

SkySculptor: Intuitive Drone Control Through Ground-Integrated Radar and Foot Gestures in Smart Indoor Environments

Alexandru-Ionuț Șiean
MintViz Lab, MANSiD Research Center
Stefan cel Mare University of Suceava,
720229 Suceava, Romania

Abstract—SkySculptor is a software application designed to optimize drone control in smart indoor environments. The primary focus is on using gesture input for drone control, particularly investigating mid-air free-foot interactions detected by radar sensing. This software application simplifies the process of controlling drones in smart indoor environments. Additionally, outcomes of utilizing a 15-antenna ultra-wideband 3D radar are presented, establishing a dictionary of six directional swipe gestures for controlling drone functions. Based on the findings of this research article, guidelines for the future development of software applications for drone control in intelligent indoor environments are proposed.

Keywords—Drone control; gesture input; ultra-wideband radar; software application; smart environments

I. INTRODUCTION

Efficient, fluid, and expressive interactions that take advantage of body position and movement can be achieved by integrating radar gestures into smart indoor environments. This involves recognizing body position and movement and employing taxonomies to incorporate radar sensing technology [1]. This integration facilitates various interactions, including controlling smart TVs [2], utilizing wearable devices for gesture detection [3], and drone interactions [4]. Airborne gestures can also interact with content displayed on ambient screens [5], while foot movements offer additional avenues for interaction, covering both natural and unnatural gestures [6], [7]. Symbolic touch gestures commonly used on mobile devices can also be incorporated [8].

Recognizing user gestures involves a combination of hardware devices such as accelerometers embedded in smart devices like smart rings [3], [9], [10], various recognition algorithms (e.g., KNN, Random Forests, Convolutional Neural Networks, DTW) and data sets establishing correlations between gestures and system functions. Radar devices such as Google Soli, Walabot, and Sense2Go exemplify non-contact gesture sensors that offer new interaction possibilities with smart environments. Furthermore, attaching a radar device to a drone enables the control of various objects in a smart indoor environment [11]. Radar-detected gestures can range from basic presence and proximity detection to multitouch gestures or foot-based gestures [12], [13], [6]. Studies on radar device interactions suggest various sets of gestures, some focusing on hand gestures [14], [15] while others advocate interactions that involve the whole body [16]. These examples highlight

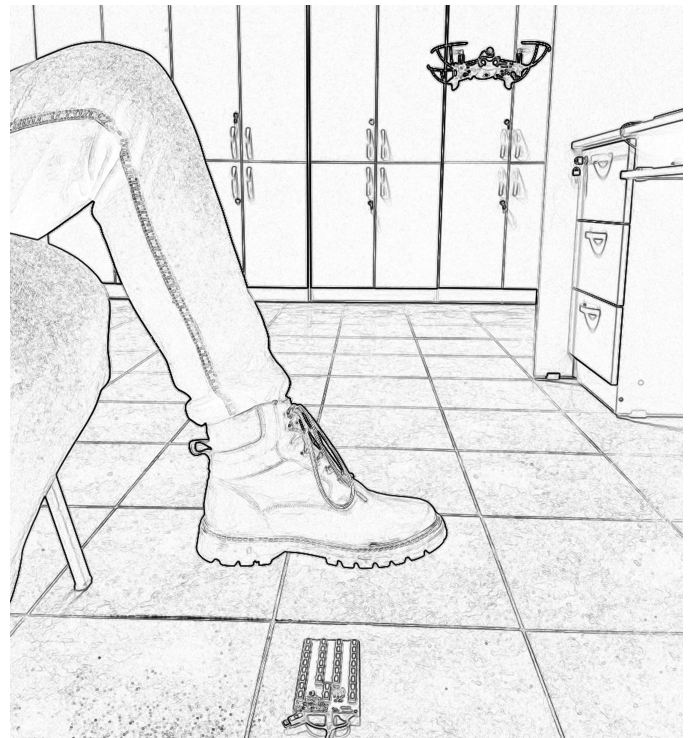


Fig. 1. The walabot radar device, used in SkySculptor, is shown alongside the parrot mambo drone and interaction modes.

that radar device interactions typically involve the upper body, as detailed by Șiean *et al.* [1]. Despite these advances, the lower part of the body was overlooked in interaction with radar devices.

In our contribution, emphasis was placed on the following aspects:

- 1) Software design requirements were defined and incorporated into the design of an interactive system that enables drone control using a radar device integrated, attached, or placed on the floor of a smart environment.
- 2) A set of six radar gestures executed with the foot was proposed, each associated with specific commands for drone control.
- 3) An implementation of the software application using a radar with a 15-antenna ultra-wideband 3D radar.

II. RELATED WORK

Gesture-based drone control and radar-based gesture sensing and recognition will be our focus.

