HapticDrone: An Encountered-Type Kinesthetic Haptic Interface with Controllable Force Feedback: Example of Stiffness and Weight Rendering

Muhammad Abdullah, Minji Kim, Waseem Hassan, Yoshihiro Kuroda, and Seokhee Jeon

***Abstract*— HapticDrone is our new approach to transform a drone into a force-reflecting haptic interface. While our earlier work proved the concept, this paper concretizes it by implementing accurate force control along with effective encountered-type stiffness and weight rendering for 1D interaction as a first step. To this end, generated force is identified with respect to drone's thrust command, allowing a precise control of force. Force control is combined with tracking of drone and hand, turning the HapticDrone into an ideal end-effector for the encountered-type haptics and making the system ready for more sophisticated physics simulation. Stiffness and weight of a virtual object is our first target of the simulation. Conventional stiffness and gravity simulation is merged into the encountered-type position control scheme, allowing a user to feel the softness and weight of an object. We further confirmed the feasibility by proving that the system fulfills the physical performance requirements commonly needed for an encountered-type haptic interface, i.e., force rendering bandwidth, accuracy, and tracking performance.**

I. Introduction

A force-reflecting haptic interface generates synthetic force using mechanical actuators and delivers it to a user through physical contact or coupling with a user's body part [1]. In order to convey the generated force to a certain body part, a haptic interface should be affixed elsewhere. Generally haptic devices are classified into either the grounded or ungrounded category. Many conventional haptic interfaces use the ground or earth as a counterpart of the action-reaction principle, so it should be "grounded" to exert grounded-force [2], [3]. In this case, due to the "grounded" nature, the workspace is fixed on the ground and restricted in size by the mechanical limits of the interface. On the other hand, ungrounded haptic interfaces are commonly attached to the user's body, exploiting a body part as a reaction support [4]. In this case, only "relative-force" among body parts can be generated, e.g., grasping force. From user's point of view, both cases significantly decrease the usability of the system, which has been one of the main reasons preventing haptics technology from being widely applied.

Our new concept, HapticDrone, ideally overcomes these usability issues. HapticDrone is an encountered-type drone-based haptic interface. A drone can actively generate kinetic energy and can push and pull a user's hand. If this physical interaction is done in a well-controlled manner, it can be used as a force reflecting haptic interface. The benefits of this concept are significant. First, it does not have to be tethered to anywhere but can still generate grounded-force. We have proved that a typical drone can generate considerably large force [5]. Second, due to its untethered nature, it could potentially generate a limitless workspace, significantly increasing the usability and applicability. Third, if tracking of the drone and user is integrated, it can be turned into an ideal end-effector for an encountered-type interaction, i.e., mechanically coupled with the user's hand only when needed, providing perfect transparency and high usability.

General user interaction with drones has been explored for quite some time. Earlier focus was on teleoperation [6], [7] or controlling visually interactive displays [8]. Recent advances in this field were performed by Cauchard et al. who evaluated user interaction with drones by performing a Wizard of Oz elicitation study [9]. Users were found to interact quite naturally with the drones, treating them as pets or even persons. A recent study was conducted on direct touch interaction with drones by Abtahi et al. [10]. They reported that users felt comfortable performing direct touch interaction with a safe to touch drone.

Haptic user interaction with drones is a relatively new field of study. BitDrones provided the simple interaction of direct touch with drones which represented floating programmable matter [11]. Knierim et al. used the impact force of small accelerating drones to create tactile feedback in virtual reality [12].

Recently, we reported very initial prototypes of the encountered-type drone concept in [13] and [5]. In [13], as proof-of-concept, instead of using direct contact, the air flow from the rotor was used to create a fixed small reaction force (0.118 N) laterally via a flexible sheet of paper attached to the side of the drone. To feel the force, the user interacted with the sheet using a stick for safety reasons. Our recent work in [5] presented new approach that utilized direct contact to generate force. The force output from same drone model was made controllable and improved up to 2.97 N. Simple force-thrust control algorithm was made, and safety cover is attached to allow direct touch.

