APSIPA Transactions on Signal and Information Processing, 2024, 13, e16

Industrial Technology/Advances

Using Occluded 3D Objects for Gamified Mixed Reality Captcha

Serhad Doken* and Dhananjay Lal

Adeia, 3025 Orchard Parkway, San Jose, CA 90325, USA

ABSTRACT

Spatial computing has been a popular technical topic for some time. After more than a decade of research, commercial products practicing spatial computing are now finally coming to the market. As we transition from the pure digital objects on desktop and mobile screens into blending physical objects with them, many user interface, computing and software engineering paradigms must be reimagined. As opposed to the shift from desktop to mobile, the transition from the basic 2D screen regime to 3D Digital+ Physical world needs more fundamental rethinking including "object depth" as a significant parameter. This paper focused on how CAPTCHA will need to be reimagined in the new Spatial Computing world.

Keywords: Mixed Reality, Spatial Computing, Security, Captcha.

1 Introduction

With the advent of spatial computing, digital content no longer simply lives on 2D screens such as desktops, tablets and mobile devices. The physical world also enhances the digital content on the web with cameras on devices (such as smart glasses and HMDs, in addition to mobile phones) acting as browsers to

*Corresponding author: Serhad Doken, serhad.doken@adeia.com.

Received 16 November 2023; Revised 03 February 2024
ISSN 2048-7703; DOI: 10.1561/116.20230076

create, discover and augment content, seamlessly mixing digital content with the world. Thus, the web becomes a 3D web, i.e. spatial web.

Currently, many websites require that evidence of human activity be verified in order to fill out forms or enter an authenticated area. CAPTCHA (Completely Automated Public Turing Test to tell Computers and Humans Apart) [14] has been an important tool for protecting websites against bots and automated hacking tools (Figure 1). The principle behind CAPTCHA [15] is that malicious apps are very good at completing forms automatically, but not so good at decoding the text hidden in images.

Figure 1: An example of a CAPTCHA query.

Google has developed another alternative called 'Invisible reCaptcha' (Figure 2). The system uses AI to detect how we interact with the webpage, tracking mouse movements and click times for instance. Because a bot does not make these "human" motions, the system can identify and block them.

Figure 2: An example of a reCAPTCHA query.

As web technology has evolved, creators are exploring new ways to offer information to users and solicit information from them. These interactions, often collectively referred to as the "Metaverse", model the world not as a 2D screen in which the user reads information and fills out forms using a keyboard and mouse, but rather as a 3D world in which the user interacts with objects to gather and provide information. Consider, for example, a website that seeks to deliver a geography lesson using a video to explain the formation of different types of rocks, followed by an authenticated area where the user answers some multiple choice questions. In a 3D equivalent of the website, the user would be delivered a lesson in a 3D environment where the user could, in a self-paced manner, perform interactions with the environment to understand the formation of rocks. To extend this example, the user could change the flow of water in a river to see how a gorge or valley is affected by the volume and velocity of water. To answer questions, the user may pick specific 3D objects and place them in a bin rather than pick an alternative A, B or C from a multiple-choice question.

In the Metaverse, it would still be important to verify human activity to enter authenticated 3D spaces. Thus, there will be a need to extend CAPTCHA for mixed reality. Consider, for example, a digital twin of the White House in the Metaverse that is maintained by the US government and visited by authorized visitors, whether guests or tourists. In this case, it would be important to prevent unfriendly countries' bots from entering this White House digital twin as they may attempt to create a DoS attack when inside the 3D space that prevents other visitors from entering.

2 Prior Methods of Captcha in the Spatial Domain

The problem with CAPTCHA is that it can be relatively easily defeated. Artificial intelligence and machine learning technologies allow bots to teach themselves how to analyze images and identify the letters hidden in them. They can even accurately identify elements in images, allowing them to circumvent newer CAPTCHA systems. Therefore, there is a need to design a much more solid CAPTCHA system that will take advantage of the spatial domain as well as information augmentation properties.

Sawant *et al.* [10] describes an augmented reality CAPTCHA system. Upon determining that a user, operating a user device, is attempting to access a website, a host server can cause a camera on the user device to activate and begin streaming an image feed to the host device across a network. The host device can determine an appropriate augmentation to the image feed that is germane to the context and/or environment of what is being displayed in the image feed. The augmentation can be displayed to the user on a display of the user device. The augmentation can also include a prompt instructing the user how to interact with the augmentation. The host server can determine whether to grant the user access based on the user's interaction with the augmentation. A key aspect of this idea, based on described methods, is that the augmentation is a 2D image overlay. The system verifies that the user interaction with the interactive image overlay is a correct interaction. The techniques described to determine a correct interaction include processing the augmented streaming image feed using one or more of image processing, gesture recognition, hand movement recognition, or object recognition on the augmented streaming image feed; and using the processed augmented streaming image feed to determine that a user interaction conforms to a predetermined user interaction. In some methods, verifying that the user interaction with the interactive image overlay is a correct interaction for the interactive image overlay includes receiving an indication of a gesture motion of the device; and determining that the gesture conforms to a gesture associated with the interactive image overlay determined for the environment (Figure 3).

