2024 IEEE International Conference on Artificial Intelligence and eXtended and Virtual Reality (AIxVR) | 979-8-3503-7202-1/24/$31.00 ©2024 IEEE | DOI: 10.1109/AIxVR59861.2024.00010

Stealthy and Practical Multi-modal Attacks on Mixed Reality Tracking

Yasra Chandio
University of Massachusetts Amherst
ychandio@umass.edu

Noman Bashir
Massachusetts Institute of Technology
nbashir@mit.edu

Fatima M. Anwar
University of Massachusetts Amherst
fanwar@umass.edu

***Abstract*—Mixed Reality (MR) is rapidly becoming an essential technology for critical applications such as physical therapy and surgery, in addition to its early use for leisure activities. This transition necessitates a focused look at the security aspects of MR devices and applications. Prior work on MR security focuses on generic aspects such as secure authentication and vulnerability analysis. However, MR devices are multi-modal and spatiotemporal, which exposes them to attacks on sensor modalities across spatiotemporal axes. Prior work has demonstrated attacks on individual sensing streams, but modern, state-of-the-art sensor fusion-based tracking algorithms can easily mitigate such attacks.**

In this paper, we introduce a practical attack surface; it simultaneously launches attacks on multiple sensing streams across spatiotemporal axes to yield *effective*, *stealthy*, and *precise* outcomes. To the best of our knowledge, our work is the first to propose, design, and evaluate simultaneous multi-modal spatiotemporal attacks. In doing so, we solve key challenges in deciding what attack mechanisms to use, when to launch attacks, and how to configure attacks. Using the tracking and navigation use case of a user wearing an MR headset in real-world settings, we demonstrate the effectiveness of our attacks over the user's trajectory in the presence of state-of-the-art sensor fusion-based tracking algorithms and system checks.

***Index Terms*—mixed reality, tracking, sensor fusion, hololens**

I. Introduction

The emerging applications in Mixed Reality (MR) rely on multi-modal sensing and tracking of user activities and surrounding environment via commodity sensing devices such as Head Mounted Displays (HMD) [1], [2]. MR systems enhance their experience by designing human-in-the-loop systems [3], which are susceptible to security vulnerabilities that can impact the physical safety of humans [4]. These vulnerabilities, combined with the ubiquity of sensors, reduced field of view, and use in critical applications, make MR systems an attractive target for malicious activities targeting human safety [5]. To mitigate such vulnerabilities, recent work on MR security and privacy has explored issues of authentication [6], access control [7], and digital biomarkers [8]. While these studies target a private MR experience, they *do not explore threats to the security of fundamental MR services, such as tracking, that pose threats to users' physical safety*.

In MR systems, the headset's tracking capabilities are provided to the applications as a core service, akin to timing services. The prevalence of immersive and interactive applications raises concerns about potential physical harm to users relying on system services. If an adversary targets core services, all downstream applications suffer, especially when the attack goes unnoticed (stealthy) by the system or users. Unfortunately, such concerns are justified, as prior work has identified vulnerabilities in various stages of the application pipeline. For instance, prior studies have successfully demonstrated attacks on individual sensing modalities such as inertial sensors [9] and visual sensors [10]. These attacks have serious implications for the accuracy and security of tracking services that rely on data from these sensing modalities.

Prior work on developing secure tracking services uses data from multiple sensing modalities to mitigate threats on individual sensing streams. For example, sensor fusion algorithms like SelectFusion [11] combine data from visual and inertial sensing modalities to improve tracking accuracy. Sensor fusion-based tracking is effective against attacks on single sensor modalities, where traditional tracking approaches like VINet [12] fail. The fundamental principle behind SelectFusion tracking is that when data from one sensing modality deteriorates under attack, the fusion algorithm prioritizes other modalities to minimize degradation. Although sensor fusion successfully counters attacks on individual sensing modalities, other attack vectors in the MR tracking pipeline remain unexplored. Consequently, users of such platforms are vulnerable to attacks that can jeopardize their physical safety.

This paper proposes a novel multi-modal attack surface in MR tracking services resistant to sensor fusion-based tracking methods. Our key insight lies in attacking the fundamental principle of sensor fusion-based tracking by simultaneously attacking multiple sensing streams. However, launching multi-modal attacks simultaneously in a *stealthy*, *precise*, and *practical* manner poses significant challenges. The state-of-the-art multi-modal learning methods leverage distinct sensor strengths to estimate system states coherently. For instance, inertial streams with strong temporal components and higher frequency than visual streams are encoded using LSTMs [13]. Visual streams emphasize spatial components and are encoded using CNNs [14]. The spatial and temporal features from various modalities are complementary, forming a shared spatiotemporal attack surface, which is not trivial to exploit.

A successful implementation of our proposed multi-modal attack surface requires solving multiple challenges. A prerequisite is that the attacker can manipulate multiple sensing streams simultaneously. As detailed in Section III, prior work

This work was supported by the National Science Foundation under Grant No. 2237485 and 2230143.

2771-7453/24/$31.00 ©2024 IEEE
DOI 10.1109/AIxVR59861.2024.00010

has successfully demonstrated attacks on both visual [10] and inertial [9] sensing streams. However, the ability to launch simultaneous multi-modal attacks does not guarantee a desired outcome for the attacker. The non-visual streams, such as inertial data [15], have strict semantic constraints, and simple random perturbations are insufficient to deceive fusion-based algorithms, necessitating careful manipulations that avoid semantic inconsistencies. Also, stealthily launching practical attacks requires simple manipulations executed at precise moments to evade system checks or user observations. Lastly, these manipulations must be controlled to achieve precise and desired outcomes for the attacker.

We take an MR-assisted jogging application shown in Figure 1, where a user follows an avatar on a set trajectory at a fixed pace, as an example to demonstrate the efficacy of our attack surface. The goal of the attacker is to alter the jogger's speed and change their trajectory. In doing so, we make the following contributions in proposing and evaluating our novel multi-modal spatiotemporal attack surface.

1) We propose novel frame-level manipulations for sensor fusion applications that are simple, fast, practical, and free from semantic inconsistencies. In doing so, we introduce the notion of a frame for inertial data and develop spatiotemporal frame-level manipulations for inertial sensors.
2) We develop new metrics based on the similarity between consecutive frames, determining the threshold for manipulations to evade system checks and user observations. Our approach is systemic and iterative, addressing numerous low-level challenges that arise in ensuring stealthiness.
3) We propose the Right Frame Selection (RFS) algorithm, which guides an attacker on when to initiate attacks and adjusts the magnitude of manipulations to achieve precise outcomes. We leverage the RFS algorithm to propose multiple sophisticated attacks on the multi-modal spatiotemporal attack surface of the application shown in Figure 1.
4) We implement the tracking and attack pipelines to comprehensively evaluate our attacking approach in realistic settings regarding stealthiness, effectiveness, and precision.

II. Related Work

This section discusses related work on MR security, sensor attacks, sensor fusion algorithms, and critical MR applications.

MR Security. Prior work on MR security focuses on vulnerability analysis of application interactions [5], [10], [16], secure authentication and pairing [6], [8], [17], security across- users and devices [7], [18], and privacy in remote collaborations [1], [7]. No prior work explores the multi-modal attacks on fundamental tracking services we explore in this paper.

Spatiotemporal Sensor Attacks. There is significant prior work on sensor attacks across multiple dimensions, such as `data-level` temporal and spatial attacks on inertial [9], [19] and visual sensors [10], [20], `device-level` manipulation of timing and location services [21], [22], and spatiotemporal attacks to misalign content in MR and fool its tracking [8]. The state-of-the-art sensor fusion-based tracking algorithms can overcome these individual data- and device-level attacks.

