EPiC Series in Computing
Volume 82, 2022, Pages 53–62
Proceedings of 37th International Conference on Computers and Their Applications

Designing Application Appropriateness for Augmented Reality with Requirement Integration from the User-Cursor Paradigm

Dr. Bradford A. Towle Jr. and Gabrielle Tristani
Florida Polytechnic University, Florida, US
btowle@floridapolytechnic.edu, gtristani8152@floridapoly.edu

Abstract

Technological advances have increased the potential for wearable augmented reality (AR) devices. More applications are being ported for this platform with this development, and not all are suited for AR. After the initial novelty fades, users often find the application gimmicky and unwieldy, causing them to become incredulous towards AR's potential. Developers need to ensure that the application emphasizes elements that are unique to AR for a successful port. This paper analyzes the human-cursor paradigm to derive a set of requirements and integrate them into an application. With this integration, an application that has no exclusive need of the AR platform can be redesigned to take advantage of AR's capability making it more valuable to the user. This emphasis will promote targeted development of meaningful applications further expanding the augmented reality platform.

1. Introduction

Technology advancement is pushing the feasibility of wearable augmented reality (AR) devices ever closer to realization. Augmented reality is defined as overlaying virtual information onto the real world from a particular perspective [1], [2]. If the virtual perspective coincided with the user's perspective, the experience would be more immersive and dynamic. As a result, head-mounted devices (HMDs) are now considered a viable product.

It is important to note that augmented reality (AR) is not virtual reality (VR). Virtual reality entails the user is wholly immersed in a virtual environment replacing, instead of adding, to the real environment. The two technologies rely on common fields such as 3D graphics, HMDs, and user movement tracking within a virtual space [3]. However, beyond these common dependencies, the two applications serve significantly different purposes. It is important to mention that both AR and VR are often viewed as the same platform. This view is particularly derogatory to AR when the platform's relevance is questioned as it cannot fulfill the same roles as VR.

Augmented reality is often referred to as a solution in search of a problem [4]. This phenomenon is especially noticeable when an application is ported to an AR platform. At first, users are interested, and the experience is unique and exciting. However, after the initial impact, the user loses interest since the application is better suited on a traditional platform such as a pc, laptop, or smartphone. This loss of interest promotes the view that AR is not viable and causes a new problem for porting applications to AR. Traditionally, the most significant problem with porting applications to new devices was addressing different input modalities of the device. Even though AR has its share of input problems, a plethora of technologies exist to overcome input modality limitations [3], [5]–[7]. The fundamental problem to address is whether an application is appropriate for AR or can be redesigned to be applicable to AR.

The proposed solution in this paper distills core requirements that promote unique features of the AR platform and then re-integrates them into an application for porting. Features unique to AR involve moving, interacting and viewing in both virtual and real environments. This research builds upon a previous project referred to as the human-cursor paradigm. The human-cursor paradigm promotes the view that the user is an input device in an AR application, meaning movement, selection, and collision all should influence the application. When designing the human-cursor paradigm, the requirements were mapped to positive HCI metrics to demonstrate its merit. The literature review for this research revealed that the human-cursor paradigm requirements also promoted best-practice for AR applications. Therefore, these elements were used in creating the core requirements to promote the unique features of AR.

Once these core requirements were derived, the design of a proof-of-concept application began. This application was a puzzle game that initially did not have specific requirements for AR. The core requirements were then integrated with the game mechanic causing significant changes to its design. The design was then implemented and tested on the HoloLens 1.

2. Related Work

Augmented reality has struggled to find a clearly defined purpose when it comes to applications. Many different proofs-of-concept exist, but most are nothing more than a technology demo. One field that AR has been effectively applied is in healthcare. The healthcare industry has employed AR for three main categories: teaching, therapy/exercise, and visualization. AR's visual overlay of information can provide health students with additional information while learning a subject. This additional information coupled with modern simulation is becoming a cost-effective teaching tool [8]–[10]. Therapy and exercise often require individualized coaching or feedback to the patient, which can be accomplished by augmented reality's visual projections of different data providing appropriate cues to the patient [11]–[14]. Finally, the amount of information in the healthcare industry is immense. Whether it is a heads-up display for a surgeon, nutritional information at a grocery store, or patient information, augmented reality can be used to project the information in a more helpful manner [8], [15], [16].