The current paper directly extends our work in [5]. While [5] is a conceptual introduction, this work realizes and practically demonstrates the concept. Along vertical direction, force vs. thrust is formulated for complete and accurate force control, and based on this formulation, general

Muhammad Abdullah, Minji Kim, Waseem Hassan & Seokhee Jeon are with the Department of Computer Engineering, Kyung Hee University, Yongin, South Korea. {abdullah, mj.kim1102, waseem.h, jeon}@khu.ac.kr

Yoshihiro Kuroda is with the Graduate School of Engineering Science, Osaka University, Japan. ykuroda@bpe.es.osaka-u.ac.jp

Fig. 1. The final design of the HapticDrone with the safety cover and tracking markers in place.

stiffness and weight simulation algorithms are implemented. The feasibility of our implementation is also assessed with three physical performance evaluations, showing a potential of the approach.

The paper is organized as follows. A brief overview of our previous research with details of the HapticDrone system is provided in Sect. II. The haptic rendering methods developed for rendering stiffness and weight using the device are detailed in Sect. III. In Sect. IV different technical aspects of the HapticDrone are evaluated and discussed. The paper is concluded in Sect. V.

II. HAPTICDRONE

HapticDrone is an ungrounded and encountered type device capable of rendering forces in vertical direction. During free movement without contact with a virtual object, the drone can follow the user's movement in all directions, remaining in the vicinity preparing for contact. To create feedback, the drone makes direct contact with the user's hand and pushes it to generate force output when needed, e.g., touching virtual objects.

The Parrot AR Drone 2.0 quadcopter with a safety cover made from aluminum mesh was used as shown in Fig. 1. Detachable contact surfaces in the form of card board or a handle can be mounted on top of the drone as an end-effector for user interaction. The control PC communicates with the drone using a custom python program via Wi-Fi. Occulus Virtual Reality headset and Unity 3D are used to create a virtual environment for the user while Optitrack V120 Trio is used to track the drone, HMD and the user's hand. The complete setup is shown in Fig. 2. The following subsections explain the experiments conducted to find the force rendering equations and the types of modes developed to make an encountered type interaction.

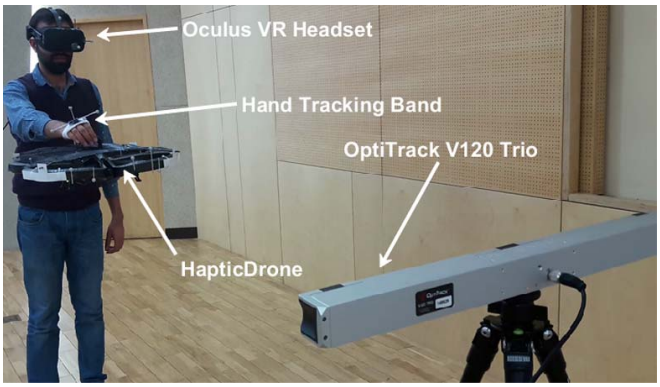


Fig. 2. The user is interacting with the drone while wearing the HMD and a tracking band. Optitrack is used to track the drone and the user's movement.

A. Input Command to Force Output Mapping

To control the force output we conducted an experiment to find the relationship between the input command value for the thrust control and the rendered force. The force applied by the drone is directly proportional to the thrust command. So we experimentally found a direct mapping from the normalized thrust command to exerted force.

1) Procedure: The normalized thrust command, with predefined incremental values, was used as input to the system and force values were recorded as the output. According to the SDK of the AR drone, normalized vertical thrust (from -1 to 1; 0 for hovering) can be commanded as an input to the drone. A push/pull digital force gauge (SF-20; Wenzhou Tripod Instrument Manufacturing Company) with 0.01 N sensing resolution was used to measure the force. The testing range was from -1.0 to 1.0, with increments of 0.05 resulting in 41 intervals.

2) Results: Figure 3 shows the measurement data. Second order polynomial fitting is performed on the data, resulting in (1) for upward and (2) for downward direction. The drone becomes unstable for input thrust above 0.85 and produces no tangible force between thrust values of -0.1 and 0.1. The maximum stable upwards force that the system generates is 1.53 N, while the maximum downward force is 2.97 N. For comparison, the Geomagic Touch (a grounded device) can produce 3.3 N of force [14]. The polynomials are as follows.

$$y_1 = 0.3628x_1^2 - 0.0607x_1 + 0.0905 \quad (1)$$

$$y_2 = -0.1000x_2^2 + 0.0115x_2 - 0.0798, \quad (2)$$

where the inputs are the required force outputs x_1 and x_2 in the upwards or downwards direction respectively, while the resultant input speeds needed to generate them are y_1 and y_2.