A. Gesture-based Drone Control

Research on gesture-based human-drone interaction has explored various approaches [17], [18]. Gestures serve as an effective means of conveying emotions, thoughts, and nonverbal communication, and hand gestures are widely accepted as intuitive communication methods [19]. Controlling drones through hand gestures offers simplicity and intuitiveness. Past studies have investigated hand gesture recognition using vision-based sensors such as RGB and depth cameras, as well as Microsoft Kinect sensors [20]. However, recent developments in safe-to-touch drones have spurred interest in novel forms of interaction, including direct touch and manipulation. For example, HoverBall is a ball-shaped quadcopter capable of sports or game applications [21]. Drones have also been used as haptic feedback devices in virtual reality scenarios [22]. However, these systems face limitations related to environmental factors such as light conditions, viewing angles, and spatial positioning.

B. Radar-based Gesture Sensing and Recognition

The core of radar detection lies in the use of electromagnetic waves. When these waves strike a target, a portion of the signal reflects back to the radar source and the receiver captures the signals. The characteristics of the received signal, such as frequency, amplitude, and delay time, provide information about the detected object, including its shape, orientation, distance, and speed relative to the radar. Radars are advantageous for human detection (e.g. gestures, posts, movement, etc.) as they operate under conditions of high illumination, low lighting, or darkness, when obstructed by surfaces or objects [23], [24], or during different weather conditions [25]. Previous research has presented and evaluated multiple techniques to recognize radar gestures. For example, Gigie *et al.* [14] showcase a case study of data explosion for radar-based human gesture detection. They introduced a simulation framework based on a physical model to generate radar signals corresponding to various human gestures. With the availability of Google Soli in the mobile ecosystem, Leiva *et al.* [26] devised a hybrid CNN+LSTM deep learning model. They conducted a comprehensive study exploring the performance of mid-air gesture recognition while covering the radar sensor with three distinct fabrics (leather, wool, and cotton). The model demonstrated an exceptional average performance accuracy of 95%, AUC of 99%; RadarNet [27] utilizes an effective recognition approach for radar gestures, utilizing a Convolutional Neural Network. It operates efficiently on processors with limited computational capabilities. On the other hand, mHomeGes [28] is a system designed to detect radar gestures with an accuracy of 95%, particularly designed for smart home interactions. Therefore, we propose a radar sensor-based gesture recognition system for drone control that extends body gestures using *foot* gestures.

III. DESIGN REQUIREMENTS FOR SKYSCULPTOR

By analyzing the relevant scientific literature and considering the potential application domain, which focuses on the

interaction between drones and gesture-based foot movements in smart indoor environments using integrated, attached or floor-placed radar devices, we can establish the necessary design requirements for SkySculptor as shown in Fig. 1. Next, the design criteria for the development of the SkySculptor software application.

- (a) *Open source technology*: SkySculptor strongly emphasizes the adoption of open technologies to promote the development and progress of innovative specialized software applications designed for users who operate drones in smart indoor environments. This guiding principle also applies to SkySculptor, which features a fully developed software application created using open technologies. The goal of this system is to encourage research and support the ongoing advancement of drone interactions in intelligent indoor environments through the use of radar gestures.
- (b) *Smart spaces orientation*: Accurate positioning of the radar device in the recommended locations, as proposed by Şiean *et al.* [1], enables effective handling of complex gestural interactions between users. Similarly, for interactions between drones and physical objects, the approach suggested by [2] can be applied. This approach is seamlessly integrated into the software application SkySculptor, with a particular emphasis on the areas of the floor and the interaction facilitated by the movements of the feet.
- (c) *Easily integrate, attach, or place the radar sensor on objects or devices*: Traditionally, drone control has been done using joysticks or smartphones. However, in our scenario, we propose a shift towards using an integrated radar device that can be attached to or placed on the floor for drone interaction. Radar devices have unique features that allow them to integrate seamlessly into various objects or be attached by users, providing researchers with the opportunity to propose innovative methods of interaction. The SkySculptor software application places an emphasis on integrating the radar device in the floor and facilitating interaction with the feet. This is motivated by the lack of gestures involving the lower body, despite the numerous proposals for interaction focused on the upper body, as discussed by Şiean *et al.* [1]. By prioritizing the integration of the radar device into the floor and enabling interaction based on leg movements, we aim to enrich and diversify gestures and interactions in intelligent indoor environments within the context of HDI.

Requirement (c) can be satisfied by utilizing the principles that govern the functioning of radar devices and their ability to detect objects. This requirement provides various options for the installation of the radar device, allowing interaction with the drone. At the same time, it addresses the challenges associated with gesture recognition that may arise due to low light or unfavorable weather conditions. When open technology is employed in the development of a software application, it becomes easier to extend and improve its functionality. As a result, a new requirement is introduced for SkySculptor, which specifically describes the ease of its development.