Another field that intuitively seems compatible with AR is gaming. Surprisingly, gaming has not championed AR as much as its counterpart, virtual reality. There have been some successes [1]; however, most AR games do not move past the demo stage [17]. While AR games are slowly gaining traction, they have not significantly impacted the multibillion-dollar gaming industry.

This lack of impact raises the question of why some AR applications work and others do not. The list on the left focuses on generalized AR applications, and the list on the right addresses game-specific elements.

Necessary Elements from Applying HCI Principles to AR Systems Design [4]

1. Affordance – virtual objects behave like their physical equivalent
2. Reducing cognitive overhead – does not require a lot of thought to use
3. Low physical effort
4. Learnability
5. User satisfaction
6. Flexible in use
7. Responsiveness and feedback
8. Error tolerance

List 1: Necessary Elements from Applying HCI Principles to AR Systems Design

Necessary Elements from Guidelines for Designing Augmented Reality Games [18]

1. Stick to a theme
2. Do not stay digital (make use of real-world props)
3. Use the real environment
4. Keep it simple
5. Create sharable experience
6. Use various social elements
7. Show reality (Do not let virtual world obscure real world)
8. Turn weaknesses into strength (Use technical problems within the gaming experience)
9. Do not just convert (Do not simply convert a game to AR)
10. Create meaningful content
11. Choose your tracking wisely

List 2: Necessary Elements from Guidelines for Designing Augmented Reality Games

At first, these lists do not seem related. List 1 is a traditional software engineering view of HCI applied toward AR devices. It is generalized enough to be applied to almost any HCI field. List 2 provides advice specifically on AR game design. However, deeper analysis will help us answer why some AR applications fail and others do not. When writing the human-cursor paradigm, a requirement analysis compared the necessary features to the elements on List 1. The diagram below displays the number of requirements that directly support each element. The entire human-cursor paradigm was designed to be generalized and flexible in use across applications and platforms. Therefore, this element was not included since almost all the requirements would support it. The last two elements were tied directly to the hardware, so they were omitted.

The note-worthy element in this graphic (Figure 1) is the lack of requirements supporting the low physical effort. The human-cursor paradigm was designed to be user-centric, and many of the required features were used to make the experience more enjoyable and convenient. However, head-mounted displays employing AR applications require the user to be more physically active. The lack of reduction in physical effort is the first clue to why many AR applications fail.