B. Modes of Operation

To create an encounter type interaction using the drone, two different control modes are implemented. The drone follows the user's hand in *non-contact mode*, keeping a certain

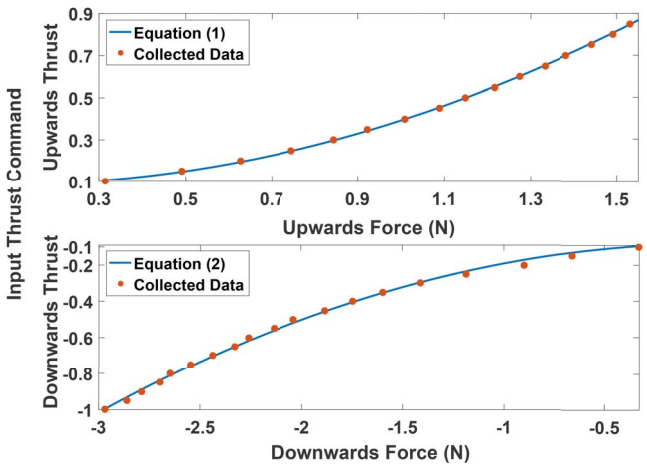


Fig. 3. The result of input command to force mapping, showing the dynamic range of the drone. The thrust input range between -0.1 and 0.1 does not produce tangible results because of slow speed. Thrust values above 0.85 create unstable output. The resultant equations are used in the stiffness and weight renderings.

distance until a virtual object is encountered, whereupon, it makes contact (mode switching) and renders a force based on the requirement in *contact mode*.

1) Non-Contact Mode: In this mode, the drone follows the user's hand maintaining a distance of 100 mm in the vertical direction. The position of the drone is controlled under a closed-loop control scheme with experimentally tuned proportional-integral-derivative (PID) gains. The custom control scheme caters for the under actuation of the drone due to the added mesh cover.

2) Mode Switching: When the user is about to make contact with a virtual object, the drone maintains its position to the nearest surface point, ignoring the 100 mm offset. When the contact happens, the drone switches to contact mode and renders the required force.

3) Contact Mode: When the drone comes in contact with the user's hand it switches to open loop control. This is necessary to focus all its energy on generating force in a certain direction. It renders force based on the rendering equations described in Sect. II-A.

III. Stiffness and Weight Rendering

We have implemented two different practical scenarios exploring the capabilities of the HapticDrone design, shown in Fig. 6. The renderings are enabled by the HapticDrone in its own distinctive manner; as providing considerable contact reaction force (2.97 N) in midair, with no grounding support, is difficult using existing technologies. In the first scenario, we design a mid-air 1D haptic display that creates virtual objects with programmable stiffness. The second scenario implements haptic weight rendering and presents the user with virtual objects of adjustable weight. Both the scenarios present different challenges during the mode switching discussed in Sect. II-B. A modification to the control algorithm has to be applied, which is mentioned in their respective sections. The user interaction takes place in a VR environment created in Unity 3D.

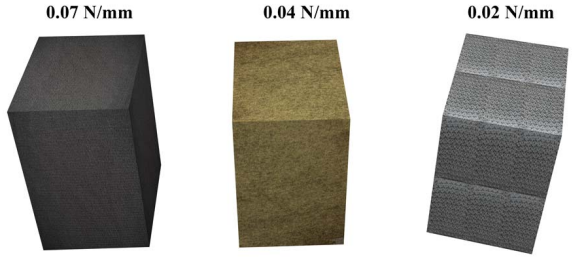


Fig. 4. The virtual objects available to the user having different stiffness coefficients for the stiffness rendering.

A. Programmable Stiffness Rendering

This section provides detail on stiffness rendering using the drone. The user can touch virtual boxes and feel their stiffness as shown in Fig. 6 (a, b). Three different virtual objects with different stiffness coefficients are available for the users to interact as shown in Fig. 4. In the non-contact mode, the drone follows the users hand movements, waiting for them to interact. When the user touches one of the virtual objects, collision detection (mode switching) is performed. The drone enters the contact mode and applies force to the user's hand based on the linear stiffness model, $f = kx$ and the penetration depth on the virtual object calculated from the tracking data. Equation (1) for the drone's upwards force is used to render the required force f. The algorithm saturates the force to its maximum if the calculated force exceeds beyond the dynamic range.