- (d) *Python-orientation*: Devices that are capable of running

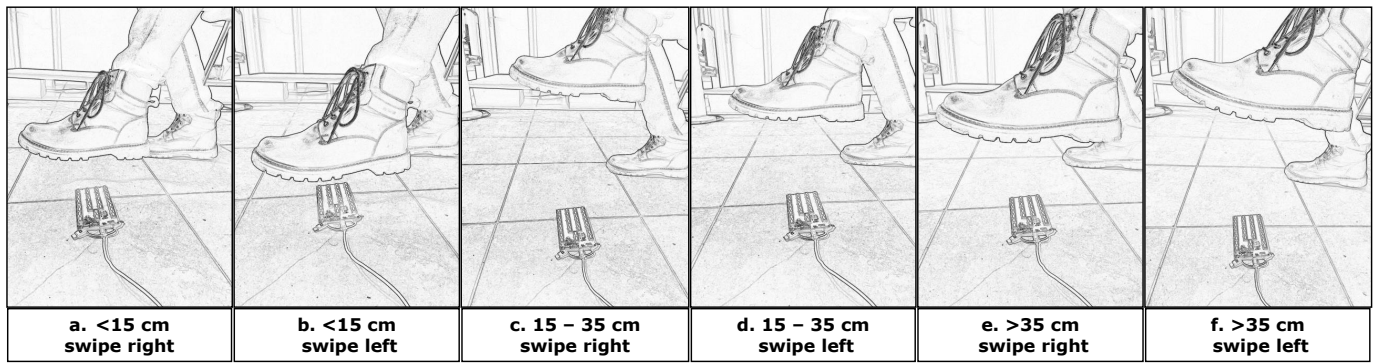


Fig. 2. Various mid-air gestures combining directional swipes and distance from the sensor enable control of drone functions.

SkySculptor are those that have a Python interpreter and are equipped with a USB port for connecting the radar device.

Software applications developed using the Python framework are well known for their fast and reliable performance. This decision was made to ensure that the tools are compatible with various devices that have a Python interpreter, providing a consistent experience for users. Using Python¹, the process of integrating drone and radar device SDKs became easy, making it easier to connect to these devices. Using the drone SDK² and establishing a connection to the radar device, we successfully incorporated the necessary functionality to control the drone without any difficulty.

IV. IMPLEMENTATION APPROACH AND PROTOTYPE

We propose a gesture recognition system to control drones using radar sensors. The system focuses on using foot gestures for control. To evaluate the technical feasibility of this radar sensor-based gesture recognition system for drone control, taking into account the availability and affordability of radar technology on the market, a prototype SkySculptor was developed using the 15-antenna Walabot Creator device and the Walabot API.³ The Mambo Parrot drone was used in conjunction with the prototype. The trajectory of the detected target above the radar was captured and expressed as x , y , and z coordinates. Two directional swipe gestures (referred to as *swipe left* and *swipe right*) along the y axis. Furthermore, the distance from the sensor on the z axis at which these gestures were made was used to define three active zones (*near*, *close*, and *far*) above the radar. Fig. 2 shows visual representation of the gesture set. When the two directions and three zones were combined, a total of six gestures were obtained. These gestures can be assigned to three types of functions commonly used in drone control: *take-off/land* for initiating or ending the flight of the drone, *start/stop video* for controlling the camera, such as taking photos or starting/stopping video recording, and *forward/backward* for moving the drone in the forward or backward direction. This method is the same as the paper [2].

¹The Python programming language is widely recognized as one of the most popular programming languages. It consistently holds the top position in the Tiobe ranking (<https://www.tiobe.com/tiobe-index/>), with a steady rating of 13.97% from 2023 to 2024.

²<https://developer.parrot.com/docs/index.html>

³https://api.walabot.com/_sample.html

We conducted a preliminary evaluation of our application by performing foot gestures that corresponded to various scenarios, including take-off, landing, starting and stopping video recording, moving forward, and moving backward a drone. A visual representation of these gestures can be seen in Fig. 2. If additional gestures are required, modifications to our basic gesture recognition pipeline may be necessary, such as preprocessing the raw signal or implementing new recognition techniques [29]. Fig. 3 displays $\theta-R$ images obtained from the Walabot radar when placed on the floor and corresponding to the gestures shown in Fig. 2, with varying distances measured by the radar sensor.

The Walabot radar sensor was placed on the floor at various locations to generate the heatmap screenshot for different scenarios. This placement was chosen to allow for interaction with the feet. In each scenario, the experimenter sat on a chair and used his right foot to perform gestures (a) and (b) within the active zone labeled *near*. For scenarios (c) and (d), a mouse pad was placed on the floor with the Walabot radar sensor underneath. To represent the situation in which the radar sensor is placed below an object such as a carpet, the experimenter raised his foot higher on the z axis and made gestures for the active zone labeled *close*. In the last case, the radar sensor was placed on a vibrotactile floor, which is available in our research laboratory. The experimenter executed gestures (e) and (f) to simulate the scenario in which the sensor is integrated into the floor within the active zone labeled *far*. This method is the same as the paper [2].