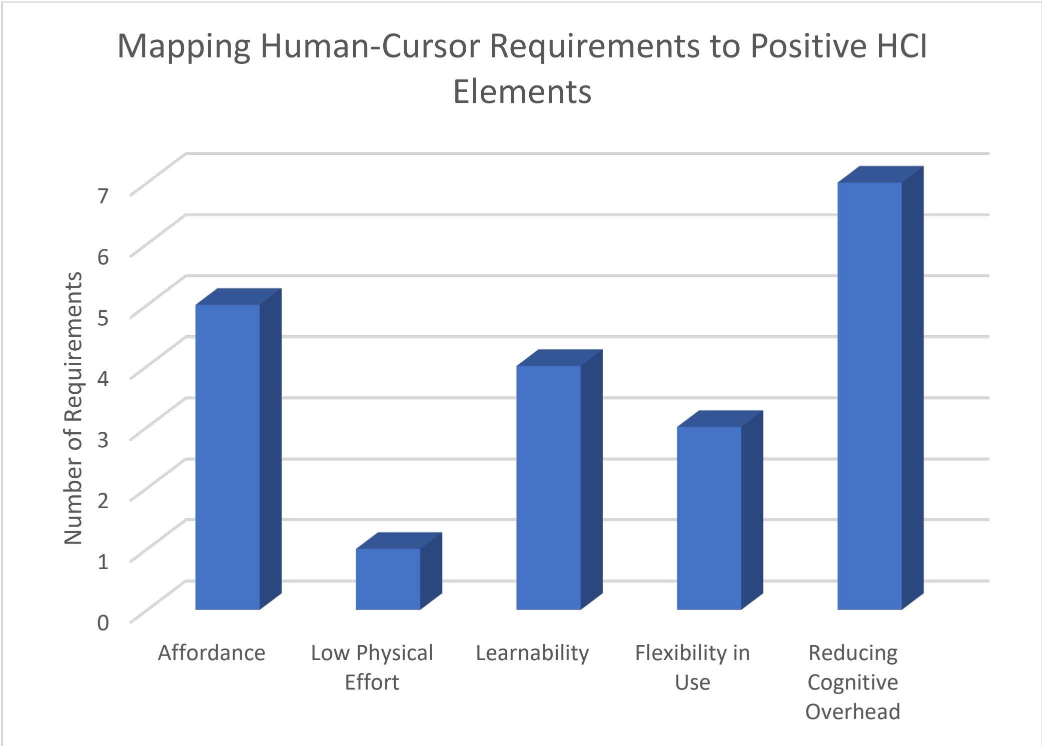

Figure 1: Mapping Human-Cursor Requirements to Positive HCI Elements

Any input device that requires significant physical effort will eventually be discarded even if it has an amazing user experience. This effect is accelerated should the user interact with the platform over a long period. Thus, users will be impressed by the graphics and presentation of an AR application, but as they use the application, they realize it will be easier to use with a keyboard/mouse or even a touch screen. This realization causes the application to be used less, and people return to using a platform that requires less physical effort.

One could incorrectly conclude that the AR platform was not worth continued investigation due to the higher degree of physical effort. However, there are two situations where AR has shown to be useful. First, AR applications are practical when data is required to be displayed to the user on top of reality as demonstrated by the health care applications. Secondly, AR is also effective when the user's movement within reality has a meaningful impact on the application validated by List 2. Within List 2 four of the eleven elements recommend that the programmer integrate reality into the application and not specifically with game design.

2. Do not stay digital (make use of real-world props)

3. Use the real environment

7. Show Reality (Do not let virtual world obscure real world)

9. Do not just convert (Do not simply convert a game to AR)

In summary, health care applications have demonstrated successful deployment on AR while games have not been as successful. After taking analyzing requirements of the human-cursor paradigm it was noted that a head-mounted display requires greater physical effort then a computer or smart phone. Therefore, it was concluded that users will prefer platforms with lower physical effort unless the application itself involves moving, interacting, and viewing in both a virtual ***and*** real environment. Notice the 'and' not 'or'. In other words, users will only prefer to use the AR platform when the application promotes and emphasizes its unique features.

3. Methodology

This paper explored a novel method to assure application appropriateness for the AR platform by integrating a set of distilled vital requirements from the human-cursor paradigm into a new program. Instead of using the human-cursor paradigm code directly, the new program had to be significantly altered to align with the requirements. This integration was done to ensure that AR was needed. If requirements cannot be incorporated seamlessly, there is a strong indication that the program will fail on an AR platform. To clarify, the initial concept of a program may not require any AR features; however, by forcing the adoption of AR requirements, a program will be altered to integrate more efficiently with AR.

To understand the specific requirements used in the above integration, a deeper explanation of the human-cursor paradigm is necessary. The paradigm centers on the idea that the user is an input device and tracks the user's movement. This movement input can be subscribed to by an event system similar to a mouse API. When the user moves, the program is notified of the new coordinates. In fact, throughout the entire human-cursor paradigm, different events were added to provide developers with abundant opportunities to provide visual and audio cues. Another objective of this paradigm was to be as general and platform-independent as possible; therefore, the paradigm will allow users to select objects directly in front of the user, avoiding any complex hand or eye-tracking. The range in which a user may select an object is also configurable. The paradigm also addresses how user collision will be handled. There were initially three different modes: No Restraint, No Collision, and Valid Path. Each option places different constraints on the user should they collide with an invalid virtual obstacle.