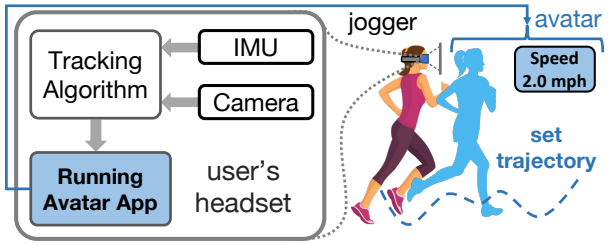


Fig. 1: ***An MR-assisted jogging application. A headset-wearing user follows an avatar on a set path for a run.***

In contrast, our proposed concurrent multi-modal attacks are resilient to sensor fusion algorithms and system checks.

Sensor Fusion. The state-of-the-art sensor fusion algorithms use deep learning (DL)-based tracking with visual-inertial odometry (VIO). In [11], the authors propose a robust DL-based multi-modal fusion for VIO, utilizing latent features from various sensor modalities to handle adverse conditions like noise, occlusions, misalignment, and missing data. In contrast, our work tackles malicious spatial and temporal misalignment, distorting the latent features' fundamentals. This raises new challenges in securing sensor fusion, necessitating exploring spatiotemporal attacks.

Security of Critical MR Applications. MR systems are designed for fun and to support users with serious objectives. It can enhance user experience in leisure non-critical applications such as navigating an unfamiliar college campus [23], museum [24], [25], and escape rooms [26], [27] to critical applications such as construction [28], maintenance [29], facility management [30], e-learning [31] to support highly critical applications like fitness [32], driving assistance [33], rehabilitation [34], and surgery [35]. MR can enhance applications' immersiveness, but its effectiveness relies on precise tracking, essential for meeting efficacy standards and safeguarding users against external attacks and internal malfunctions. These risks vary from minor inconveniences like wrong navigation to severe physical harm caused by collisions or reaching unsafe destinations. In medical surgeries where surgeons rely on accurate tracking for precise procedures [35], or in construction where workers collaborate remotely on complex structures [28], tracking errors could have perilous outcomes.

III. Threat Model

This section presents the threat model for the proposed multi-modal spatiotemporal attacks that aim to alter the trajectory of a user wearing an MR HMD (see Figure 1). There are two types of MR HMDs: opaque HMDs, e.g., Lynx R1 [36] or Vision Pro [37], relying on a video stream to perceive the physical world, allowing the attacker to manipulate both the tracking system and the video stream; and see-through HMDs, such as Hololens 2 [38], which provide direct view of the physical world, requiring attacks on the tracking system.

Our assumptions involve standard attacking capabilities. The attacker is physically close to the user, can visually access the surroundings, can utilize a smartphone to launch context-aware attacks [39], [40], and the attacker can choose the type of divergence or damage [41]. However, the attacker cannot

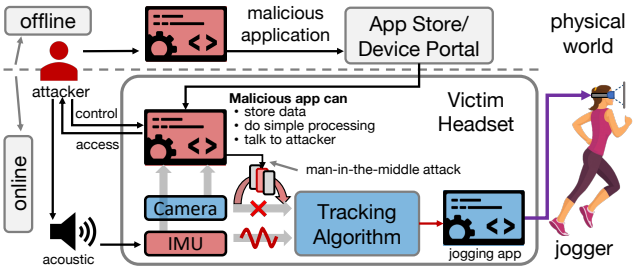


Fig. 2: ***Threat Model Overview.***

access the headset or its firmware. However, they can acquire a similar device to profile the behavior of sensor data, underlying firmware, sensor fusion algorithms, and system verifiers.

We next describe the key aspects of our threat model and the specific assumptions about our scenario, referring to Figure 2.

Malicious App Injection on MR HMD. Our attack methods rely on reading sensor data to time attacks (details in §IV). Eavesdropping on Hololens sensor data internally does not need privileges as motion sensors are zero permission sensors in recent devices [42]. Apps and background services can access sensors without specific permissions as operating systems lack fine-grained restrictions [43], allowing malicious apps to gather sensor data, as shown by prior work [42], [44], [45]. Such apps can continuously operate as background services to access data, like fitness trackers or virtual keyboards.

To gain data access, the attacker can disguise the malicious app using malware-based internal eavesdropping techniques through a device portal or offline app store [46]. By posing as a tracking service app, the attacker can perform real-time computations on the sensor data, establish wireless connections, and collaborate with nearby attackers [47], [48]. Attackers on the local network can also infiltrate the device portal to access sensor data from the malware [46]. Hololens 2 includes ample memory, multiple network channels, and computing power, allowing a malicious app to execute computations on sensor data, buffer frames, and transmit data to nearby devices.

Visual attack. The malicious app launches a man-in-the-middle attack[1] on visual frames that have been demonstrated on HMDs [8], [49]. Protection against such an attack can not be guaranteed unless the communication channels [50], data integrity, and user authentication are improved [51], [52]. Given such capabilities, the attacker can easily achieve the visual frame manipulations, including duplicating [53], dropping [54], and controlling the change between consecutive frames [55] via man-in-the-middle attacks [10]. We assume that she can buffer visual frames [20]. As our attacks do not use forgery or occlusions, system checks do not flag them.

Inertial attack. The inertial sensing stream is analog and cannot be attacked using the same method as the visual stream. However, prior work shows that an attacker can externally manipulate the analog signals on the signal conditioning path before digitization using acoustics signals [9], [19]. In §IV, we define the notion of frames for inertial data and discuss how manipulations of acoustic signals [9] can be used to manipulate

[1]An adversary positions themselves between the user and the system, intercepting and modifying the data exchanged between them.

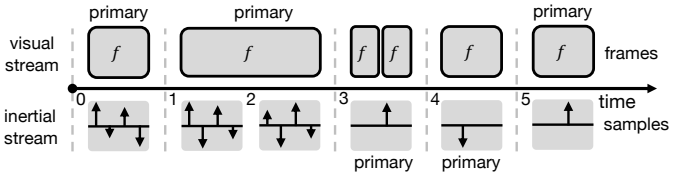


Fig. 3: ***The dynamic relationship between inertial samples and visual frames as sampling and frame rates change.***

inertial frames. Given this notion, the attacker can achieve the inertial frame manipulations needed for our attacks, including stretching, compressing, and shifting sensor measurements.

Given the computing power of modern handheld devices, it is feasible to launch both attacks concurrently. The attacker can use a smartphone's speaker to generate inaudible acoustic signals for inertial attacks [56] and control visual attacks through the malicious app. Prior work [9], [10], [19], [20], [53] has demonstrated both attacks; reimplementing them is outside the scope of our work. Additionally, when combined across modalities, the manipulations can resemble minor errors that the tracking algorithm may ignore. For example, spatial alterations may appear as axis misalignment and incorrect sensor calibration [57]. The time shift between input image windows and inertial measurement windows can be perceived as relative clock drift between independent sensor subsystems [58].

IV. Design

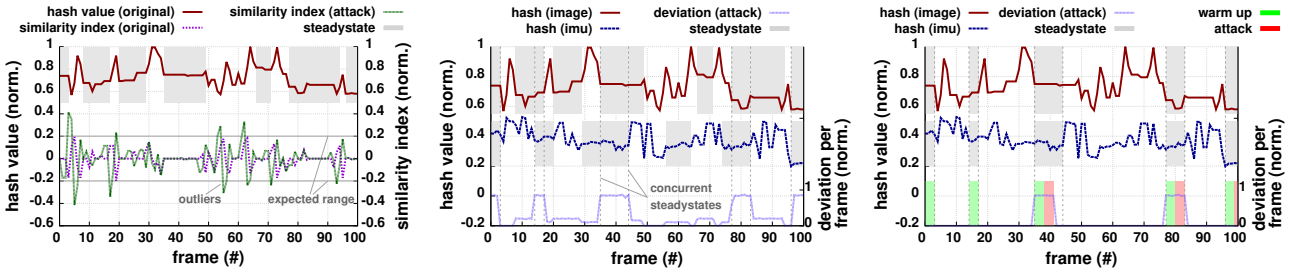
We present our approach to solving key challenges in enabling the proposed multi-modal spatiotemporal attacks. First, we introduce frame-level manipulations for inertial sensors (§IV-A). Second, we outline our approach to developing *stealthy*, *effective*, and *precise* attacks (§IV-B). Finally, we present attacks that achieve desired outcomes (§IV-C).

