BackSnoop: Vision-based Keystroke Inference in VR via Rear-View Upper-Arm Kinematics

ANONYMOUS AUTHOR(S)

Virtual Reality (VR) technologies are increasingly employed in numerous applications across various fields. To facilitate text entry within these applications, hand-based virtual keyboard input has been widely deployed, prized for its great convenience and immersive experience. However, this interaction method introduces new security vulnerabilities. In this paper, we disclose a novel vision-based side-channel attack where a user's upper arm movement can unintentionally reveal private typing information, even when the user's input hand is completely occluded. We propose BackSnoop, a keystroke inference method that leverages rear-view video recordings to estimate typed inputs on a VR keyboard. Specifically, when a user inputs on a virtual keyboard, the physical movements of input hand naturally transfer to the upper arm through hand-forearm-arm kinetic chain. Capitalizing on the biomechanical dependency, BackSnoop reconstructs hidden hand movements from observable upper-arm motions, which subsequently serves as the basis for keystroke inference. In the experiment involving 29 participants, BackSnoop achieved a Top-20 accuracy of 43.2% for isolated word inference and a word-level Top-20 accuracy of 60.1% in the context of sentence reconstruction. In addition, extensive experiments demonstrate that BackSnoop maintains its efficacy against various real-world scenarios. Our source codes are available at https://anonymous.4open.science/r/BackSnoop.

CCS Concepts: • **Security and privacy** → **Side-channel analysis and countermeasures**; • **Human-centered computing** → *Ubiquitous and mobile computing*.

ACM Reference Format:
Anonymous Author(s). 2026. BackSnoop: Vision-based Keystroke Inference in VR via Rear-View Upper-Arm Kinematics. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 0, 0, Article 0 (May 2026), 24 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 Introduction

Virtual Reality (VR) technologies are increasingly employed in numerous applications across various fields, such as gaming [49], healthcare [38, 42], and vocational education [27, 33]. According to a recent industry report [12], the global VR market is projected to exceed $26.7 billion in 2026, driven by tens of millions of active users worldwide. To enable communication in VR, users typically rely on virtual keyboards for text input [48]. During input, the VR headset's front-facing cameras track and reconstruct the user's hand gestures to enable virtual keystrokes, as shown in Fig 1, which simulates the traditional typing experience in the digital space. This mid-air hand-based input paradigm has been widely deployed in VR, prized for its great convenience, immersive experience, and independence of additional hardware (e.g., handheld controllers).

While this direct, mid-air hand-based input method is prevalent, it still introduces security vulnerabilities. A growing body of research has demonstrated that adversaries can reconstruct a user's sensitive keystrokes by exploiting various side channels generated during VR interactions. For example, an adversary can leverage VR device's motion [18, 37, 43] and infrared sensor data [29], virtual avatar's hand gestures [39, 47] and eye movements [41] to recover sensitive user inputs. These methods impose several prerequisites for the adversary or the victim, such as presupposing that the adversary can compromise the target device or the user has actively enabled the virtual avatar. A more intuitive approach involves visually capturing the user's physical hand motions [9] for input recovery, leveraging the inherent consistency between hand movements in both virtual and physical environments. The basic premise of this attack is that the adversary maintains a direct line-of-sight to the user's input hand. This naturally inspired an interesting question: does such a physical world attack remain feasible when the input hand is entirely occluded, such as shooting from a rear-view perspective?

2026. ACM 2474-9567/2026/5-ART0
https://doi.org/10.1145/nnnnnnn.nnnnnnn

Fig. 1. Virtual interface from the victim's perspective. The user performs mid-air typing on a standard QWERTY keyboard, with a hand silhouette (bare-hand tracking) pointing to select keys.

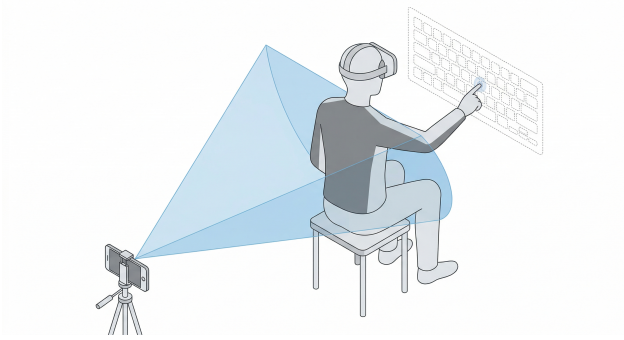


Fig. 2. Illustration of the attack scenario. The adversary uses a camera positioned behind the victim to record upper body movements, while the victim's typing hand is occluded by their torso.