Another assessment was made when the experimenter was seated, allowing interaction through knee gestures. By sliding the knee to the left and right at varying distances from the radar sensor, the experimenter could access three active areas: *near*, *close*, and *far* for interacting with drone. The Walabot radar device was placed on the wall, 20 cm above the floor. Alternatively, when the experimenter was not seated, more extensive sliding movements of the left or right foot could be performed to execute foot gestures. Fig. 3 illustrates foot gestures, the distance measured by the radar sensor, and the different placements of the sensors. For more examples of where to place radars, we recommend looking at [1].

Different types of data can be extracted from a radar system, such as velocity, range, and direction of motion. The specific details obtained vary depending on the modulation technique utilized to control the drone, for example:

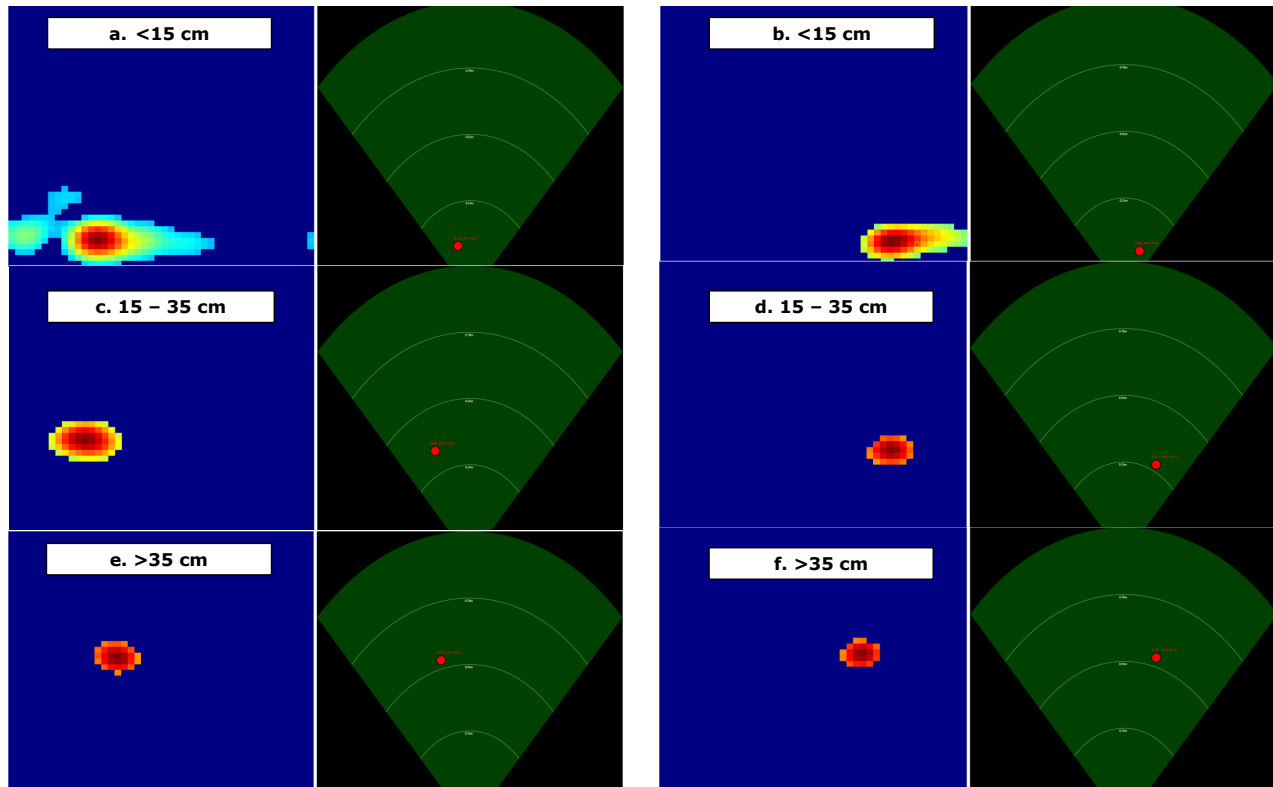


Fig. 3. The radar sensor captures θ - R images of various actions, including (a) take-off, (b) landing, (c) starting and (d) stopping video recording, (e) moving forward, and (f) moving backward. Each image shows foot gestures and the distance measured by the radar sensor.