A. Physical Basis of Frame Level Attacks

A *frame* is intuitive for visual attack vectors; refers to individual images in a visual stream [54]. Manipulating visual frames, e.g., dropping, duplicating, or rotating, is also well-defined. However, *no concept of frames and manipulations exists for inertial sensors.* Prior work achieves precise amplitude modulation to print "WALNUT" [9] using inertial sensor output. However, they do not define the notion of a *frame* or devise frame-level manipulations. We introduce the concept of an *analog frame* and propose mechanisms for manipulating them. While we focus on inertial sensors, our approach extends to other analog sensors, e.g., audio and pressure.

Defining inertial frames. The data rates for various sensors vary due to resource or technology limitations [59], making developing a shared temporal basis across sensors a challenge. We propose a dynamic notion of an inertial frame that selects the low-frequency sensor as the primary reference and maps the data from the other sensor to the reference, as shown in Figure 3. Our mapping adapts to the dynamic rate of both sensors. This intuitive mapping simplifies the selection of visual frames and inertial samples to add/drop during attacks.

Manipulating inertial vector. We must control a sensor's output and map the control to frame-level manipulations to create a discrepancy between the true physical property a



(a) steadystate enables stealthiness (b) concurrency ensures effectiveness (c) parameterization yields preciseness

Fig. 4: ***Motivating design decisions: (a) every other frame is dropped as an attack, but attacks outside steadystate significantly change the similarity index, (b) each modality is attacked whenever it is in steady-state, but concurrency of steadystates makes attacks more effective, and (c) parameterizing attacks removes small steadystates to achieve precise deviation.***

sensor measures (signal) and its digitized representation (data). This should happen before data digitization, as the attacker cannot manipulate the data once it enters the tracking pipeline.

We attack the core sensing mechanism that converts the physical phenomenon into electric signals. As shown in prior work, we can deliberately perturb the sensor to manipulate the sensed signal in a controlled and precise manner. For instance, acoustic [60], [61] and optical [62], [63] signals can manipulate the conversion process of inertial and visual sensors, respectively. Based on Microelectromechanical systems (MEMS), modern inertial sensors consist of a sensing mass connected to a spring, displaced under acceleration to create a continuous voltage signal. Using acoustic waves, we can cause the inertial sensor's sensing mass to vibrate. A properly tuned acoustic frequency can predictably alter the output [9], [19].

Physical model of manipulations. Suppose the sensor captures a periodic motion, producing a time-varying signal $S_{orig}(t)$. We generate an acoustic signal, $S_{acou}(t)$, as our attack mechanism to manipulate the sensor.

$$S_{orig}(t) = A_{orig} \cdot sin(f_{orig}t)$$

$$S_{acou}(t) = A_{acou} \cdot sin(2\pi f_{acou}t + \phi_{acou})$$

Here, A, f, and ϕ are the signal's amplitude, frequency, and phase, respectively. f_{acou} equals sensor's resonant frequency.

The signals pass through amplifiers to limit the signal range and remove abnormal readings. Low pass filters (LPF) remove high-frequency components and satisfy the Nyquist requirement. The final signal is a combination of the two,

$$S(t) = S_{orig}(t) + k \cdot \ A_{att} \cdot S_{acou}(t).$$

The attenuation coefficient, A_{att}, is 1 at the resonant frequency, the natural frequency of a sensing object with the highest vibration amplitude. The attack frequency must match this frequency to displace the sensing mass effectively. The sensor's resonant frequency and output magnitude, denoted as A_{acc}, can be profiled using a similar device under acoustic effects to identify suitable attack frequencies or by posing as an app to obtain unrestricted read access to sensor data [46].

Add, drop, and misalign manipulations. In the visual stream, dropping a frame (image) rapidly changes the scene between consecutive frames, giving the impression of the user walking faster. To achieve a similar effect for inertial frames, we amplify the output of the inertial sensor by a factor of 2 ($k = 2$) along each axis (x, y, and z). Conversely, we divide the magnitude by 2 ($k = 1/2$) to create a slowdown effect. For misalignment, we multiply the signal by -1 for a 180° shift in user orientation ($\phi = 180°$). The high fidelity control [9] and the concept of the inertial frame form the basis of our attacks. Finally, the low computational cost of adding or dropping a frame in both attack vectors enables simultaneous attacks.

Matching inertial and visual manipulations. The attacker uses the malicious app with finite buffering capacity for the visual stream to drop or duplicate frames [20]. The camera frame rates for modern MR headsets vary significantly; e.g., HoloLens 2's frame rate fluctuates between 5-30 frames per second [38]. This will make frames' slow addition or drop appear as normal fluctuations. The dropping of an inertial sample differs from dropping an image, as their sampling rates differ. We pick a time-varying primary sensor that determines samples for the other streams for an equivalent effect.

B. Enabling Stealthy, Effective, and Precise Attacks

We start with a naive attack strategy and iteratively improve it to design *stealthy*, *effective*, and *precise* attacks.

1) ***Attack Environment****:* In our naive strategy, we manipulate alternate frames of visual and inertial attack vectors by dropping, adding, or misaligning them. We evaluate the efficacy of attacks using multiple metrics we define next.

Stealthiness. We measure stealthiness using the similarity index (SI), which quantifies semantic overlap between adjacent frames (f_i and f_{i+1}) and is used in VIO [64]. A 0 similarity index means no overlap; 1 means identical frames. We use the hamming distance between perceptual hashes for visual frames as SI [65], which is robust to minor changes and effective in error detection [66], [67]. We use locality-sensitive hashing and scale-insensitive cosine distance for inertial frames [68].

Effectiveness. Effectiveness is measured as the deviation from the original trajectory. A successful attack yields significant deviation. The definition of *significant* depends on the context, may vary across different scenarios, but always increases monotonically with increased attack frequency or duration.

Preciseness. It evaluates an attack's ability to achieve a desired deviation consistently. This means an attack of a specific duration should consistently result in the same deviation.

2) ***Achieving Stealthiness****:* Figure 4a shows the visual attack vector's hash values and similarity index under our naive

Algorithm 1: Right Frame Selection (RFS) algorithm.

```
Input: frames, history_length, warmup_length,
       attack_length, SI_upperbound, SI_lowerbound,
Output: attack flag
attack flag ← 0; steadystate_counter ← 0
attack_counter ← 0; history ← FIFO buffer: [0: history]
while frames arrive do
    history.append( frame)
    if (history is full) then
        get similarity index (SI) of adjacent frames
        if (similarity index is within bounds) then
            increase steadystate_counter
        else
            steadystate_counter ← 0; attack flag ← 0
        if (steadystate_counter ≥ warmup_length) and
          (attack_counter ≤ attack_length) then
            increase attack_counter; attack flag ← 1
    else
        attack flag ← 0
```

strategy. The similarity index between consecutive frames is within a tight range without malicious data. However, attacking during scene transitions can drastically change the similarity index due to rapid frame changes. System checks can detect such anomalies and alert the user to a potential attack or close the application. Conversely, under stable scenes, the frame-dropping attack does not yield similarity index outliers. This suggests that the attack vector's state can help determine the optimal timing for stealthy attacks. Interestingly, the metric used to detect attacks also enables stealthy attacks.