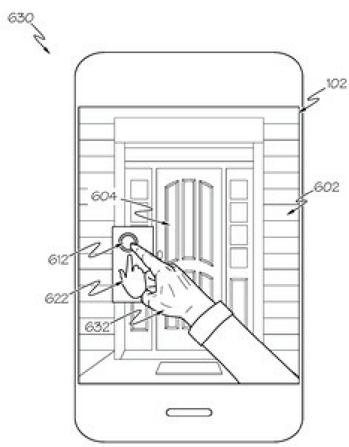

Figure 3: Gesture based Augmented Reality CAPTCHA.

[8] is another idea that describes an augmented reality CAPTCHA system. A computer device may include memory storing instructions and a processor configured to execute the instructions to provide a presentation image to a user device, wherein the presentation image is associated with a first model. The processor may be further configured to receive an image captured by a camera of the user device and a recorded alignment of the presentation image on a display of the user device; generate a second model based on the received image captured by the camera of the user device; determine an alignment of the first model with the second model based on the recorded alignment of the presentation image on the display of the user device; and generate an authentication determination verifying whether a user of the user device is

human, based on the determined alignment of the first model with the second model (Figure 4).

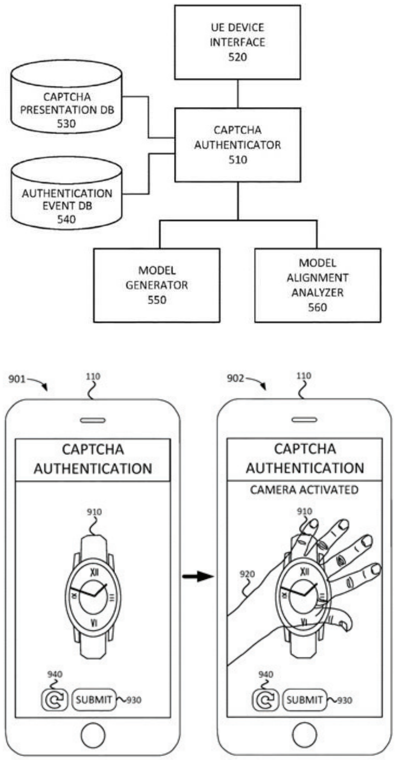

Figure 4: Image model alignment based Augmented Reality CAPTCHA.

Again, in this idea, a 2D presentation image is selected from a CAPTCHA database, and its placement and alignment in the user environment is noted (the first model). This is then compared against an actual image of the user action of alignment with the augmented asset (the second model). If the alignment passes, the host website provides access to the user.

Ganeshmani [4] describes another similar idea – an augmented reality human challenge. A human challenge can be presented in an augmented reality user interface. A user can use a smart device's camera to capture a video stream of the user's surroundings, and the smart device can superimpose a representation of an object on the image or video stream being captured by the smart device. The smart device can display in the user interface the image or video stream and the object superimposed thereon. The user will be prompted to perform a task with respect to one or more of these augmented reality objects displayed in the user interface. If the user properly performs the task, e.g., selects the correct augmented reality objects, the application will validate the user as a person.

In one example method, the augmented reality objects can be images or visual representations of objects. Each augmented reality object can have at least one attribute and the prompt can ask the user to identify at least one object with a specific attribute. For example, the interface can display augmented reality animals, and the prompt can ask the user to identify augmented reality dogs. If the user selects the augmented reality dogs displayed in the user interface, the user will be allowed to open an account on the application.

In another example method, the application can display the augmented reality objects in a 3D space. The camera in its initial position, i.e., initial field of view, can capture only part of the 3D space. Some of the augmented reality objects can appear in the initial field of view displayed in the user interface, but some of the augmented reality objects require the user to change the field of view (e.g., by rotating or tilting the smart device). Once the initial field of view is changed, the camera can capture other parts of the 3D space, and thus, the user interface can display other augmented reality objects. The user can successfully complete the prompt only if the user selects all the objects specified by the prompt, which can include objects in parts of the 3D space not displayed in the initial field of view.