When the user leaves the simulation area the drone switches back to non-contact mode. During the mode switching the drone needs to apply a large negative force to immediately disconnect from the user's hand. The weights of the PID are temporarily changed to compensate the drone's inertia. This scenario is developed for linear stiffness model, but it is possible to use more sophisticated stiffness models, e.g., Hunt-Crossley nonlinear stiffness model.

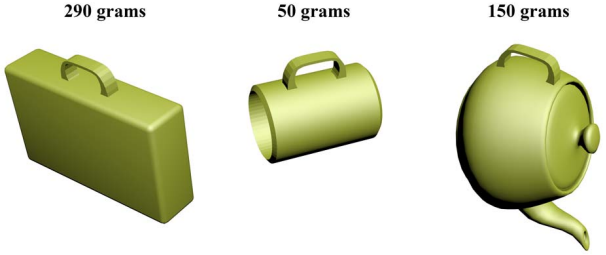
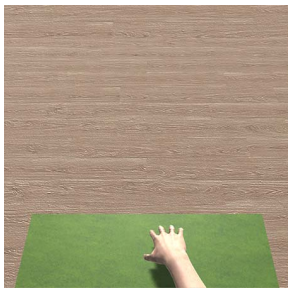


Fig. 5. The example virtual objects with different weight created for the weight rendering.

(a) The user experiencing stiffness of the virtual object by touching the contact area on the drone. The force is rendered in the upward direction.



(b) The virtual environment as seen by the user. The object deforms according to the movement of the user's hand and the drone.

(c) The user experiencing the weight sensation from the drone, by grasping the handle. Here the drone applies force in the downward direction.

(d) The object as seen by the user in the virtual reality. The object moves according to the position of the drone.

Fig. 6. The stiffness rendering and weight rendering scenarios, they are implemented using the 1D force feedback from the HapticDrone.

B. Haptic Weight Rendering

In this scenario, a user can feel the weight of a virtual object while grabbing and lifting it, as demonstrated in Fig. 6 (c, d). The weight of an object and the gravity is treated as a variable. Equation (2) for the drone's downwards force is used to create the feeling of weight. A selection of the objects presented in the virtual environment are shown in Fig. 5. Users can slide their hand into the handle attached to the top of the drone and lift the object to feel its weight.

In non-contact mode, the drone moves itself based on the position of virtual object. When the user grasps the handle to lift the object, the drone enters contact mode applying a downwards force according to the weight of the virtual object. The maximum downwards force possible is 2.97 N. This is about the weight of a filled soda can.

As soon as the object is returned to its origin or is dropped by the user, mode switching is activated. During this process, the weights of the PID are dynamically adjusted based on the drone's instantaneous speed. This is necessary to make sure the drone applies sufficient force in the upward direction to regain its position and not fall to the ground. After the drone regains its altitude, the normal non-contact mode is established.

IV. Technical Evaluation

As a haptic interface, HapticDrone should fulfill certain performance requirements. As drones are relatively new in haptics, there are no performance criteria to be applied. Hence we are providing a comprehensive set of evaluations that can set a baseline for future research.

A. Force Rendering Bandwidth

The maximum exert-able force range is already discussed in Sect. II-A. Another important aspect is the force rendering bandwidth, i.e., how fast the rendered force follows the command force. As discussed in Sect. II-A the thrust command is directly proportional to the force output of the drone. As the drone is a mechanical device, thus it takes some time to reach the desired thrust after the input command is given. So a delay exists between the command to render a force and the actual rendering time.

1) Setup: From basic drone mechanics, thrust command is directly proportional to the rotating speed of the rotors (rpm) with small delay. Since it can be assumed that rpm is instant representation of exerted force, measuring the changing speed of this rpm with respect to the change of thrust command can represent the response speed of the rendered force. For this measurement, we put a step input on the thrust command and measured the response delay on rpm. The current drone's API can relay normalized rpm values to the control PC with local time stamp, which we used for measuring rpm.

2) Procedure: In order to examine the force response delay, we gave a step thrust command (from 0 to 1) to the drone. This changes the rotor's speed from the lowest rpm (hovering) to full rpm (rising at maximum velocity). The thrust command was also given in the downward (from 0 to -1). This experiment was conducted ten times for each direction of force.

3) Result: The longest response time recorded for the drone was 0.043 s.