Stealthiness Takeaway. *System checks can detect attacks launched during big scene changes. Launching fast attacks during the attack vector's steadystates can be stealthy.*

3) ***Being Effective****:* While attacks during steady states are ideal, the steady states of sensing streams do not always align. In Figure 4b, we show the deviation achieved when attacks are launched during each sensor's steady state. We observe that the deviation is small when a single attack vector is attacked, even if severely, and a simultaneous attack on both streams yields a larger deviation than the sum of individual attacks. Also, launching attacks when both vectors are in a steady state enables alignment and synchronization of individual attacks. It allows the attacker to selectively launch attacks in specific contexts, such as when the scene is stable.

Effectiveness Takeaway. *The state-of-the-art sensor-fusion-based tracking algorithms can mitigate attacks on individual streams. However, a concurrent attack on both streams yields a bigger deviation that cannot be mitigated.*

4) ***Targeting Precision****:* Figure 4b shows aligned steady state windows of varying lengths for both attack vectors. However, accurately determining these windows in practice is non-trivial. The attacker needs to estimate the start of the steady state and determine when to conclude the attack.

Our approach monitors frame similarity indices and establishes a "normal" range or the steady state, defined by upper and lower bounds for an attack vector. The steady state starts when a specified number of frames (warm-up length) fall within the range. Once in a steady state, the attack initiates and continues for a set number of frames (attack length). Figure 4c shows this approach, where the attacker waits for the warm-up period (green windows) to launch the attack. This eliminates short concurrent windows and allows controlling deviation by terminating the attack during large windows. These parameters provide a trade-off between stealthiness, effectiveness, and preciseness. Choosing a shorter warm-up period increases attack effectiveness but reduces precision and stealthiness. A longer warm-up period ensures precision and stealthiness but reduces available steady states and effectiveness. Similar trade-offs apply to the attack length: longer attacks are effective but lack stealthiness and preciseness, and vice versa.

Attack	Attack Mechanism	Attack Vector	Attack Knob	End Effect
SpeedUp	Drop	Camera	Temporal	App perceives speedup; prompts jogging slower.
		IMU	Temporal	
SlowDown	Add	Camera	Temporal	App perceives slowdown; prompts jogging faster.
		IMU	Temporal	
Zero Displacement	Add, Drop, Hist. Shift	Camera	Both	Reaches the destination using a different path.
		IMU	Temporal	
Path Deviation	Misalign, Hist. Shift	Camera	Spatial	Unable to reach the destination.
		IMU	Spatial	

TABLE I: ***Proposed multi-modal spatiotemporal attacks.***

We formalize our approach as a simple algorithm, called Right Frame Selection (RFS), which determines when to attack (shown in Alg. 1). We next define the key terminologies.

- **History Length (`history_length`):** The number of frames an attacker stores to decide on steadystate bounds.
- **SI upperbound (`SI_upperbound`):** The X%ile value for the stored history. X is determined by offline profiling.
- **SI lowerbound (`SI_lowerbound`):** The (1-X)%ile value for the stored history. X is based on offline profiling.
- **Warmup Length (`warmup_length`):** The number of frames that need to be within bounds to declare steadystate.
- **Attack Length (`warmup_length`):** The number of frames that are attacked within a steadystate.

Algorithm 1 automates frame selection for stealthy attacks. Its input is a stream of *frames* and attack parameters, producing the `attack_flag` as an output. A value of 1 hints the attacker to continue; 0 hints the attacker to stop. Concurrency is ensured by attacking only when the `attack_flag` is 1 for all attack vectors. RFS stores incoming frames using a FIFO buffer of size `history_length`. Once the buffer is full, we calculate the similarity index for all frames and increment the steady state counter if the index is within range. Steady state is achieved when the scene remains consistent and the similarity index stays within the range for `warmup_length` frames. In a steady state, the RFS algorithm signals to initiate the attack. The attack continues until the similarity index exceeds the range or the number of attacked frames reaches `attack_length`. All counters are reset now, and the algorithm resumes searching for the next steady state.

Preciseness Takeaway. *The attacker needs a simple configurable attacking approach to balance attack effectiveness and preciseness. Finding the right configuration is crucial.*

C. Proposed Attacks

We present four lightweight attacks that achieve a concrete and quantifiable goal for the attacker, summarized in Table I.

1 - SpeedUp Attack. This attack drops frames across both attack vectors. The attacker drops images to create a fast-changing scene and reduces the inertial frame amplitude to achieve the same effect. The speedup illusion prompts the jogger app to slow the user, leading to ineffective jogging [69].

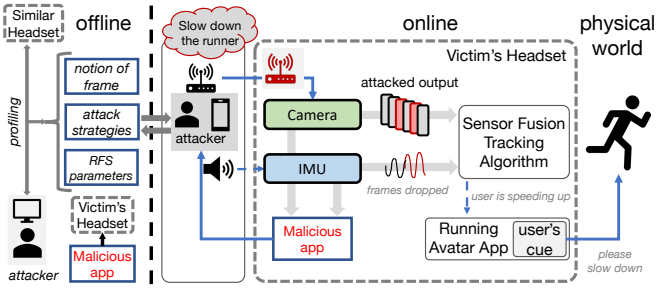


Fig. 5: ***Multi-modal concurrent attack strategy overview consisting of offline profiling and online attack stage.***

2 - SlowDown Attack. This attack adds frames to the sensing streams. The attacker duplicates the old images and reduces the signal amplitude for the visual and inertial attack vectors. The pacer app perceives a user slowed down and makes her run faster, leading to harmful consequences [70].

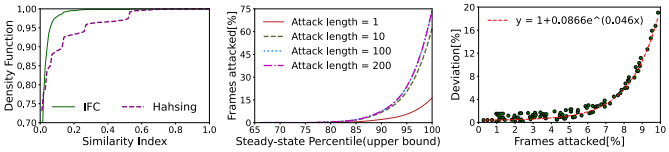
3 - Zero Displacement Attack. This is the *most challenging* attack that redirects the user from the $source$ to the $destination$ through a new path leveraging frame-level manipulations, such as adding, dropping, misaligning, and histogram shifting. In this attack, we assume the attacker knows the destination. The tracking algorithm relies on frames from i to $i+m$ to determine the user's position and orientation [71]. The attacker employs misalign manipulation to change the user's orientation away from the set path. The attacker can alter the orientation back towards the set path at any point to ensure the user reaches the destination. For a "misalign" attack on the visual stream, the attacker employs a histogram shift attack that keeps the histogram difference between adjacent frames below a threshold. Our implementation detects changes in frames using edge detection and applies the *difference image* to minimize significant changes in the histogram difference.

4 - Path Deviation Attack. This attack uses *misalignment* and *histogram shift* manipulations to disrupt spatial alignment for the inertial and visual frames, similar to the path deviation attack. However, it sets a destination point different from the original destination. The incorrect pose data affects the tracking application, causing the avatar to deviate from the set path and leading the user towards the new destination.

D. Putting All Things Together

We have described the components for launching stealthy, effective, and precise attacks. The attacker aims to deviate the jogger from their trajectory or speed. To achieve this, we present an attack pipeline with two stages in Figure 5.

In the offline phase, the attacker installs a malicious app on the victim's device with read-only access to the camera and sensors. They analyze frame rates for attack vectors and profile the relationship between RFS algorithm parameters and deviations. This data helps them create attack strategies and gather online stage parameters. In the online phase, the attacker sets a goal, like slowing the user down, and uses the profiled info to drop frames based on the context. They receive data from the app, including RFS algorithm similarity index bounds. By launching attacks on specific vectors, they trick



(a) Hashing vs IFC (b) Effect of bounds (c) Achieved deviation

Fig. 6: ***Configurability and Preciseness of Attack Surface.***

the tracking algorithm into thinking the user is jogging faster. The app provides user feedback to achieve the attacker's goal.