Turgeman and Levin [13] describe a system that includes a spatial challenge unit to distinguish between a human user and a non-human user. The spatial challenge unit requires the user to perform one or more spatial operations that modify the spatial properties of an electronic device operated by the user. In a demonstrative method, an electronic device (e.g., smartphone, tablet, laptop, smart-watch) may comprise one or more accelerometers, one or more gyroscopes, one or more compass modules, one or more sensors, one or more modules able to determine acceleration and/or deceleration and/or orientation and/or position and/or location, one or more sensors able to determine tilt and/or angular orientation of the electronic device, one or more sensors able to determine whether the electronic device is horizontal or vertical or tilted or slanted (e.g., relative to the ground, or relative to another plane of reference), one or more sensors able to determine whether the electronic device is being rotated or is spinning or is positioned (or moving) upside-down or is positioned (or moving) sideways, one or more sensors able to determine physical pressure applied by a user onto or into or towards a particular region or component of the electronic device (e.g., a "Force Touch" touch-screen able to sense or measure the amount of force applied by the user to the touch-screen), and/or other spatial or three-dimensional properties of the electronic device and/or its position and/or its orientation and/or its movement, and/or changes or modifications in such spatial or three-dimensional properties of the electronic device.

In [9] a variation method, as described, includes receiving distinctive data relating to a user, where the distinctive data includes a video clip of the user, where the distinguishing data is captured using one or more cameras. The method may further include recognizing one or more features of the user from the video clip to determine whether the user is a human, and testing one or more characteristics of the user or the video clip to determine whether the human is a live human. The method may further include allowing access to the user to an application, if the user is determined to be the live human.

3 New Proposed Methods of 3D Captcha for the Mixed Reality Domain

In the previously described ideas, [10] and [8] rely on augmenting a scene received from a user camera using an image rather than a 3D object. Both of these ideas then rely on a specific user action to the overlaid image which is compared against a reference modeled action, to grant access. Neither of the ideas envision perspective-accurate rendering of an object, nor changing representations of the overlaid object based on the user's view. They explicitly call out just the action of augmenting the user image with an overlaid image.

In [4], we see the authors have incorporated 3Dness in an implementation of the idea, wherein the user device itself can be manipulated (ex., tilted or rotated) to complete the challenge. However, the objects overlaid are still 2D images. 3Dness has been incorporated only in representing the space, as it related to the Field-of-View of the user, but not a property of the overlaid image.

Turgeman and Levin [13] rely on spatial operations to a physical electronic device rather than a virtual 3D object in order to answer a human challenge. Piccolotto and Smith [9] rely on distinctive data relating to the user's video clip.

In the following idea, we describe a method for human user authentication (CAPTCHA) using the properties of AR/VR/MR to present spatially anchored virtual 3D objects such that multiple operations need to be performed on their 3Dness to reveal a secret or to answer a challenge question.

The remainder of this paper is organized as follows. In the next section, we propose our novel method for implementing 3D CAPTCHA challenges. Potential variations in implementation are illustrated in this section. Subsequently, we discuss key aspects including an exemplary architecture, and key considerations for implementation. We conclude by suggesting other application domains to which the proposed method may be extended.

4 A Novel Method

The procedural implementation of our method is similar to [8, 10] and [4], in the sense that the system activates a user's camera for the human challenge. The system, then similar to [8], presents a CAPTCHA challenge from its database based on user context. However, unlike [8], the system presents a 3D object to the user, such that the answer to the challenge is not related to user alignment with an overlaid image, but rather the answer to the challenge is occluded from user view and is only revealed in further exploration of the 3Dness of the object. This requires the user to do one or more of:

- Looking around the side of a virtual object (or objects) to reveal information on an invisible face (requires 6DOF capability in AR)
- Rotating a virtual object
- Opening and looking inside one virtual object to reveal another virtual object or objects
- Moving in space to reveal a virtual object behind a physical object
- Moving in space to move a foreground virtual object behind a physical object
- Moving one virtual object in view to reveal another virtual object
- Placing a virtual object with the boundaries of a matching physical object

5 Example Implementation Variations

5.1 Variation 1

Figure 5: 6DOF Rotation, Contact and Count Challenge.

In the Figure 5 implementation variation, a user sees a spatially anchored virtual dice and the challenge question requires him/her to walk or move around the object to reveal occluded secrets. In this case the challenge question may be:

- What is the total count on 3 faces on the die
- Touch each dot on all 3 faces of the die (front, top and right)

User movements are recorded and compared with either an answer or a set of gesture movements needed to complete the challenge. Despite the 3Dness of the challenge object, gestures can be executed on a 2D or 3D screen, not necessarily requiring a 3D screen at all times. For the cases of using a 2D screen, using the IMU sensors on a device (smartphone, tablet, smart glass) will enable the system to collect 6DOF 3D gesture information.