- No Restraint – Collision with a virtual obstacle will not disable the user's input ability or stop reporting their movement.
- No collision – The user's movement and selection will not be acknowledged until the user is no longer colliding with an invalid virtual object.
- Valid Path – The user's position and selection will not reenable until the user has returned to the point in which they initially collided with an invalid virtual obstacle.

One assumption the paradigm makes is that the AR device can be spatially aware. Spatial awareness means the device maps the surroundings and tracks the user's movement through the real environment. This functionality is crucial if the AR application is going to use objects in the real world and the virtual world. Not all AR devices are capable of spatial awareness, although it is becoming far more widespread.

As shown above in Figure 1, the formal requirements of the human-cursor paradigm were mapped to positive attributes for HCI. The fact that the paradigm promotes elements from List 2 also served as validation. There were twenty requirements for the initial human-cursor paradigm; however, they can be summed up with the following four over-arching statements:

- The user's movement and position should be an input.
- The user should be able to pick (select) a virtual object.

- There should be some affect when the user collides with virtual objects, including consequences for invalid obstacles.
- The real world should have direct impact on the AR application.

These four requirements were integrated meaningfully into the initial requirements of a game, which caused significant change to the original concept. It should also be noted, that displaying information related to specific real-world positions also emphasizes AR features. However, this research is focused more on the user input then visualization. The initial game concept was simple. A player would have to move around and collide with a virtual character to control them. Each character had a different skill, strength, and weakness. The player would also have to collect keys to open the chest by selecting the key within range. The overall goal was to maneuver the virtual character to a specific point in the level to proceed to the next. The players would lose control of the virtual character if they bumped into a wall or any hazards destroyed it. Notice nothing about this game implies the need for AR to function. This game could be designed for PC, consoles, and even smartphones.

The above four requirements were integrated into the game in the following manner. The user would act as the main character. The main character would move and rotate in the same manner as the user. The user could also interact with objects, such as switches and buttons, by colliding with them providing meaningful interactions with valid game objects. The 3D picking provided by head tracking was used when selecting keys and chests; however, this could only be done if a virtual character was being controlled. This constraint meant that the user could not collide with any walls and was enforced by modifying the Valid Path mode. When the player collided with a wall, they would lose the ability to provide input or control a virtual character. However, instead of returning to the point where they collided with the wall, they would have to return to the virtual character to regain the ability to provide input. This modification of the human-cursor paradigm demonstrates its flexibility for different applications. The last requirement mandated that real-world objects would act as obstacles. This requirement meant a real-world object could be used to block an enemy or a trap from colliding with the virtual character.

Once these requirements were integrated, the design was re-evaluated. It was determined that the project was still rational, and the modifications were not so obtuse that the gameplay would be awkward. This step is critical in determining the appropriateness of the application. If the program is unwieldy or non-sensical after the AR requirements are integrated, then the developers should know that the port will most likely fail.

4. Results and Analysis

After the design of the game was complete, it was implemented on the Microsoft HoloLens 1. The player object would follow the camera, which was controlled by the user's position and movement. The HoloLens localized itself with its surroundings to ensure the user's position is as accurate as possible. The player object could interact directly with two other objects: the red panel (Path A), and the switch (figure 2).

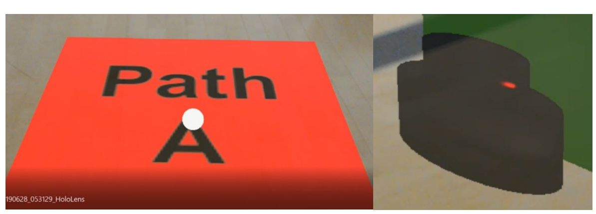

Figure 2: Red Switch (left), Toggle Switch (right)

The player could control three virtual characters: Shield Turtle, Battle Tank, and Scout Wolf (Figure 3). The Shield Turtle would not take damage if it remained still, the Battle Tank could shoot a bullet if the user made the select gesture, and the Scout Wolf was faster than the other two making certain puzzles easier.