- 1D, *continuous-wave (CW)* — modulation separates objects by their velocity. There is no possibility of distinguishing objects of similar velocity from objects' location. Frequently used for motion detection applications. Drones have the potential to be used in various sports, including running [30]. In addition, a 1D radar system can be used to indicate the moment when a runner crosses the finish line.
- 2D, *frequency-shift keying (FSK)* — modulation separates objects by distance and velocity. Location of objects in a one-dimensional environment without information about the angle. The modulation being FSK, it only separates objects based on speed, and it offers the advantage of measuring the distance. Drones, for example, are found on construction sites before or during the execution of construction work, working alongside human workers [31]. The 2D radar will be utilized to gauge the distance between objects.
- 3D, *frequency-modulated continuous wave (FMCW)* — multiple input/multiple output (MIMO) modulation separates objects by velocity, distance and angle. Objects of the same speed, distance, and angular position can be detected and their 2D location determined. Multiple transmission and reception antennas increase the resolution of the sensor. Drones have various applications in emergency situations, such as search and rescue operations [32]. In the event of an avalanche, a 3D radar sensor can be used to determine the distance between the rescuers and the individuals in need of assistance.
- 4D, *FMCW MIMO* — modulation separates objects by velocity, distance and angle (horizontal and vertical). Objects with the same speed, distance, and angular distance can be detected. Objects can be located in 3D. Compared to a 3D radar, 4D radars use more antennas and can detect the angle between horizontal and vertical. This brings about the advantage of locating objects in a 3D environment. Drones or clusters of drones are commonly used for aerial shows [33], as well as for image projection [34] and aiding in projections [35]. The 4D radar sensor can be used to calculate the speed of the aerial displays to coordinate their positions.

We did not use the 1D-RADAR and 2D-RADAR categories because we want to cover as many functionalities SkySculptor as possible and, in certain situations, distinguish gestures from different objects in the room. Depending on the resolution, we can choose different commercial radar sensors, also putting emphasis on the operating distance. For example, in the case of integrating the radar sensor into the floor or wall, we are going to need a much smaller distance, compared to the situation where the sensor is integrated ceiling. The initial form of the algorithm scans targets to flip pages of an opened document; If the y coordinates of the identified targets were decreasing, the up button was pressed to access the previous page, and if the targets were increasing, the down button was pressed to access the next page. We modified this algorithm at both the recognition level and the semantic level of gestures. We wanted to better recognize the original form, so we calculated the sum of the evaluated y coordinates. Now, if the values are

decreasing and the sum is negative, we stand for a *swipe-left*; otherwise, if the values are increasing and the sum is positive, a *swipe-right* is identified. For each identified gesture, we send a specific command via WebSocket to the Drone running on the PC in our situation, such as: forward [36], [37], backward [38], [39], take a photo [11], etc.

Radar-sensing gestures imply that a gesture is made in air, without touch. To this end, another feature that is interesting for a gesture is the distance of foot from the Walabot. In this way, we modified the original algorithm by also capturing the z coordinate. Because a gesture is made up of a set of points, we calculate the average distance between the foot and Walabot. We obtain the result that a gesture can be performed near *near* Walabot, at a reasonable distance from it *close* or very far from Walabot *far*. In summary, we identified *two* swipe gestures \times 3 possible positions of each gesture, that is, a set of *six* gestures.

V. RESULTS

The creation and implementation of SkySculptor, a specialized software designed to improve drone interactions using foot gestures detected by radar, have produced positive results. The software was developed with specific design requirements to ensure smooth compatibility with open-source technologies. Integrating radar sensors into the application's framework enabled drone control by recognizing foot movements detected by the Walabot radar device. A set of six unique directional swipe gestures was introduced, each linked to predefined drone functions, allowing users to easily perform actions like drone take-off/landing, starting/stopping video recording, and changing directions. During the prototype phase, the use of a 15-antenna ultra-wideband 3D radar in combination with the Parrot Mambo drone was used to validate the effectiveness of the gesture recognition system. The ability of the radar sensor to capture detailed foot trajectories at various distances and angles provided reliable data for accurate gesture recognition. Detailed radar heat maps showed the gestures and spatial positions of the radar sensor. Overall, the successful implementation of SkySculptor demonstrates its potential to extend the human-drone interaction in smart indoor environments, paving the way for future advancements in this evolving field.

VI. DISCUSSIONS

The introduction of SkySculptor, a software application aimed at simplifying drone control through radar-based foot gestures in intelligent indoor spaces, involves a series of discussions. The central theme of this scientific article revolves around the effective merging of open-source technologies with the intricacies of smart indoor environments to ensure a variety of user interaction methods and technological functionalities. Using radar sensors, SkySculptor uses conventional drone control techniques, offering users an intuitive interface based on foot gestures, which improves the accessibility of indoor drone manipulation. Moreover, the successful deployment and verification of SkySculptor underscore the potential to extend radar-based gesture recognition systems to transform the dynamics of human-drone interactions. These outcomes collectively pave the way for a wider acceptance of radar-based foot gesture control systems.