4) Discussion: The force rendering bandwidth of the HapticDrone system depends on the component with the lowest response time. The sampling rate of Optitrack V120 Trio is 200 Hz while the control loop is updated at 250 Hz. However, the rotors of the drone which provide lift and consequently the force output are mechanical in nature. Thus they require some time to adjust their speed. The longest response time of the drone was found to be 0.043 s, which amounts to 23.6 Hz. The human movement bandwidth is reported to be about 10 Hz [15]. Thus the force response is fast enough for conventional haptic rendering algorithms that use closed-looping of human interaction motion. However, the device can face limitations when implementing scenarios that requires rendering of rapid changing force, e.g., a highly rigid virtual wall. This can affect the stability of a haptic device. Devices with slow response time can become unstable when rendering high impedance surfaces. One method currently used to increase the stability is the inclusion of electrical or mechanical damping [16].

In our case, as mentioned in Sect. III-A, if the required force is above the dynamic range we saturate the output

force to maximum. However, even at maximum force we did not observe any oscillation or instability in the device. This can be attributed to the ungrounded nature of the drone and the fact that it generates force by applying thrust and thus virtually moves in the required direction. This directional movement is inherently damped by the drones internal PID control loop to prevent movement instability. It must be noted that this PID loop is different from the one we implemented for *non-contact mode* and is active at all times. Generally, the force rendering bandwidth can be increased by using better quality rotors with lower response time.

B. Force Output Accuracy

In this section, the force rendered by the system was scrutinized by measuring the error between the commanded and the actual force.

1) Setup: For this experiment, the drone was programmed to render multiple forces with different values. The force values ranged from 0.4 N to 1.4 N with an interval of 0.2, resulting in a total of 6 values. For measuring the rendered force, a digital force gauge (SF-20; Wenzhou Tripod Instrument Manufacturing Company) with 0.01 N sensing resolution was used.

2) Procedure: The gauge was used to interact with the drone, mimicking the movement of a user's hand. The experiment was performed ten times for each force value.

3) Result: The measured values were compared with the commanded force and the results are plotted in Fig. 7. The results show that absolute mean error remains almost constant across the experiment (2.72%). Out of all the test forces the highest corresponding relative mean error was 13.04% which appeared for 0.4 N.

4) Discussion: The just noticeable difference (JND), also known as the difference threshold is the smallest amount of physical difference that can be perceived. The human JND for force perception is about 10% [17], [18]. For the lowest test value (0.4 N), the maximum relative mean error was 13.04% which is slightly above the JND (3.04%). This discrepancy can be attributed to the fact that the drone has problems creating force output at low thrust values as mentioned in Sect. II-A. The results also show that the absolute mean error remains almost constant across all test values. This causes the relative mean error to decrease as the test force increases. For the remaining test values above 0.4 N the maximum relative mean error is lower than the JND and can be considered perceptually insignificant.

C. Movement Bandwidth and Accuracy in Non-Contact Mode

The HapticDrone is primarily an encounter type haptic interface. A critical performance criterion for such devices is the movement accuracy and agility of the end-effector in non-contact mode. Due to the extra weight of the safety cover, the native speed of the drone is reduced. Thus the custom PID loop which accounts for the extra weight of the cover and the user's hand movements needs to be evaluated. Therefore, we conducted an experiment to examine the movement accuracy of the system.

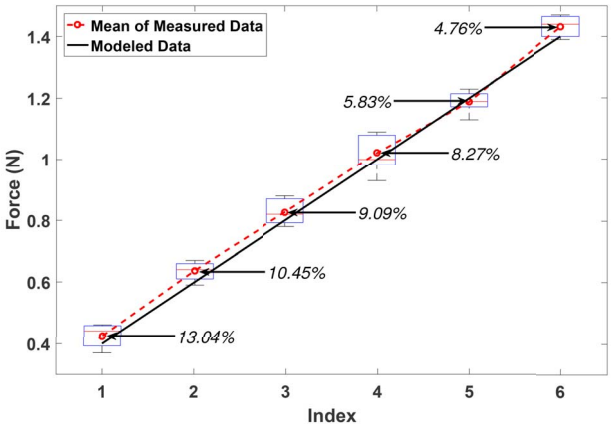


Fig. 7. Commanded force values compared against the actual measured force. The arrows report the maximum relative mean error for each force value.