These attacks enable speed manipulation and deviation and serve as a foundation for more sophisticated attacks on safety-critical applications. An attacker can achieve specific malicious tracking objectives by combining different attack types.

V. Evaluation

This section evaluates proposed spatiotemporal multi-modal attacks' preciseness, efficacy, stealthiness, and robustness.

A. Evaluation Setup

We mimic the real-time setup of an MR device using the Robot Operating System (ROS) [72]. We detail our setup next.

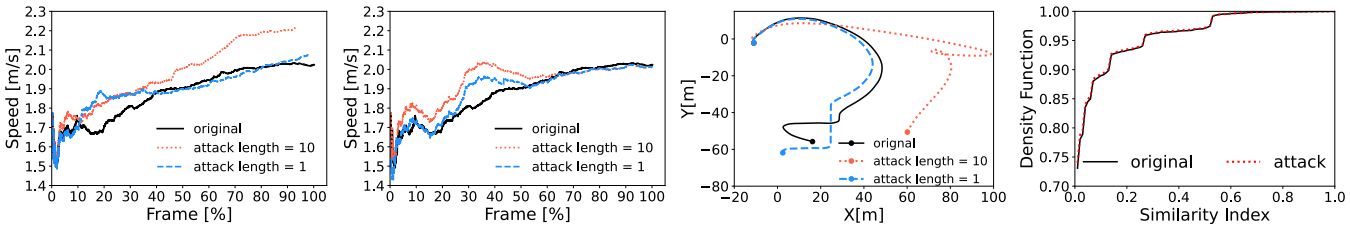
Data. We collected real-world data by simulating a jogger with a partner avatar in different environments (indoor, outdoor) and scene setups (trails, suburbs, downtown). We used Hololens 2 in research mode [38], which includes an RGB camera, four grayscale visible light tracking (VLC) cameras, and an inertial measurement unit. The VLC cameras recorded data at 5-30 fps, and the inertial sensor data was collected at 12-20 samples per second [73]. We synchronized the inertial data with images. We gathered ∼154k samples, equivalent to 2.5 hours of data with an average frame rate of 17 fps. We released the dataset for public use as HoloSet [74].

Tracking. Since the exact tracking algorithm used in Hololens 2 is unknown, we use the state-of-the-art sensor fusion-based tracking algorithm Select-Fusion [11]. Our model combines raw images and inertial frames for pose transformations. We train using collected data, with a 75%-25% train-test split. We train Select-Fusion in stochastic fusion mode with PyTorch on an NVIDIA GeForce RTX 2070 GPU, batch size of 8, Adam optimizer, and learning rate of $1e^{-4}$. The mean squared errors for training and testing were 0.0095 and 0.017, respectively.

Metrics. To evaluate system behavior under attack, we compare the Cumulative Density Function (CDF) of the similarity index before and after the attack. CDF is commonly used for attack detection [75]; if the frame distribution remains unchanged, the system is not perceived as under attack. We use the Kolmogorov-Smirnov test to measure distribution similarity [76]. A higher difference in arithmetic means signifies higher variation among similarity indices.

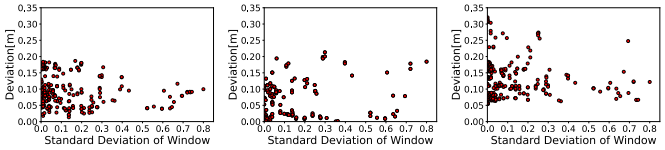
B. Configurability and Preciseness of Attacks

As outlined in §IV-B4, an attacker can launch precise attacks by configuring: (i) the percentage of frames to attack and (ii) the number of frames attacked in an attack sequence. They can configure these parameters using empirical experiments that quantify the relationship between parameters and deviation. Below, we present the results of such experiments.



(a) SpeedUp (b) Zero Displacement (c) Path Deviation (d) SI (Path Deviation)

Fig. 7: ***The effect of SpeedUp (a) and Zero Displacement (b) attacks on the eventual speed of the jogger. We show the new trajectory under attack (c) for the path deviation attack. We also show the similarity index (SI) for (c).***



(a) Speedup (b) Zero Disp. (c) Path Deviation

Fig. 8: ***Effect of steadiness of a given window on the deviation achieved in that particular window. Low variation in a given window results in a more controllable and effective attack.***

1 - Sensitivity Analysis of Similarity Index Metrics. Our attack approach relies on frame similarity to configure the attacks. Since the actual metric used by platforms like Hololens 2 is not publicly available, we compare two commonly used metrics: Inter Frame Correlation (IFC) and hashing. Figure 6a displays the CDF of hashing and IFC for a single sequence. IFC fails to detect the subtle variations in the frame similarity index, while hashing provides a more nuanced distribution. We use hashing for subsequent experiments.

2 - Impact of Steady-state Window. The attacker can manipulate the similarity index bounds to balance stealth, effectiveness, and preciseness (see Section IV-B4). Figure 6b shows the impact of steady-state bounds and attack length on the percentage of frames attacked. Setting the bounds too low results in fewer frames that can be attacked, while a 90% upper-bound value encompasses almost all frames. Also, the number of attacked frames increases as the attack length goes from 1 to 10, with marginal growth beyond that.

3 - Stealthiness vs. Deviation. The final step is to map the number of attacked frames to the deviation. Figure 6c illustrates the exponential relationship between the frames attacked and the resulting deviation. The attacker selects attack parameters carefully for stealthiness and to avoid large deviations. However, the attacker can profile the relationship effectively since it remains consistent within a given environment.

Risk-reward trade-off. *A large deviation requires attacking more frames, increasing the risk of detection. A small attack is stealthy but fails to achieve the desired outcome. An operation within the 85%-95% steady-state bound and an attack length of 1 enables a steady relationship. Targeting only 10% of the frames can achieve a deviation of 2%-20%.*

C. Effectiveness and Stealthiness of Attacks

In this section, we evaluate the effectiveness and stealthiness of our proposed attacks. Since the SlowDown attack is similar to the SpeedUp attack, subsequently, we only show the latter.

1 - Effectiveness. We measure the effectiveness of attacks using the deviation from the intended speed or trajectory. Figure 7 presents the time series of jogger speed before and after 1-frame and 10-frame attacks for *SpeedUp* and *Zero Displacement* attacks. The longer attack lengths cause significant speed changes, up to 0.3 m/s, while shorter attacks have minimal impact. Similarly, large attack lengths are needed for the *Path Deviation* attack if the new destination points are farther away from the original destination. At an attack length of 10, the attacker achieves a 32-meter change in position and 1.5 radians change in the orientation. Note that all attacks require an initial warm-up period where sequences match.

Key result. *Our attacks are successful at smaller and larger attack lengths, demonstrating their sensitivity to attack parameters and effectiveness against our attack surface.*

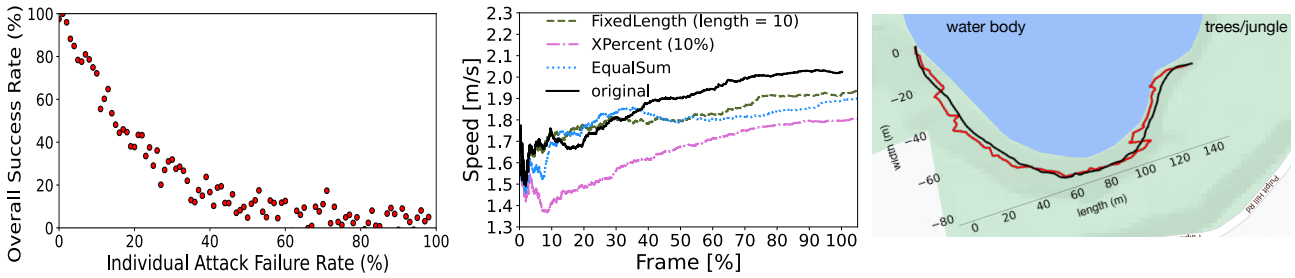
2 - Stealthiness. The stealthiness of our attacks is measured by their ability to achieve the desired goal without significantly altering the distribution of the similarity index. Figure 7d presents the distribution of the hashing function for the path deviation attack before and after an attack of length 10. The attack does not alter the distributions significantly. Note that the y-axis starts from 0.7 to highlight even the minute changes.