VII. LIMITATIONS AND FUTURE WORK

Our implementation uses Python-based APIs to control drones and radar devices. While Python's versatility and widespread support are commendable, certain specialized or restricted environments may encounter difficulties in fully accommodating Python. This is particularly true for wearable devices or embedded systems, which may have limited resources or be optimized for alternative programming languages. Another limitation is associated with the data transmission capacity of the radar device. Enhancing this capability could enable for a more extensive implementation of a wider range of gestures for drone control. Currently, the SkySculptor software tool does not have a predefined set of user-customized gestures. Future development considerations include improving the gesture dictionary through code generation and incorporating a radar device capable of processing larger volumes of data. Lastly, there is a limitation in terms of drone compatibility, which is currently limited to Parrot drones. Adapting the software implementation to be compatible with other drone brands presents a challenge. In addition, limitations extend to factors such as drone size, flight time, and sensor configurations. These challenges present opportunities for further development within the SkySculptor software system. Furthermore, different types of radar will produce different types of data. Therefore, it is useful to investigate other characteristics of radar sensors, such as resolution or field of view, in future work to implement various designs of gestures. We leave the examination of diverse radar technologies for future research.

VIII. CONCLUSION

The software application, SkySculptor, presents a novel approach to controlling drones in smart indoor environments by using gestures of the foot on radar. This addresses a gap in the current literature regarding the use of foot gestures for drone control. The article provides an overview of the context and scope of SkySculptor, emphasizing the fundamental principles of radar sensing and the potential of this technology for controlling drones in intelligent indoor environments. The related work highlights the limited information available on foot gestures for drone control despite previous research efforts. Additionally, the paper presents a detailed workflow for SkySculptor, which includes design requirements, an implementation approach, and a prototype of the proposed system.

ACKNOWLEDGMENT

The reported research was conducted in the Machine Intelligence and Information Visualization Research Laboratory of the Ștefan cel Mare University of Suceava.

DECLARATIONS

The authors declare no conflict of interest. Data sharing is not applicable to this article, as no data sets were generated or analyzed during the current scientific paper.

REFERENCES

- [1] A.-I. Șiean, C. Pamparau, A. Sluțters, R.-D. Vatavu, and J. Vanderdonck, "Flexible gesture input with radars: systematic literature review and taxonomy of radar sensing integration in ambient intelligence environments," *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–15, 2023.