Figure 3: Shield Turtle (left), Battle Tank (center), and Scout Wolf (right)

The user would have to collide with one of the above characters and navigate them to the blue panel (Path B) to progress to the next level. If the user collided with a wall, they would lose control of the character and must return to it to regain control. The collision would also deactivate their ability to provide input (Figure 4).

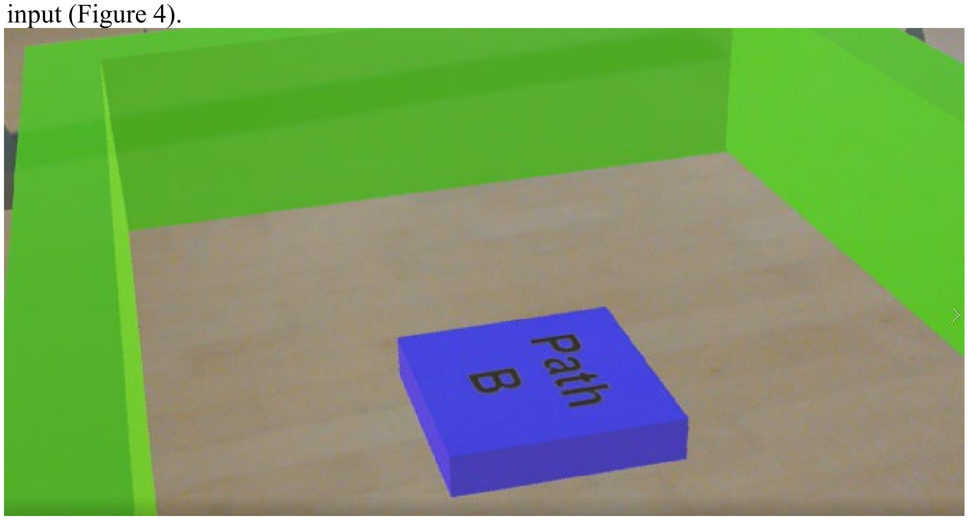

Figure 4: Blue Panel Surrounded by a Wall

The user could select objects by placing the gaze cursor (small white ball) on the object and making the select gesture for the HoloLens 1. If the object was in range and the user had control of a virtual character, they would select it. Please note that the HoloLens screen capture painted the virtual elements over the user's finger (Figure 5).

There were three different kinds of traps that could hurt the virtual characters (Figure 6). The turret would shoot bullets periodically. On one level, the user had to shield the virtual character using a real-world object. Spikes provided a static obstacle that would damage the virtual characters. There were also patrolling robot drones. Often, the user would have to destroy these drones with the Battle Tank.

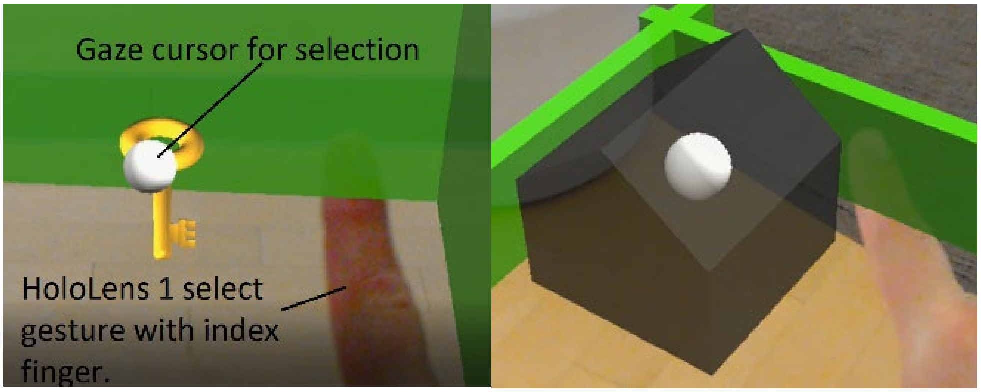

Figure 5: User Selecting a Key (left), User Selecting a Chest (right)