1) Setup: In the non-contact mode the drone follows the users hand in all direction i.e *x*,*y* and *z*. It is supposed to keep the *x* and *y* position of the hand exactly but maintain an offset of 100 mm in the *z* direction. The test was to ascertain error in the target position of the drone due to movement speed of the user's hand. Six participants took part in the experiment. The participants were asked to stand in a fixed spot. The drone was put into non-contact mode and hovered approximately 400 mm away from the participants. The drone was restricted to a virtual bounding box of 500 mm^3. This was done to maintain the experimental conditions of an outstretched human arm.

2) Procedure: The participants were asked to move their hands left, right, up and down at a comfortable speed and imagine reaching for something. Each participant was engaged for approximately five minutes. The positions of the drone and the user's hand were recorded.

3) Result: Results show that the maximum offset error was in the *z* axis at 71 mm. The drone's maximum recorded speed was 1189 mm/s, while the maximum speed of the participant's hand was 890 mm/s. As the maximum error encountered was lower than 100 mm, it is assured that the drone will never touch the user's hand during non-contact mode despite some offset error due to quick changes in direction. In addition, it is also inferred from the data that position control bandwidth of the drone exceeds that of human hand movement in normal interaction.

4) Discussion: According to Jones et al., the normal human hand movement speed for reaching and grasping an object at a respectable distance of 400 mm is approximately 800 mm/s [19]. In our case the modified drone moved at speeds higher (1189 mm/s) than our recorded participants (890 mm/s) but it still faced offset errors. On the surface it would appear that the drone can match the movement speed of the user's hand. However, as the drone has added weight it has to deal with increasing inertia at higher speed. After analyzing the data, we found that whenever the users changed their movement direction quickly the drone struggled to match the opposite acceleration. These errors can be further

reduced by building a better PID loop with a higher update rate. The recorded offset error remained below the threshold of 100 mm with no unintentional contact. This qualifies the HapticDrone as an encountered type device.

D. General Discussion

An important aspect of the device is its operation time. For the current system it is about 15 minutes on average. Lower battery usage was observed during the weight rendering scenario as it slows down the rotors, using the drone's weight to apply downwards force. One method to solve the battery problem is to use multiple drones and a self-docking and inductive charging mechanism. The HapticDrone is an encountered type device and is not visually rendered in the virtual environment. Therefore, the user will be visually unaware if one drone leaves while another takes its place. The sound emitted by the drone is another issue. The sound can be heard through headphones and the Occulus HMD. One quick solution is to use noise-canceling headphone during interaction. More fundamental solution would be eliminating noise sources in the drone mechanism, e.g., adding dampers in the rotor joints and using noise-free mechanical parts.

In the future the concept can be extended towards unconstrained 3D interaction. The drone can provide three degree of freedom force rendering with each direction of motion. Torque rendering may also be possible by rotating the drone but will require thorough analysis of drone dynamics with external contact. The equations formulated in Sect. II-A are specific to our device but the procedure is not. The same steps can be followed by researchers building their own haptic device using any drone.

V. Conclusion

In this research, we presented the implementation of stiffness and weight rendering on the HapticDrone device. These scenarios prove that adequate haptic rendering is possible with the device and lay the ground work for more sophisticated applications in the future. We also performed technical evaluations and discussed the results in perspective of human perceptual requirements. Our findings are the first to be reported for this kind of device. These results can be used as a baseline comparison for new drone based haptic devices.

Acknowledgments

This work was supported by the NRF of Korea through the Basic Research program (NRF-2017R1D1A1B03031272).

References

[1] K. Salisbury, F. Conti, and F. Barbagli, "Haptic rendering: introductory concepts," *IEEE Computer Graphics and Applications*, vol. 24, no. 2, pp. 24–32, March 2004.

[2] T. H. Massie, J. K. Salisbury, *et al.*, "The phantom haptic interface: A device for probing virtual objects," in *Proceedings of the ASME winter annual meeting, symposium on haptic interfaces for virtual environment and teleoperator systems*, vol. 55, no. 1. Citeseer, 1994, pp. 295–300.

[3] R. Q. Van der Linde, P. Lammertse, E. Frederiksen, and B. Ruiter, "The hapticmaster, a new high-performance haptic interface," in *Proc. Eurohaptics*, 2002, pp. 1–5.

[4] M. Bouzit, G. Popescu, G. Burdea, and R. Boian, "The rutgers master ii-nd force feedback glove," in *Proceedings 10th Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems. HAPTICS 2002*, 2002, pp. 145–152.

