms: Towards Direct Human Control of Robot Swarms," *IRIDIA – Technical Report Series*, 2013.

[12] J. Alonso-Mora, A. Breitenmoser, M. Rufli, P. Beardsley, and R. Siegwart, "Optimal Reciprocal Collision Avoidance for Multiple Non-Holonomic Robots," *Proc. Int. Symp. on Distributed Autonomous Robotics Systems*, 2010.

[13] M. Karam and m. c. Schraefel, "A taxonomy of gestures in human computer interactions," Tech. Rep. U. Southampton, 2005.

[14] S. Hauri, J. Alonso-Mora, A. Breitenmoser, R. Siegwart, and P. Beardsley, "Multi-Robot Formation Control via a Real-Time Drawing Interface," *Int. Conf. on Field and Service Robotics*, 2012.

[15] J. Cortes, S. Martínez, T. Karatas, and F. Bullo, "Coverage Control for Mobile Sensing Networks," *Robotics and Automation, IEEE Transactions on*, vol. 20, no. 2, pp. 243–255, 2004.

[16] H.-K. Lee and J. H. Kim, "An HMM-based threshold model approach for gesture recognition," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 21, no. 10, pp. 961–973, 1999.