- [2] A.-I. Şiean, C. Pamparău, and R.-D. Vatavu, "Scenario-based exploration of integrating radar sensing into everyday objects for free-hand television control," in *ACM International Conference on Interactive Media Experiences*, ser. IMX '22. New York, NY, USA: Association for Computing Machinery, 2022, p. 357–362. [Online]. Available: <https://doi.org/10.1145/3505284.3532982>
- [3] B.-F. Gheran, J. Vanderdonckt, and R.-D. Vatavu, "Gestures for smart rings: Empirical results, insights, and design implications," in *Proc. of Designing Interactive Systems Conference*. ACM, 2018, pp. 623–635. [Online]. Available: <https://doi.org/10.1145/3196709.3196741>
- [4] A.-I. Şiean, B. Gradinaru, O. Gherman, M. Danubianu, and L. D. Milici, "Opportunities and challenges in human-swarm interaction: Systematic review and research implications," *International Journal of Advanced Computer Science and Applications*, 2023. [Online]. Available: <https://api.semanticscholar.org/CorpusID:258476779>
- [5] C. Ardito, P. Buono, M. F. Costabile, and G. Desolda, "Interaction with large displays: A survey," *ACM Comput. Surv.*, vol. 47, no. 3, feb 2015.
- [6] E. Velloso, D. Schmidt, J. Alexander, H. Gellersen, and A. Bulling, "The feet in human-computer interaction: A survey of foot-based interaction," *ACM Comput. Surv.*, vol. 48, no. 2, sep 2015.
- [7] R.-D. Vatavu, "From natural to non-natural interaction: Embracing interaction design beyond the accepted convention of natural," in *Proceedings of the 25th International Conference on Multimodal Interaction*, ser. ICMI '23. New York, NY, USA: Association for Computing Machinery, 2023, p. 684–688. [Online]. Available: <https://doi.org/10.1145/3577190.3616122>
- [8] N. Magrofuoco, P. Roselli, and J. Vanderdonckt, "Two-dimensional stroke gesture recognition: A survey," *ACM Comput. Surv.*, vol. 54, no. 7, jul 2021.
- [9] A.-T. Andrei, A.-I. Şiean, and R.-D. Vatavu, "Tap4light: Smart lighting interactions by tapping with a five-finger augmentation device," in *13th Augmented Human International Conference*, ser. AH2022. New York, NY, USA: Association for Computing Machinery, 2022. [Online]. Available: <https://doi.org/10.1145/3532525.3532535>
- [10] A.-I. Şiean, "A set of smart ring gestures for drone control," *Proceedings of the 12th International Conference on "Electronics, Communications and Computing"*, 2022. [Online]. Available: <https://api.semanticscholar.org/CorpusID:258054122>
- [11] A.-I. Şiean, R.-D. Vatavu, and J. Vanderdonckt, "Taking that perfect aerial photo: A synopsis of interactions for drone-based aerial photography and video," in *ACM International Conference on Interactive Media Experiences*, ser. IMX '21. New York, NY, USA: Association for Computing Machinery, 2021, p. 275–279. [Online]. Available: <https://doi.org/10.1145/3452918.3465484>
- [12] M. Cirelli and R. Nakamura, "A survey on multi-touch gesture recognition and multi-touch frameworks," in *Proc. ITS '14*. ACM, 2014, p. 35–44.
- [13] S. Villarreal-Narvaez, A.-I. Şiean, A. Sluÿters, R.-D. Vatavu, and J. Vanderdonckt, "Informing future gesture elicitation studies for interactive applications that use radar sensing," *Proceedings of the 2022 International Conference on Advanced Visual Interfaces*, 2022. [Online]. Available: <https://doi.org/10.1145/3531073.3534475>
- [14] A. Gigie, S. Rani, A. Chowdhury, T. Chakravarty, and A. Pal, "An agile approach for human gesture detection using synthetic radar data," in *Adjunct Proceedings of the 2019 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2019 ACM International Symposium on Wearable Computers*, ser. UbiComp/ISWC '19 Adjunct. New York, NY, USA: Association for Computing Machinery, 2019, p. 558–564. [Online]. Available: <https://doi.org/10.1145/3341162.3349332>
- [15] A. Patra, P. Geuer, A. Munari, and P. Mähönen, "Mm-Wave Radar Based Gesture Recognition: Development and Evaluation of a Low-Power, Low-Complexity System," in *Proc. mmNets '18*. ACM, 2018, p. 51–56.
- [16] F. C. Y. Li, D. Dearman, and K. N. Truong, "Virtual shelves: Interactions with orientation aware devices," in *22nd Annual ACM Symp. on User Interface Software and Technology*, 2009, pp. 125–128. [Online]. Available: <https://doi.org/10.1145/1622176.1622200>
- [17] M. Obaid, F. Kistler, G. Kasparavičiūtė, A. E. Yantaç, and M. Fjeld, "How would you gesture navigate a drone? a user-centered approach to control a drone," in *Proceedings of the 20th International Academic Mindtrek Conference*, ser. AcademicMindtrek '16. New York, NY, USA: Association for Computing Machinery, 2016, p. 113–121. [Online]. Available: <https://doi.org/10.1145/2994310.2994348>
- [18] M. Monajjemi, S. Mohaimeniyanpour, and R. Vaughan, "Uav, come to me: End-to-end, multi-scale situated hri with an uninstrumented human and a distant uav," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016, pp. 4410–4417.
- [19] D. Avola, M. Bernardi, L. Cinque, G. L. Foresti, and C. Massaroni, "Exploiting recurrent neural networks and leap motion controller for the recognition of sign language and semaphoric hand gestures," *IEEE Transactions on Multimedia*, vol. 21, no. 1, pp. 234–245, 2019.
- [20] J. Nagi, A. Giusti, G. A. Di Caro, and L. M. Gambardella, "Human control of uavs using face pose estimates and hand gestures," ser. HRI '14. New York, NY, USA: Association for Computing Machinery, 2014, p. 252–253. [Online]. Available: <https://doi.org/10.1145/2559636.2559833>
- [21] K. Nitta, K. Higuchi, and J. Rekimoto, "Hoverball: Augmented sports with a flying ball," in *Proceedings of the 5th Augmented Human International Conference*, ser. AH '14. New York, NY, USA: Association for Computing Machinery, 2014. [Online]. Available: <https://doi.org/10.1145/2582051.2582064>
- [22] K. Yamaguchi, G. Kato, Y. Kuroda, K. Kiyokawa, and H. Takemura, "A non-grounded and encountered-type haptic display using a drone," ser. SUI '16. New York, NY, USA: Association for Computing Machinery, 2016, p. 43–46. [Online]. Available: <https://doi.org/10.1145/2983310.2985746>
- [23] S. Palipana, D. Salami, L. A. Leiva, and S. Sigg, "Pantomime: Mid-air gesture recognition with sparse millimeter-wave radar point clouds," *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, vol. 5, no. 1, mar 2021. [Online]. Available: <https://doi.org/10.1145/3448110>
- [24] D. Avrahami, M. Patel, Y. Yamaura, and S. Kratz, "Below the surface: Unobtrusive activity recognition for work surfaces using rf-radar sensing," in *23rd International Conference on Intelligent User Interfaces*, ser. IUI '18. New York, NY, USA: Association for Computing Machinery, 2018, p. 439–451. [Online]. Available: <https://doi.org/10.1145/3172944.3172962>
- [25] H.-S. Yeo and A. Quigley, "Radar sensing in human-computer interaction," *Interactions*, vol. 25, no. 1, p. 70–73, dec 2017. [Online]. Available: <https://doi.org/10.1145/3159651>
- [26] L. A. Leiva, M. Kljun, C. Sandor, and K. Copic Pucihar, "The wearable radar: Sensing gestures through fabrics," in *22nd International Conference on Human-Computer Interaction with Mobile Devices and Services*, ser. MobileHCI '20. New York, NY, USA: Association for Computing Machinery, 2021. [Online]. Available: <https://doi.org/10.1145/3406324.3410720>
- [27] E. Hayashi, J. Lien, N. Gillian, L. Giusti, D. Weber, J. Yamanaka, L. Bedal, and I. Poupyrev, "Radarnet: Efficient gesture recognition technique utilizing a miniature radar sensor," in *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, ser. CHI '21. New York, NY, USA: ACM, 2021. [Online]. Available: <https://doi.org/10.1145/3411764.3445367>
- [28] H. Liu, Y. Wang, A. Zhou, H. He, W. Wang, K. Wang, P. Pan, Y. Lu, L. Liu, and H. Ma, "Real-time arm gesture recognition in smart home scenarios via millimeter wave sensing," *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, vol. 4, no. 4, dec 2020. [Online]. Available: <https://doi.org/10.1145/3432235>
- [29] A. Sluÿters, S. Lambot, and J. Vanderdonckt, "Hand gesture recognition for an off-the-shelf radar by electromagnetic modeling and inversion," in *Proceedings of the 27th International Conference on Intelligent User Interfaces*, ser. IUI '22. New York, NY, USA: ACM, 2022, p. 506–522. [Online]. Available: <https://doi.org/10.1145/3490099.3511107>
- [30] M. Seuter, E. R. Macrillante, G. Bauer, and C. Kray, "Running with drones: desired services and control gestures," in *Proceedings of the 30th Australian Conference on Computer-Human Interaction*, ser. OzCHI '18. New York, NY, USA: Association for Computing Machinery, 2018, p. 384–395. [Online]. Available: <https://doi.org/10.1145/3292147.3292156>
- [31] A. Freistetter, M. Pollak, and K. A. Hummel, "Performance of a networked human-drone team: command response and interaction effects," in *Proceedings of the 6th ACM Workshop on Micro Aerial Vehicle Networks, Systems, and Applications*, ser. DroNet '20. New