5.2 Variation 2

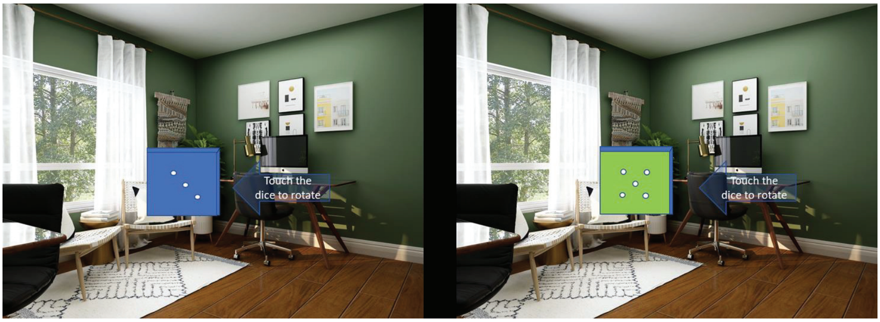

Figure 6: Locating a specific surface challenge.

In this challenge in Figure 6, the user is asked to touch a virtual object to rotate it. Every rotation simulates a throw of the dice. The user is asked to provide the count on the "green" face of the dice. After a random number of throws of the dice, green face is exposed to the user, and user provides an answer to the challenge question.

5.3 Variation 3

In this challenge in Figure 7, the user is required to physically move towards a virtual object and touch it to reveal an occluded secret. The secret is the answer to a challenge question. The user's movements are recorded.

5.4 Variation 4

In this challenge in Figure 8, the system places a virtual object occluded by a physical object. The system uses a computer vision-based depth map creation using AI/ML models [12] or LIDAR-based depth maps. The system spatially

Figure 7: Revealing an answer buried inside a 3D object.

anchors the virtual object in space and the user uncovers the occluded object to obtain the challenge answer. Different difficulty of challenges can be applied by the system as different steps of granularity of providing access. For instance, in order to make the challenge even more challenging, a partially occluded digital cat image is offered to the user in order to be able to differentiate from the real physical world and move it to bring the full scale of the digital object into focus.

Figure 8: Virtual 3D object “partially” occluded by a physical object.

Moreover, leveraging the computer vision algorithms mentioned above, the system can acquire a dominant color/shade present in the background physical environment and place a similar color mask on the digital object to make it blend more with the environment in order to make the challenge exercise more difficult.

5.5 Variation 5

In this challenge, the system places a virtual object occluding the entirety or portions of a physical object. The system uses a computer vision-based depth map creation using AI/ML models [12] or LIDAR-based depth maps. The system spatially anchors the virtual object in space, and as the challenge, the user is asked to move the virtual object behind the physical object. On screen the virtual model of the statue, in Figure 9, should be behind the person and farther however it appears as an overlay. This breaks the natural perception and breaks the experience. Hence, this challenge is a good way to isolate humans from bots.

Figure 9: Virtual 3D object occluding the physical object.

5.6 Variation 6

This is similar to variation 4 of the method, however, a first virtual object is placed in the physical environment that occludes a second virtual object. The second virtual object holds the answer to the challenge question. The user is required to perform movement to reach the first virtual object, move it (guided by the system), uncover the second virtual object, and then perform an operation to reveal the occluded secret. For instance, Figure 10 shows that the challenge for the user is to rotate the virtual wooden block to reveal the virtual green angry bird in its place and then tap it to make it fly.

5.7 Variation 7

The system locates a physical object in the environment and pulls a relevant digital object from its repository. Then the challenge is issued to the user to locate the best matching physical object that matches with the digital and perform the 6DOF gestures to match them spatially. For instance, in Figure 11, the user is expected to identify the bowl from the physical environment and place the bananas in the bowl. If any portion of the bananas stick out of the bowl after the user's movements, the user will fail the exercise.

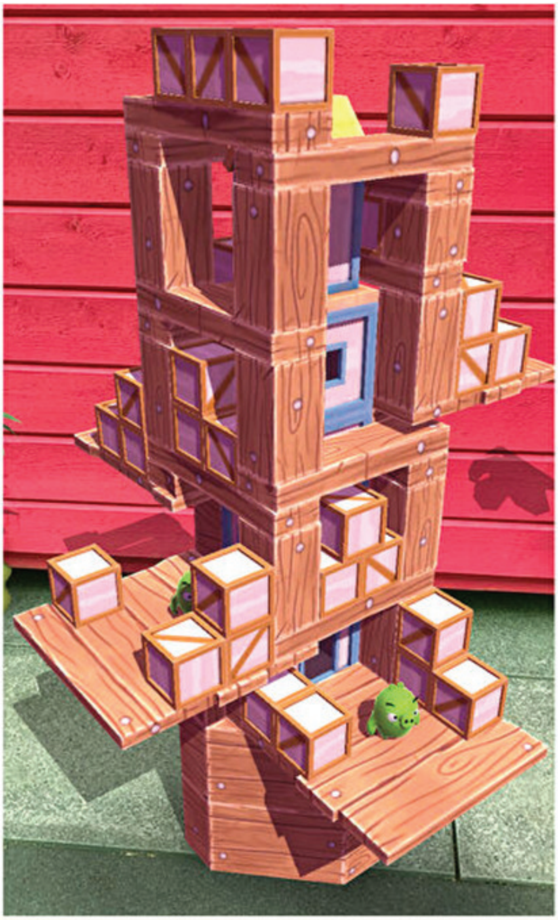

Figure 10: Virtual 3D object occluding another virtual object.