Key result. *Our attacks are stealthy despite achieving significant changes in speed and trajectory for a wide range of attacks and cannot be detected by the system checks.*

D. Effect of Steadiness on Attacks

Our proposed attacks are executed during the steady state, where the frame similarity index can vary depending on the environment. If the scene is unchanged, frames exhibit high similarity, and vice versa. To evaluate our attacks under different steady-state scenarios, we show the achieved deviation for all the attacks against the standard deviation of the similarity index within a window in Figure 8. Our attacking approach selects steady-state windows with low standard deviation, indicating minimal scene changes to ensure stealthiness. Furthermore, our results demonstrate that high deviation can be achieved even when scenes undergo minimal changes. Consequently, our attack methodology applies to critical applications such as surgery [35] and rehabilitation [77] that require precise micro-movement tracking in a stable environment.

Key result. *Our attacks are effective in stable and dynamic scenes, making them applicable to critical applications such as surgery, where scenes remain relatively stable.*



(a) Robust to Attack Failure (b) Robust to Frame Picking (c) Robust to Real World

Fig. 9: ***Effect of a window's steadiness on the deviation achieved (a) and the robustness of attacking approach against (b) attack failures, (c) configuration errors, and (d) real-world scenarios.***

E. Robustness of Attacks

We demonstrate the robustness of our attacks against frame-level attack failures, frame selection, and real-world scenarios.

1 - Attack Failure. The attacker must manipulate sensor data in real-time (see §V-F) without complete information on the steady states of all sensors leading to some attacks failing. Figure 9a shows a non-linear relationship between frame-level attack failures (x-axis) and the achieved deviation compared to no failure (y-axis). The initial failures significantly reduce the achieved deviation. However, the attacker can alter the user's speed from 2 m/s to 2.1 m/s (intended 2.2 m/s) despite a 20% frame-level attack failure. Even when most frame-level attacks fail, our approach achieves a small deviation. Note that a 20% failure rate for such manipulations is an overestimate [9], [10].
Key result. *Our attacks are sensitive to attack failure rate but still achieve most of the intended deviation.*

2 - Frame Selection. We next show that an arbitrary frame selection is less effective, less configurable, and less stealthy. *FixedLength* approach launches periodic attacks of length ten every N frames. *EqualSum* launches periodic attacks but attacks the same total frames as RFS. *XPercent* attacks X% of the frames in each sequence. Figure 9b shows all frame-picking schemes when launching a *slowdown* attack. The attacker cannot launch: a consistent attack (*FixedLength*), a precise attack (*EqualSum*), or a stealthy attack (*XPercent*). Table II compares attacks using *ks* value and speed change.
Key result. *Arbitrary frame-picking schemes, besides RFS, cannot launch effective, precise, or stealthy attacks.*

3 - Real World. Figure 9c shows the original trajectory (black line) and under-attack (zero-displacement attack, red line) trajectories mapped to the real world. Our attack achieves higher deviation when the path is straight and avoids attacking when the scene changes or the user turns, showing context awareness. Despite not using any technique to map surroundings, its steady state implicitly incorporates that information.
Key result. *Our attacks are effective in the real world and adapt their parameters to match the surrounding environment, demonstrating context-awareness and adaptation.*

F. Microbenchmarks

Our benchmarking shows that our attacks are fast, simple, and suitable for the MR environment. The attacker must make decisions, compute similarity indices, manipulate frames, and complete the attack within 33.33ms. The average inertial similarity index computation takes 0.8ms (1.5ms max, 0.7ms min). For visual attacks, it is 10ms on average (13.8ms max, 9.2ms min). Frame manipulation takes 5ms on average, and the RFS algorithm runs in 0.0011ms. The average time is less than 24ms, ensuring fast and stealthy attacks.

Attack	Attack percentage	ks test statistic	Mean difference
SlowDown	7.2	0.06	0.14
Speedup	7.2	0.07	0.11
Zero Displacement	7.2	0.05	0.03
Path Deviation	7.2	0.07	0.15
FixedLength	4.5	0.07	0.23
EqualSum	7.2	0.08	0.35
XPercent	10	0.13	0.49

TABLE II: ***Comparison of different frame picking schemes.***

VI. Conclusion and Future Work

In this paper, we presented a novel multi-modal spatiotemporal attack surface for MR systems. We propose an attacking approach that allows launching attacks that yield *effective*, *stealthy*, and *precise* outcomes. Our evaluations demonstrate the efficacy of our attacks against the state-of-the-art sensor fusion-based tracking algorithm and demonstrate that the attacker is able to achieve the goal across a wide range of environments and user action scenarios.

In the future, we plan to conduct research to both improve our attacking approach as well as develop defense mechanisms against our proposed attacks. First, we will explore machine learning techniques to find the right time to attack and develop context-aware optimization techniques for maximizing the attack impact and stealthiness. We plan to explore use cases for our approach in safety-critical applications such as surgery that include micro-movements.

Second, on the defense side, we aim to devise multi-modal learning mechanisms to neutralize the proposed attack surface. To the best of our knowledge, there is no single approach that can be used to mitigate the effect of our proposed attacking approach. However, a combination of online intrusion detection [78], acoustic dampening [79], and filtering [9] or visual challenges [20] can be used to mitigate the efficacy of the attacking approach to some extent.

References

[1] J. Happa, M. Glencross, and A. Steed, "Cyber Security Threats and Challenges in Collaborative Mixed-Reality," in *Frontiers in ICT*, 2019.

[2] S. Andrist, D. Bohus, A. Feniello, and N. Saw, "Developing Mixed Reality Applications with Platform for Situated Intelligence," in *IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*, pp. 48–50, 2022.

[3] S. Rokhsaritalemi, A. Sadeghi-Niaraki, and S.-M. Choi, "A review on mixed reality: Current trends, challenges and prospects," in *Applied Sciences*, vol. 10, MDPI, 2020.

[4] C. Warin and D. Reinhardt, "Vision: Usable Privacy for XR in the Era of the Metaverse," in *European Symposium on Usable Security (EuroUSEC)*, 2022.

[5] K. Lebeck, T. Kohno, and F. Roesner, "How to Safely Augment Reality: Challenges and Directions," in *ACM International Workshop on Mobile Computing Systems and Applications (HotMobile)*, 2016.

[6] I. Sluganovic, M. Serbec, A. Derek, and I. Martinovic, "HoloPair: Securing Shared Augmented Reality Using Microsoft HoloLens," in *Annual Computer Security Applications Conference (ACSAC)*, (New York, NY, USA), p. 250–261, ACM, 2017.

[7] X. Ran, C. Slocum, M. Gorlatova, and J. Chen, "ShareAR: Communication-Efficient Multi-User Mobile Augmented Reality," in *ACM Workshop on Hot Topics in Networks (HotNets)*, 2019.

[8] R. A. Sharma, A. Dongare, J. Miller, N. Wilkerson, D. Cohen, V. Sekar, P. Dutta, and A. Rowe, "All that GLITTERs: Low-Power Spoof-Resilient Optical Markers for Augmented Reality," in *ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*, 2020.