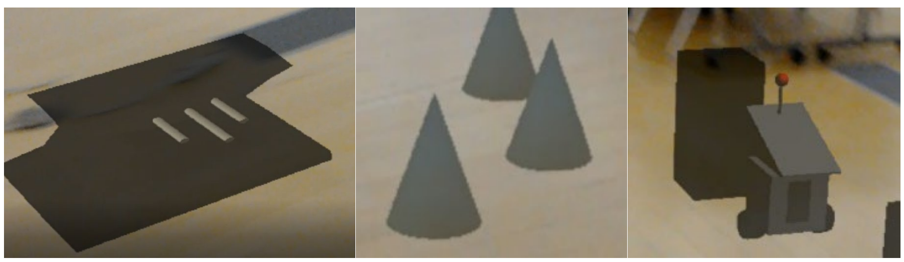

Figure 6: Turret (left), Spikes (center), and Patrol Robot (right)

After the AR application was initially programmed, it tested heuristically to determine if it was appropriate for the platform. The game itself integrated well into the physical environment, and integrating augmented reality did not seem forced. Only two problems were noted that fell under technical issues. The first problem was that the HoloLens 1 has a very narrow field of view making navigating the maze challenging since the user would often have to look down to determine if they were going to collide with a wall. This technical issue increased the challenge of the game but did not detract from the gameplay. The second problem observed was low framerate. After some analysis, it was determined that even though the game logic was rudimentary, the HoloLens could not keep up with it every frame. Current work is underway to redesign the human-cursor paradigm so it updates slower than the frames per second. Augmented reality, like virtual reality, requires a faster framerate to ensure the user does not become disoriented or nauseous. The application proved to be appropriate for the AR platform and was only limited by technical issues of the HoloLens.

5. Future Work and Conclusion

Several items require further exploration. The human-cursor interface needs to be redesigned, taking into account the limited CPU resource for HMDs. This work is already in progress and will improve the performance significantly. A user study needs to be performed on this application to evaluate whether players would lose interest. After completing these two elements, the final step is to quantify the appropriateness with a generalized equation. If such an equation could be derived, it would provide a quantitative confidence value of a program being successful on the AR platform.

In conclusion, this paper discussed the phenomena affecting many ports to the AR platform where interest quickly waned. A brief overview of successful applications was provided, comparing positive traits from two different papers. This comparison deduced that a successful application port requires users to view the AR platform contribution as an added value. Otherwise, the user will choose a conventional platform with lower physical effort. This paper then demonstrated integrating the following four requirements into an application making the application better suited for the AR platform:

- The user's movement and position should be an input and affect the application more than simply moving the camera.
- The user should be able to pick (select) a virtual object.
- There should be some affect when the user collides with virtual objects, including consequences for invalid obstacles.
- The real world should have a direct impact on the AR application.

For a proof-of-concept, a game mechanic that did not directly require AR was derived. Then, the game mechanic was altered to emphasize the unique features of augmented reality by integrating the four requirements listed above. The resulting game integrated nicely with the AR platform except for two technical issues noted in the results. This experiment is the first of several to help developers design and create meaningful AR applications, which will improve the view of the general public and take strides to employ this powerful new platform more effectively.

References

[1] P. A. Rauschnabel, A. Rossmann, and M. C. tom Dieck, "An adoption framework for mobile augmented reality games: The case of Pokémon Go," *Computers in Human Behavior*, vol. 76, pp. 276–286, 2017.

[2] A. Javornik, "Augmented reality: Research agenda for studying the impact of its media characteristics on consumer behaviour," *Journal of Retailing and Consumer Services*, vol. 30, pp. 252–261, 2016.

[3] D. A. Bowman, D. Koller, and L. F. Hodges, "Travel in immersive virtual environments: an evaluation of viewpoint motion control techniques," in *Proceedings of IEEE 1997 Annual International Symposium on Virtual Reality*, Mar. 1997, pp. 45–52. doi: 10.1109/VRAIS.1997.583043.

[4] A. Dünser, R. Grasset, H. Seichter, and M. Billinghurst, "Applying HCI principles to AR systems design," 2007, Accessed: Oct. 19, 2021. [Online]. Available: https://ir.canterbury.ac.nz/handle/10092/2340

[5] B. Nuernberger, E. Ofek, H. Benko, and A. D. Wilson, "SnapToReality: Aligning Augmented Reality to the Real World," in *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, New York, NY, USA, May 2016, pp. 1233–1244. doi: 10.1145/2858036.2858250.