If, in addition to embedding a screen and camera, the user's device is a head mounted wearable, further 6DOF data collection can utilize eye tracking and head tracking to collect data to conform that the 6DOF data corresponds to what is being seen by the user device as opposed to being spoofed.

Furthermore, another method that is applicable to all the implementation variations disclosed above is utilizing spatial audio while issuing the challenge to the user. For this method, we assume that the system has access to a wearable audio device (such as Air Pods, earbuds or audio speakers on HMDs). The system will issue spatial audio commands or simple cues to direct the user to the virtual objects or directions of where the challenges are hidden (if

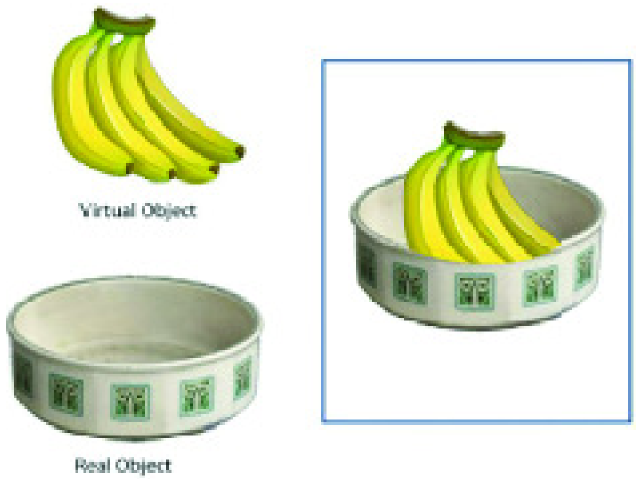

Figure 11: Relationally and spatially matching a physical and virtual object.

a very complex 3D model is used). This method will employ a HRTF (Head Related Transfer Function) data from the wearable audio device to ensure that the user is compliant with the audio clue before displaying the 3D model or updating the 3D model after user movement. This additional method acts as a secondary step of verifying the challenge answer and it can be used against a sophisticated bot attempting to solve the problem (and sometimes with success) however from a different angle, orientation or direction (as recorded by 6DOF data). This way, the system can enforce a specific 6DOF path and/or sequence for the challenge answer.

As a very basic example just from a cue perspective, as shown on Figure 12, the system may issue a spatial tone to the left ear that the virtual cat should be placed behind the leg, as the challenge. Then, the receiving user is expected to move the cat behind the left leg, as the physical object (man with the blue suit in this case) is mobile and walking through the physical space (optionally could be a virtual space). This method requires the system to continuously track the dynamic scene and the challenge object (the leg, in this case) via CV algorithms and match its location with respect to the virtual object in real-time.

In one implementation variation, only the user device or HMD is monitored for 6DOF movement while user action is identified using computer vision-based gesture recognition [3]. In this case the user device's camera is engaged to identify that a certain interaction that triggers model transformation has occurred. This could be touching a die to trigger a virtual roll of the die, touching a dot to change its color, touching a box to open it and reveal its

Figure 12: Directing a specific orientation for the virtual object.

contents, touching an object to reveal another object behind it, etc. In another variation, the user shall use hand, wrist or finger trackers, for example VR controllers, AR/VR gloves, pen, stylus or another similar device [1, 2, 6] that uses inertial sensors to calculate the position of the hand.

6 Discussion

The fundamental principle of CAPTCHA design is to devise problems that are easier to solve by humans rather than by machines, also known as CAPTCHA solvers. In its most frequent implementation this is typically done through retrieval of obfuscated images that are not easily machine-interpretable, due to difficulty in abstracting the associated metadata. On the other hand, humans may arrive at the answer to the challenge through visual inspection.

An important aspect of a CAPTCHA challenge is the hardness of the problem. Too much noise in an image may make it difficult even for a human, while too little may allow an attacker to feed the challenge to a processor with capabilities such as Optical Charter Recognition (OCR), Object and feature recognition etc. to successfully overcome the challenge. Thus, CAPTCHA challenges are not deterministically successful, rather only probabilistically so. This leads to an arms race between designers of CAPTCHA challenges, i.e., defenders and designers of CAPTCHA solvers, i.e., attackers.