- York, NY, USA: Association for Computing Machinery, 2020. [Online]. Available: <https://doi.org/10.1145/3396864.3399703>
- [32] J. Cacace, A. Finzi, V. Lippiello, M. Furci, N. Mimmo, and L. Marconi, "A control architecture for multiple drones operated via multimodal interaction in search & rescue mission," in *2016 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, 2016, pp. 233–239.
- [33] W. Yamada, K. Yamada, H. Manabe, and D. Ikeda, "isphere: Self-luminous spherical drone display," in *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*, ser. UIST '17. New York, NY, USA: Association for Computing Machinery, 2017, p. 635–643. [Online]. Available: <https://doi.org/10.1145/3126594.3126631>
- [34] R. Lingamaneni, T. Kubitzka, and J. Scheible, "Dronecast: towards a programming toolkit for airborne multimedia display applications," in *Proceedings of the 19th International Conference on Human-Computer Interaction with Mobile Devices and Services*, ser. MobileHCI '17. New York, NY, USA: Association for Computing Machinery, 2017. [Online]. Available: <https://doi.org/10.1145/3098279.3122128>
- [35] R. Darbar, J. S. Roo, T. Lainé, and M. Hachet, "Dronesar: extending physical spaces in spatial augmented reality using projection on a drone," in *Proceedings of the 18th International Conference on Mobile and Ubiquitous Multimedia*, ser. MUM '19. New York, NY, USA: Association for Computing Machinery, 2019. [Online]. Available: <https://doi.org/10.1145/3365610.3365631>
- [36] Y. Yu, X. Wang, Z. Zhong, and Y. Zhang, "Ros-based uav control using hand gesture recognition," in *2017 29th Chinese Control And Decision Conference (CCDC)*, 2017, pp. 6795–6799.
- [37] E. Peshkova, M. Hitz, and B. Kaufmann, "Natural interaction techniques for an unmanned aerial vehicle system," *IEEE Pervasive Computing*, vol. 16, no. 1, pp. 34–42, 2017.
- [38] V. L. Popov, K. B. Shiev, A. V. Topalov, N. G. Shakev, and S. A. Ahmed, "Control of the flight of a small quadrotor using gestural interface," in *2016 IEEE 8th International Conference on Intelligent Systems (IS)*, 2016, pp. 622–628.
- [39] S.-Y. Shin, Y.-W. Kang, and Y.-G. Kim, "Hand gesture-based wearable human-drone interface for intuitive movement control," in *2019 IEEE International Conference on Consumer Electronics (ICCE)*, 2019, pp. 1–6.