[5] M. Abdullah, M. Kim, W. Hassan, Y. Kuroda, and S. Jeon, "Hapticdrone: An encountered-type kinesthetic haptic interface with controllable force feedback: Initial example for 1d haptic feedback," in *Adjunct Publication of the 30th Annual ACM Symposium on User Interface Software and Technology*, ser. UIST '17. New York, NY, USA: ACM, 2017, pp. 115–117. [Online]. Available: http://doi.acm.org/10.1145/3131785.3131821

[6] S.-w. Leigh, H. Agrawal, and P. Maes, "A flying pantograph: Interleaving expressivity of human and machine," in *Proceedings of the TEI '16: Tenth International Conference on Tangible, Embedded, and Embodied Interaction*, ser. TEI '16. New York, NY, USA: ACM, 2016, pp. 653–657. [Online]. Available: http://doi.acm.org/10.1145/2839462.2856347

[7] M. Matrosov, O. Volkova, and D. Tsetserukou, "Lightair: A novel system for tangible communication with quadcopters using foot gestures and projected image," in *ACM SIGGRAPH 2016 Emerging Technologies*, ser. SIGGRAPH '16. New York, NY, USA: ACM, 2016, pp. 16:1–16:2. [Online]. Available: http://doi.acm.org/10.1145/2929464.2932429

[8] J. Scheible, A. Hoth, J. Saal, and H. Su, "Displaydrone: A flying robot based interactive display," in *Proceedings of the 2Nd ACM International Symposium on Pervasive Displays*, ser. PerDis '13. New York, NY, USA: ACM, 2013, pp. 49–54. [Online]. Available: http://doi.acm.org/10.1145/2491568.2491580

[9] J. R. Cauchard, J. L. E, K. Y. Zhai, and J. A. Landay, "Drone & me: An exploration into natural human-drone interaction," in *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, ser. UbiComp '15. New York, NY, USA: ACM, 2015, pp. 361–365. [Online]. Available: http://doi.acm.org/10.1145/2750858.2805823

[10] P. Abtahi, D. Y. Zhao, L. E. Jane, and J. A. Landay, "Drone near me: Exploring touch-based human-drone interaction," *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, vol. 1, no. 3, pp. 34:1–34:8, Sept. 2017. [Online]. Available: http://doi.acm.org/10.1145/3130899

[11] A. Gomes, C. Rubens, S. Braley, and R. Vertegaal, "Bitdrones: Towards using 3d nanocopter displays as interactive self-levitating programmable matter," in *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, ser. CHI '16. New York, NY, USA: ACM, 2016, pp. 770–780. [Online]. Available: http://doi.acm.org/10.1145/2858036.2858519

[12] P. Knierim, T. Kosch, V. Schwind, M. Funk, F. Kiss, S. Schneegass, and N. Henze, "Tactile drones - providing immersive tactile feedback in virtual reality through quadcopters," in *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems*, ser. CHI EA '17. New York, NY, USA: ACM, 2017, pp. 433–436. [Online]. Available: http://doi.acm.org/10.1145/3027063.3050426

[13] K. Yamaguchi, G. Kato, Y. Kuroda, K. Kiyokawa, and H. Takemura, "A non-grounded and encountered-type haptic display using a drone," in *Proceedings of the 2016 Symposium on Spatial User Interaction*. ACM, 2016, pp. 43–46.

[14] Geomagic, "The geomagic touch (formerly sensable phantom omni)," 2017, http://www.geomagic.com/en/products/phantom-omni/overview.html.

[15] H. Z. Tan, M. A. Srinivasan, B. Eberman, and B. Cheng, "Human factors for the design of force-reflecting haptic interfaces," *Dynamic Systems and Control*, vol. 55, no. 1, pp. 353–359, 1994.

[16] J. S. Mehling, J. E. Colgate, and M. A. Peshkin, "Increasing the impedance range of a haptic display by adding electrical damping," in *First Joint Eurohaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems. World Haptics Conference*, March 2005, pp. 257–262.

[17] L. A. Jones, "Matching forces: constant errors and differential thresholds," *Perception*, vol. 18, no. 5, pp. 681–687, 1989.

[18] X.-D. Pang, H. Z. Tan, and N. I. Durlach, "Manual discrimination of force using active finger motion," *Attention, Perception, & Psychophysics*, vol. 49, no. 6, pp. 531–540, 1991.

[19] L. A. Jones and S. J. Lederman, *Human hand function*. Oxford University Press, 2006.