6.1 Occlusion

Noise is often introduced into a CAPTCHA challenge by either showing only a partial object, or by occluding an object to some extent, making an image

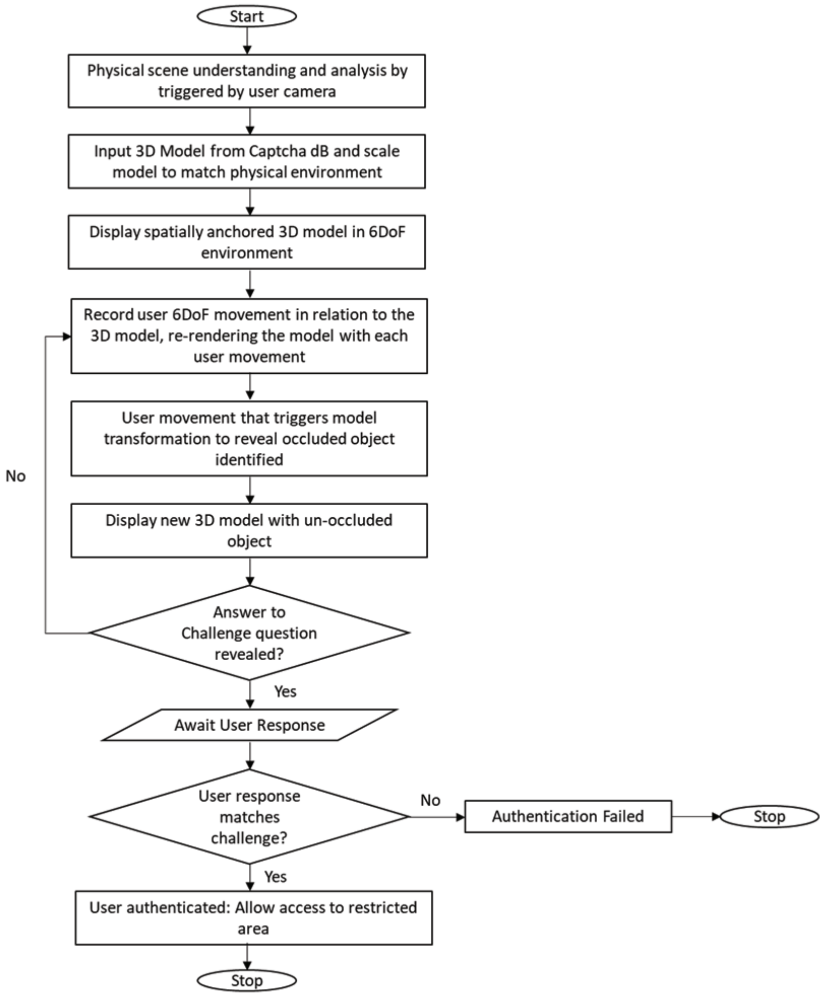

Figure 13: A high-level implementation of the novel method(s).

difficult to process accurately by a solver. The 3D CAPTCHA challenges as proposed in our method rely heavily on occlusion. Gao *et al.* [5] describe a use case of how occlusion has been used successfully in 2D CAPTCHA challenges. The authors show how a Visual Turing Test (VTT) CAPTCHA proposed by Tencent is improved using occlusion such that the attack success rate falls drastically from 86% to 69.5%, while the human pass rate is changed only imperceptibly from 93.9% to 92.9%. 3D rendering presents a new opportunity

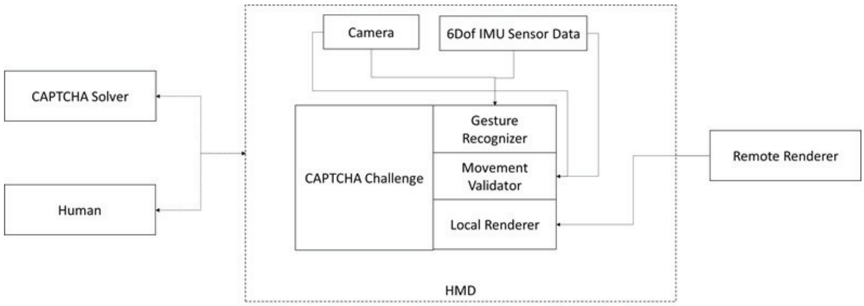

Figure 14: An exemplary architecture for the method(s).

to exploit occlusion, as intentional human movement to reveal a secret is perceptibly different from synthetic movement generated by a machine.

Figure 14 represents an exemplary architecture for implementation. An HMD poses the CAPTCHA challenge passed from a secure database, at least partially as a 3D model. Hardware resources such as a camera, and an inertial measurement unit (IMU) that collects 6Dof user data are at the disposal of higher layer software modules such as a CAPTCHA challenge. The hardware data is interpreted by software modules such as a human movement validator and a gesture recognizer that are in turn utilized by the CAPTCHA challenge software block. Finally, a local renderer provides the user a perspective-accurate representation of the synthetic objects as the human moves in the environment. The local renderer may utilize a remote renderer for obfuscation, as will be discussed shortly.