[9] T. Trippel, O. Weisse, W. Xu, P. Honeyman, and K. Fu, "WALNUT: Waging Doubt on the Integrity of MEMS Accelerometers with Acoustic Injection Attacks," in *IEEE European Symposium on Security and Privacy (EuroS&P)*, pp. 3–18, 2017.

[10] P. Casey, I. Baggili, and A. Yarramreddy, "Immersive Virtual Reality Attacks and the Human Joystick," *IEEE Transactions on Dependable and Secure Computing (TDSC)*, vol. 18, no. 2, pp. 550–562, 2021.

[11] C. Chen, S. Rosa, Y. Miao, C. X. Lu, W. Wu, A. Markham, and N. Trigoni, "Selective Sensor Fusion for Neural Visual-Inertial Odometry," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[12] R. Clark, S. Wang, H. Wen, A. Markham, and A. Trigoni, "VINet: Visual-Inertial Odometry as a Sequence-to-Sequence Learning Problem," in *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2017.

[13] B. Wagstaff and J. Kelly, "Lstm-based zero-velocity detection for robust inertial navigation," in *IEEE IPIN*, 2018.

[14] P. Dhruv and S. Naskar, "Image classification using convolutional neural network (cnn) and recurrent neural network (rnn): a review," *Springer ICMLIP*, 2020.

[15] T. Simonetto, S. Dyrmishi, S. Ghamizi, M. Cordy, and Y. L. Traon, "A unified framework for adversarial attack and defense in constrained feature space," *ArXiv preprint arXiv:2112.01156*, 2021.

[16] K. Lebeck, T. Kohno, and F. Roesner, "Enabling Multiple Applications to Simultaneously Augment Reality: Challenges and Directions," in *ACM International Workshop on Mobile Computing Systems and Applications (HotMobile)*, 2019.

[17] E. Gaebel, N. Zhang, W. Lou, and Y. T. Hou, "Looks Good To Me: Authentication for Augmented Reality," in *ACM Proceedings of the International Workshop on Trustworthy Embedded Devices (TrustED)*, 2016.

[18] F. Zhu and T. Grossman, "BISHARE: Exploring Bidirectional Interactions Between Smartphones and Head-Mounted Augmented Reality," in *ACM Conference on Human Factors in Computing Systems (CHI)*, 2020.

[19] Y. Tu, Z. Lin, I. Lee, and X. Hei, "Injected and Delivered: Fabricating Implicit Control over Actuation Systems by Spoofing Inertial Sensors," in *USENIX Security Symposium*, 2018.

[20] J. Valente, K. Bahirat, K. Venechanos, A. Cardenas, and P. Balakrishnan, "Improving the Security of Visual Challenges," *ACM Transactions on Cyber-Physical Systems (TCPS)*, 2019.

[21] F. M. Anwar and M. Srivastava, "A Case for Feedforward Control with Feedback Trim to Mitigate Time Transfer Attacks," *ACM Trans. Priv. Secur.*, vol. 23, may 2020.

[22] F. M. Anwar, L. Garcia, X. Han, and M. Srivastava, "Securing Time in Untrusted Operating Systems with TimeSeal," in *IEEE Real-Time Systems Symposium (RTSS)*, pp. 80–92, 2019.

[23] F. Lu, H. Zhou, L. Guo, J. Chen, and L. Pei, "An ARCore-Based Augmented Reality Campus Navigation System," *Applied Sciences*, vol. 11, 2021.

[24] M. Trunfio, T. Jung, and S. Campana, "Mixed reality experiences in museums: Exploring the impact of functional elements of the devices on visitors' immersive experiences and post-experience behaviours," in *Information & Management*, vol. 59, Elsevier, 2022.

[25] J. Al Rabbaa, A. Morris, and S. Somanath, "MRsive: An Augmented Reality Tool for Enhancing Wayfinding and Engagement with Art in Museums," in *HCI International 2019 - Posters* (C. Stephanidis, ed.), (Cham), pp. 535–542, Springer International Publishing, 2019.

[26] F. Wild, L. Marshall, J. Bernard, E. White, and J. Twycross, "UNBODY: A Poetry Escape Room in Augmented Reality," *Information*, vol. 12, 2021.

[27] Á. Gómez-Cambronero, A. González-Pérez, I. Miralles, and S. Casteleyn, "Mixed Reality Escape Room Video Game to Promote Learning In Indoor Environments," in *Edulearn Proceedings*, 2019.

[28] J. Zhao, E. Pikas, O. Seppänen, and A. Peltokorpi, "Using real-time indoor resource positioning to track the progress of tasks in construction sites," in *Frontiers in Built Environment*, vol. 7, 2021.

[29] L. C. Moreira, R. C. Ruschel, and A. H. Behzadan, "Augmented reality for building maintenance and operation," in *Springer Handbook of Augmented Reality*, pp. 495–532, Springer, 2023.

[30] B. Schwald and B. De Laval, "An Augmented Reality System for Training and Assistance to Maintenance in the Industrial Context," in *International Conference in Central Europe on Computer Graphics and Visualization (WSCG)*, UNION Agency–Science Press, 2003.

[31] S. G. Aekanth, *Transforming E-Learning Through the Use of Virtual and Augmented Reality: A Systematic Review*, pp. 327–346. Cham: Springer International Publishing, 2023.

[32] E. Broneder, C. Weiß, J. Thöndel, E. Sandner, S. Puck, M. Puck, G. F. Domínguez, and M. Sili, "Tactile– mixed reality-based system for cognitive and physical training," in *IHIET-FS*, 2022.

[33] A. Mukhopadhyay, V. K. Sharma, P. G. Tatyarao, A. K. Shah, A. M. Rao, P. R. Subin, and P. Biswas, "A comparison study between xr interfaces for driver assistance in take over request," in *Transportation Engineering*, vol. 11, 2023.

[34] M. Franzò, A. Pica, S. Pascucci, M. Serrao, F. Marinozzi, and F. Bini, "A proof of concept combined using mixed reality for personalized neurorehabilitation of cerebellar ataxic patients," in *MDPI Sensors*, vol. 23, 2023.

[35] T. M. Gregory, J. Gregory, J. Sledge, R. Allard, and O. Mir, "Surgery Guided by Mixed Reality: Presentation of a Proof of Concept," *Acta Orthopaedica*, 2018.

[36] "Lynx R1 Headset." https://www.lynx-r.com/products/lynx-r1-headset, 2021.

[37] "Apple Vision Pro." https://www.apple.com/apple-vision-pro/, 2023.

[38] "Hololens 2." https://www.microsoft.com/en-us/hololens/, 2020.

[39] A. K. Sikder, H. Aksu, and A. S. Uluagac, "6thSense: A Context-aware Sensor-based Attack Detector for Smart Devices," in *USENIX Security Symposium*, 2017.

[40] C. Wienrich, P. Komma, S. Vogt, and M. E. Latoschik, "Spatial Presence in Mixed Realities–Considerations About the Concept, Measures, Design, and Experiments," in *Frontiers in Virtual Reality*, 2021.

[41] C. Fu, Q. Zeng, and X. Du, "Hawatcher: Semantics-aware anomaly detection for appified smart homes," in *USENIX Security)*, 2021.

[42] Y. Li, Y. Cheng, W. Meng, Y. Li, and R. H. Deng, "Designing leakage-resilient password entry on head-mounted smart wearable glass devices," in *IEEE TIFS*, vol. 16, 2021.

[43] S. Jana, D. Molnar, A. Moshchuk, A. Dunn, B. Livshits, H. J. Wang, and E. Ofek, "Enabling fine-grained permissions for augmented reality applications with recognizers," in *USENIX Security*, 2013.

[44] S. Luo, X. Hu, and Z. Yan, "Hologlogger: Keystroke inference on mixed reality head mounted displays," in *IEEE VR*, 2022.