[6] V. Paelke, C. Reimann, and D. Stichling, "Foot-based mobile interaction with games," in *Proceedings of the 2004 ACM SIGCHI International Conference on Advances in computer entertainment technology - ACE '04*, Singapore, 2004, pp. 321–324. doi: 10.1145/1067343.1067390.

[7] D. A. Bowman, E. Kruijff, J. J. LaViola, and I. Poupyrev, "An Introduction to 3-D User Interface Design," *Presence: Teleoperators & Virtual Environments*, vol. 10, no. 1, pp. 96–108, Feb. 2001, doi: 10.1162/105474601750182342.

[8] I. C. S. da Silva, G. Klein, and D. M. Brandão, "Segmented and Detailed Visualization of Anatomical Structures based on Augmented Reality for Health Education and Knowledge Discovery," *Adv. sci. technol. eng. syst. j.*, vol. 2, no. 3, pp. 469–478, May 2017, doi: 10.25046/aj020360.

[9] C. Moro, Z. Štromberga, A. Raikos, and A. Stirling, "The effectiveness of virtual and augmented reality in health sciences and medical anatomy," *Anatomical Sciences Education*, vol. 10, no. 6, pp. 549–559, 2017, doi: 10.1002/ase.1696.

[10] B. Garrett, J. Anthony, and C. Jackson, "Using Mobile Augmented Reality to Enhance Health Professional Practice Education," *Current Issues in Emerging eLearning*, vol. 4, no. 1, Jul. 2018, [Online]. Available: https://scholarworks.umb.edu/ciee/vol4/iss1/10

[11] J. Ahn, J. Williamson, M. Gartrell, R. Han, Q. Lv, and S. Mishra, "Supporting Healthy Grocery Shopping via Mobile Augmented Reality," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 12, no. 1s, p. 16:1-16:24, Oct. 2015, doi: 10.1145/2808207.

[12] G. Riva, R. M. Baños, C. Botella, F. Mantovani, and A. Gaggioli, "Transforming Experience: The Potential of Augmented Reality and Virtual Reality for Enhancing Personal and Clinical Change," *Frontiers in Psychiatry*, vol. 7, p. 164, 2016, doi: 10.3389/fpsyt.2016.00164.

[13] K.-F. Hsiao and H. F. Rashvand, "Data modeling mobile augmented reality: integrated mind and body rehabilitation," *Multimed Tools Appl*, vol. 74, no. 10, pp. 3543–3560, May 2015, doi: 10.1007/s11042-013-1649-8.

[14] M. Wrzesien, J.-M. Burkhardt, M. Alcañiz Raya, and C. Botella, "Mixing psychology and HCI in evaluation of augmented reality mental health technology," in *CHI '11 Extended Abstracts on Human Factors in Computing Systems*, New York, NY, USA, May 2011, pp. 2119–2124. doi: 10.1145/1979742.1979898.

[15] M. Danciu, M. Gordan, A. Vlaicu, and A. Antone, "A SURVEY OF AUGMENTED REALITY IN HEALTH CARE," vol. 52, no. 1, p. 10, 2011.

[16] M. Z. Bayu, H. Arshad, and N. M. Ali, "Nutritional Information Visualization Using Mobile Augmented Reality Technology," *Procedia Technology*, vol. 11, pp. 396–402, Jan. 2013, doi: 10.1016/j.protcy.2013.12.208.

[17] C. T. Tan and D. Soh, *Augmented Reality Games: A Review*. 2011, p. 8.

[18] R. Wetzel, R. McCall, A.-K. Braun, and W. Broll, "Guidelines for designing augmented reality games," in *Proceedings of the 2008 Conference on Future Play Research, Play, Share - Future Play '08*, Toronto, Ontario, Canada, 2008, p. 173. doi: 10.1145/1496984.1497013.