6.2 Trusted Execution Environment (TEE)

As operating systems for mixed reality devices evolve, it is expected that a trusted execution environment (TEE) will be used to allow secure access to certain peripherals such as cameras and IMUs. A TEE is an area on the main processor of a device that is separated from the system's main operating system (OS). It ensures data is stored, processed and protected in a secure environment. Placing the peripheral access into a secure environment provides a barrier to CAPTCHA solvers that may attempt to synthetically generate the data to pass the challenge. The implementation may thus be able to defend against attackers that spoof hardware data.

If attackers do gain access and replace, say IMU data with a synthetic stream, then other high-level mechanisms may be used to distinguish human movement from machine generated movement data. In some examples synthetic IMU data may not account for the obstacles in the environment. This may occur when the IMU data is not secure, but camera data is secured. In this case, the CAPTCHA challenge compares the received data to the mapping data of the environment generated using simultaneous localization and mapping (SLAM) capability of the HMD. The CAPTCHA challenge can then conclude that the movement data received is synthetic, since it represents an impossible movement. However, it is possible that both camera and IMU data are compromised. In this case, the challenge block will have to analyze the movement pattern to make a determination of data integrity. A solver may generate spatial (IMU) data that mimics a robot vacuum, since it is performing an exhaustive search in an attempt to capture a view that reveals the secret to the challenge. For example, the solver may generate a random walk [7] or use a spiral path algorithm [11] to discover a perspective that reveals the answer to the challenge. These movement patterns are distinguishable from intentional human locomotion and gesturing to reveal a secret.

6.3 Local-Remote Rendering

Split rendering is a paradigm wherein computation-intensive rendering is divided between a local processor, such as a GPU in the HMD, and a cloud processor such as a data center GPU. Its main benefit is the ability to leverage greater computation power in the cloud to perform computation-intensive environment sensing, visual computing, photorealistic rendering etc. However, to ensure a high-quality user experience without perception of lag or visual artifacts such as macro-blocking, some part of the computation/rendering is performed on-device. Here, we discuss how balancing local rendering with remote rendering can help to implement stronger CAPTCHA challenges.

Thus far, in illustrating our proposed method, we have assumed that the solver is not able to compromise and gain access to the 3D models used to present the CAPTCHA challenge. This holds true if the 3D models are rendered directly from the cloud, however, this makes the application latency sensitive. Alternately, to reduce the latency constraints the model data, is retrieved and loaded on the HMD in a completely secure area (typically models are loaded into VRAM). However, solvers may be able to compromise the 3D model data if not in a TEE.

Consider, for example's sake, that an increasingly popular paradigm of using a 3D scene description technology such as Universal Scene Description (USD) is used to implement the challenge. USD uses a scene description, i.e., a scene graph, to retrieve 3D models of assets from a future CDN. Then if this scene graph data is compromised, the solver can retrieve all the data about the object including its occluded perspectives. Armed with this information, it can "guess" the answer to the challenge question.

To overcome this drawback, we propose that the on-device CAPTCHA challenge software block may retrieve all the 3D model information from an online CDN for rendering, however, a remote renderer can generate and render

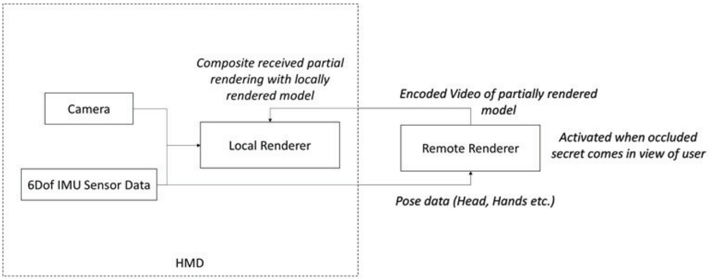

Figure 15: Remote renderer used in conjunction with local renderer.

a 3D feature to replace an aspect of the original model in real-time. The feature rendered by the remote renderer is encoded and sent in a video format to the HMD, that composites this rendering with the local rendering of the 3D model. Thus, gaining access to the 3D model does not compromise the CAPTCHA challenge. To implement this method, the movement data such as camera/IMU data would need to be sent to the remote renderer while an occluded secret comes into the user's view. In this manner, the system balances the practical need to reduce real-time bandwidth and latency constraints while also ensuring integrity of the data used for the CAPTCHA challenge. Figure 15 illustrates how the remote renderer obfuscates the locally rendered model with altered data.