[45] Z. Ling, Z. Li, C. Chen, J. Luo, W. Yu, and X. Fu, "I know what you enter on gear vr," in *IEEE CNS*, 2019.

[46] S. Zhang, Y. Liu, and M. Gowda, "I spy you: Eavesdropping continuous speech on smartphones via motion sensors," in *ACM IMWUT*, vol. 6, 2023.

[47] M. Diamantaris, S. Moustakas, L. Sun, S. Ioannidis, and J. Polakis, "This sneaky piggy went to the android ad market: Misusing mobile sensors for stealthy data exfiltration," in *ACM CCS*, 2021.

[48] M. Vanhoef, "Fragment and Forge: Breaking Wi-Fi Through Frame Aggregation and Fragmentation," in *USENIX Security Symposium*, 2021.

[49] I. Sluganovic, M. Liskij, A. Derek, and I. Martinovic, "Tap-pair: Using spatial secrets for single-tap device pairing of augmented reality headsets," in *ACM CDASP*, 2020.

[50] Z. Li, Q. Yue, C. Sano, W. Yu, and X. Fu, "3d vision attack against authentication," in *IEEE ICC*, 2017.

[51] C. S. Yadav, J. Singh, A. Yadav, H. S. Pattanayak, R. Kumar, A. A. Khan, M. A. Haq, A. Alhussen, and S. Alharby, "Malware analysis in iot & android systems with defensive mechanism," in *Electronics*, vol. 11, MDPI, 2022.

[52] F. Giannakas, C. Troussas, A. Krouska, I. Voyiatzis, and C. Sgouropoulou, "Blending cybersecurity education with iot devices: A u-learning scenario for introducing the man-in-the-middle attack," in *Information Security Journal: A Global Perspective*, Taylor & Francis, 2022.

[53] V. Singh, P. Pant, and R. Tripathi, "Detection of frame duplication type of forgery in digital video using sub-block based features," in *ICDF2C*, 2015.

[54] R. A. Biroon, P. Pisu, and Z. Abdollahi, "Real-time False Data Injection Attack Detection in Connected Vehicle Systems with PDE Mdeling," in *American Control Conference (ACC)*, pp. 3267–3272, 2020.

[55] J. Valente and A. A. Cardenas, "Remote proofs of video freshness for public spaces," in *ACM CPS (Workshop)*, Association for Computing Machinery, 2017.

[56] G. Zhang, C. Yan, X. Ji, T. Zhang, T. Zhang, and W. Xu, "Dolphinattack: Inaudible voice commands," in *ACM CCS*, 2007.

[57] M. Li and A. Mourikis, "High-precision, Consistent EKF-based Visual–Inertial Odometry," *Journal of Robotics Research*, 2013.

[58] Y. Ling, L. Bao, Z. Jie, F. Zhu, Z. Li, S. Tang, Y. Liu, W. Liu, and T. Zhang, "Modeling Varying Camera-IMU Time Offset in Optimization-Based Visual-Inertial Odometry," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.

[59] I. Giechaskiel and K. Rasmussen, "Taxonomy and challenges of out-of-band signal injection attacks and defenses," in *IEEE Communications Surveys & Tutorials*, vol. 22, 2019.

[60] C. Bolton, S. Rampazzi, C. Li, A. Kwong, W. Xu, and K. Fu, "Blue note: How intentional acoustic interference damages availability and integrity in hard disk drives and operating systems," in *2018 IEEE S&P)*, 2018.

[61] Y. Son, H. Shin, D. Kim, Y. Park, J. Noh, K. Choi, J. Choi, and Y. Kim, "Rocking drones with intentional sound noise on gyroscopic sensors," in *USENIX Security*, 2015.

[62] J. Selvaraj, G. Y. Dayanıklı, N. P. Gaunkar, D. Ware, R. M. Gerdes, and M. Mina, "Electromagnetic induction attacks against embedded systems," in *Asia CCS*, 2018.

[63] Y. Park, Y. Son, H. Shin, D. Kim, and Y. Kim, "This ain't your dose: Sensor spoofing attack on medical infusion pump," in *USENIX Workshop on Offensive Technologies*, 2016.

[64] X. Zhao, J. Liu, X. Wu, W. Chen, F. Guo, and Z. Li, "Probabilistic Spatial Distribution Prior Based Attentional Keypoints Matching Network," *IEEE Transactions on Circuits and Systems for Video Technology (TCSVT)*, 2022.

[65] A. Kulshrestha, "On the hamming distance between base-n representations of whole numbers," *arXiv preprint arXiv:1203.4547*, 2012.

[66] Q. Hao, L. Luo, S. T. Jan, and G. Wang, "It's Not What It Looks Like: Manipulating Perceptual Hashing Based Applications," in *ACM SIGSAC Conference on Computer and Communications Security (CCS)*, 2021.

[67] P. Samanta and S. Jain, "Analysis of Perceptual Hashing Algorithms in Image Manipulation Detection," *Procedia Computer Science*, 2021.

[68] Y. Xu, Y. Wu, and H. Zhou, "Multi-Scale Voxel Hashing and Efficient 3D Representation for Mobile Augmented Reality," in *IEEE CVPR Workshops*, 2018.

[69] D. Boullosa, J. Esteve-Lanao, A. Casado, L. A. Peyré-Tartaruga, R. Gomes da Rosa, and J. Del Coso, "Factors affecting training and physical performance in recreational endurance runners," *MDPI Sports*, vol. 8, 2020.

[70] N. Kakouris, N. Yener, and D. T. Fong, "A Systematic Review of Running-related Musculoskeletal Injuries in Runners," *Journal of Sport and Health Science*, 2021.

[71] T. Schöps, T. Sattler, and M. Pollefeys, "BAD SLAM: Bundle Adjusted Direct RGB-D SLAM," in *IEEE/CVF Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[72] Stanford Artificial Intelligence Laboratory et al., "Robotic Operating System."

[73] D. Ungureanu, F. Bogo, S. Galliani, P. Sama, X. Duan, C. Meekhof, J. Stühmer, T. J. Cashman, B. Tekin, J. L. Schönberger, P. Olszta, and M. Pollefeys, "HoloLens 2 Research Mode as a Tool for Computer Vision Research," *ArXiv*, 2020.

[74] Y. Chandio, N. Bashir, and F. M. Anwar, "Holoset - a dataset for visual-inertial pose estimation in extended reality: Dataset," in *Proceedings of the 20th ACM Conference on Embedded Networked Sensor Systems*, SenSys '22, (New York, NY, USA), p. 1014–1019, Association for Computing Machinery, 2023.

[75] D. Nagothu, Y. Chen, E. Blasch, A. Aved, and S. Zhu, "Detecting Malicious False Frame Injection Attacks on Surveillance Systems at the Edge Using Electrical Network Frequency Signals," *Sensors*, 2019.

[76] F. J. Massey Jr, "The Kolmogorov-Smirnov Test for Goodness of Fit," *Journal of the American statistical Association (ASA)*, 1951.

[77] C. Gorman and L. Gustafsson, "The Use of Augmented Reality for Rehabilitation After Stroke: A Narrative Review," *Disability and Rehabilitation: Assistive Technology (DRAT)*, 2022.

[78] B. Groza and P.-S. Murvay, "Efficient Intrusion Detection With Bloom Filtering in Controller Area Networks," *IEEE Transactions on Information Forensics and Security (TIFS)*, 2019.

[79] S. Castro, R. Dean, G. Roth, G. T. Flowers, and B. Grantham, *Influence of Acoustic Noise on the Dynamic Performance of MEMS Gyroscopes*, pp. 1825–1831. ASME International Mechanical Engineering Congress and Exposition, ASME, 11 2007.