7 Conclusion

The primary benefit of the proposed novel methods of 3D CAPTCHA within the mixed reality environment is to determine the distinction of humans from AI bots and they can be used from a core CAPTCHA feature perspective. However, they are not limited to only that and as such those skilled in the art may use them to implement many other applications related to AAA (access/authorization/authentication). Just a few such applications are listed below as follows:

- Providing free/freemium/paywall access to content
- Providing access to content for different user profiles (subscriber vs. free ad-sponsored user)
- Determining the level of difficulty or level of skill during a game

Though this list is just a minimum and not a comprehensive list, it is meant to intrigue and encourage the reader to contemplate potential future applications.

Note that, Mixed Reality interfaces will need to go through an adoption curve for the users to be able to use them at ease. It is expected that, like every HCI (Human Computer Interface) paradigm shift, users will struggle at first during the learning curve timeframe. There will be many enhancements in terms of user accessibility and experience over the immediate years as new device form factors are rolled out and purpose-built applications are designed. There may be risks that due to such issues some users may fail the aforementioned Mixed Reality Captcha tests. However, calibration and telemetry data can be captured via the IMU sensors to guide the users as well as biometric data can be collected to improve the tests even further as well as differentiating novice users from sophisticated bots. There are many opportunities for future research to advance the art beyond what has been described in this paper.

References

[1] M. Caeiro-Rodríguez, I. Otero-González, F. A. Mikic-Fonte, and M. Llamas-Nistal, "A systematic review of commercial smart gloves: Current status and applications", *Sensors*, 21(8), 2021, 2667.

[2] D. S. Eini, N. Z. Ratzon, A. A. Rizzo, S. C. Yeh, B. Lange, B. Yaffe, and R. Kizony, "Camera-tracking gaming control device for evaluation of active wrist flexion and extension", *Journal of Hand Therapy*, 30(1), 2017, 89–96.

[3] N. M. F. Franslin and G. W. Ng, "Vision-based dynamic hand gesture recognition techniques and applications: A review", in *Proceedings of the 8th International Conference on Computational Science and Technology: ICCST 2021, Labuan, Malaysia, 28–29 August*, Singapore: Springer Singapore, March 2022, 125–38.

[4] J. Ganeshmani, *Augmented Reality Generated Human Challenge*, (U.S. Patent No. US10860705), U.S. Patent and Trademark Office, https://patentimages.storage.googleapis.com/f2/03/9a/f3a5ff28e3e239/US10860705.pdf.

[5] Y. Gao, H. Gao, S. Luo, Y. Zi, S. Zhang, W. Mao, P. Wang, Y. Shen, and J. Yan, "Research on the Security of Visual Reasoning CAPTCHA", *USENIX Security Symposium*, 2021.

[6] M. Z. Iqbal, E. Mangina, and A. G. Campbell, "Exploring the real-time touchless hand interaction and intelligent agents in augmented reality learning applications", in *2021 7th International Conference of the Immersive Learning Research Network (iLRN)*, IEEE, May 2021, 1–8.

[7] H.-G. Kim, J.-Y. Yang, and D.-S. Kwon, "Experience based domestic environment and user adaptive cleaning algorithm of a robot cleaner", in *11th International Conference on IEEE Ubiquitous Robots and Ambient Intelligence (URAI)*, 2014, 176–8.

[8] V. J. Mantri, *CAPTCHA authentication via augmented reality*, (U.S. Patent No. US10832485B1), U.S. Patent and Trademark Office, https://patentimages.storage.googleapis.com/20/b2/44/b5421b508e53fa/US10832485.pdf.

[9] J. P. A. Piccolotto and N. M. Smith, "Fair, secured, and efficient completely automated public Turing test to tell computers and humans apart (CAPTCHA)", https://patents.google.com/patent/US10331945B2/.

[10] R. Y. Sawant, M. M. Kaladgi, J. A. Jameel, and S. Banisetti, *Augmented reality captcha*, (U.S. Patent No. US10339334B2), U.S. Patent and Trademark Office, https://patentimages.storage.googleapis.com/18/ec/f0/22517c22550f4d/US20180341779A1.pdf.

[11] K. Skrzypczyk and A. Pieronczyk, "Surface covering algorithms for semiautonomous vacuum cleaner", in *ACMOS 10*, 2010, 294–8.

[12] J. Theiner, M. Springstein, N. Nommensen, J. Rhotert, E. Müller-Budack, and R. Ewerth, "Analyzing Results of Depth Estimation Models With Monocular Criteria", in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, 3738–42.

[13] A. Turgeman and Z. Levin, "Device, system, and method of three-dimensional spatial user authentication", https://patents.google.com/patent/US20160300054A1/.

[14] L. von Ahn, M. Blum, N. J. Hopper, and J. Langford, *The CAPTCHA Web Page*, 2000, http://www.captcha.net.

[15] L. von Ahn, M. Blum, and J. Langford, "Telling Humans and Computers Apart (Automatically) or How Lazy Cryptographers do AI", To appear in Communications of the ACM.