In this paper, we introduce BackSnoop, a novel side-channel attack that recovers virtual keyboard inputs solely by analyzing rear-view videos of the user, as shown in Fig 2. This attack is grounded on the key insight of the inherent **kinetic chain characteristics** [14]: when a user inputs text on a virtual keyboard, the physical hand movements naturally transfer to the upper arm through hand-forearm-arm kinetic chain. Capitalizing on the biomechanical dependency, BackSnoop reconstructs hidden hand movements from observable upper arm motions, which subsequently serves as the basis for keystroke inference. This attack eliminates the need for a direct line-of-sight to the typing hands, posing a highly covert and practical threat in real-world environments. However, realizing this attack requires overcoming two primary challenges:

- **Challenge 1: Ambiguous Motion Mapping.** User's upper-arm motions typically serve only as a low-fidelity proxy for hand movements. For example, subtle shifts in the hand's position when moving between adjacent virtual keys often produce visually indistinguishable upper-arm motions, creating a severe many-to-one mapping problem. Decoding exact inputs from such ambiguous observations is challenging.
- **Challenge 2: Cross-User Variabilities.** Users exhibit significant differences in anatomical arm lengths, user-defined virtual keyboard placements, and individual typing postures. These physiological and behavioral differences drastically alter the observed upper arm kinematics. Consequently, models trained on an attacker's own data easily lose generalization capability when applied to new victims.

To resolve the many-to-one motion ambiguity (Challenge 1), BackSnoop employs a Transformer-based spatial mapper. This network extracts kinematic features from the upper-arm optical flow to model the inherent relationship between upper arm movements and hand displacements, enabling invisible hand motion reconstruction. To ensure cross-user resilience (Challenge 2), we wrap this core mapper in a multi-stage framework that systematically strips away user-specific artifacts. Specifically, the framework first normalizes user-specific sitting postures by establishing a canonical local coordinate system. It then isolates the victim's upper-arm region utilizing monocular depth and skeleton estimation, and extracts fine-grained motion features via sparse optical flow. Subsequently, the framework detects keystroke events based on motion energy analysis. Finally, a Transformer-based network maps arm optical flow to 2D keyboard hand displacements, which simultaneously employs adversarial learning to mitigate user-specific artifacts stemming from anatomical variances and individualized typing habits. These 2D hand displacements are ultimately decoded into readable text via dictionary matching.

To validate the feasibility of BackSnoop, we conducted extensive experiments involving 29 participants. We assess the system's inference capabilities for both isolated words and sentences against a huge vocabulary base with ~10,000 candidate words. Additionally, we evaluate the performance of the attack under various real-world factors, such as different camera distances, back-shooting angles, and clothing types, etc. The detailed experimental results are presented in Section 6. The main contributions of this paper include:

- We introduce BackSnoop, a novel side-channel attack that infers VR keyboard inputs from the user's upper arm kinematics. BackSnoop recovers a new Non-Line-of-Sight (NLoS) attack vector in the physical world, demonstrating that sensitive VR entry could be inferred even when the input hand is totally occluded.
- We propose an end-to-end framework that decodes the user's upper arm kinematics to text entries. By synergizing a Transformer-based reconstruction network with adversarial learning, our system achieves cross-user generalization, enabling an adversary to target new victims without requiring prior knowledge of their physiological or behavioral profiles.
- We assess the system's inference capabilities for both isolated words and sentences against a huge vocabulary base with ~10,000 candidate words. Through extensive experiments across various influencing factors, we demonstrate BackSnoop's efficacy in real-world scenarios.

4 The Keystroke Inference Framework

4.1 System Overview

Fig 4 illustrates the end-to-end pipeline of our keystroke inference framework. Given a raw back-view video recording, the system proceeds through four cascaded stages: (1) **Pre-processing** isolates the victim's body via monocular depth estimation, localizes the shoulder joints through skeleton detection, and extracts a standardized upper-arm region using an anchor-based bounding box. (2) **Motion Feature Extraction** generates a condensed motion profile by tracking sparse optical flow along the arm's outer edge contour frame by frame, yielding a time-series tensor $F \in \mathbb{R}^{T\times H\times 2}$. (3) **Typing Activity Detection** identifies the temporal boundaries of active typing sessions from the continuous feature sequence, and further segments the detected typing regions into inter-key transitions. (4) **Inference of Typed Inputs** translates the extracted motion feature into 2D spatial embedding representing inter-key displacements. It employs a user-agnostic representation learning framework to isolate typing trajectories from individual motion habits, and subsequently decodes the resulting sequences into readable text through biomechanically-constrained dictionary matching augmented with linguistic frequency priors.

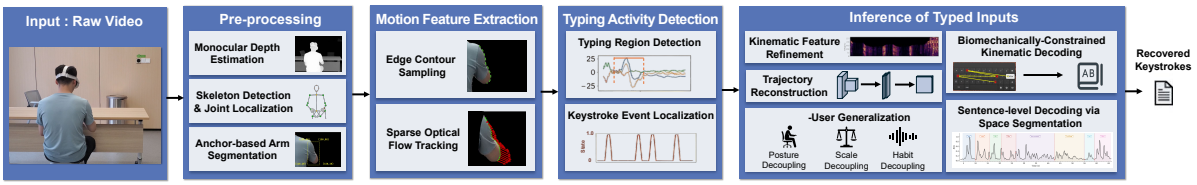
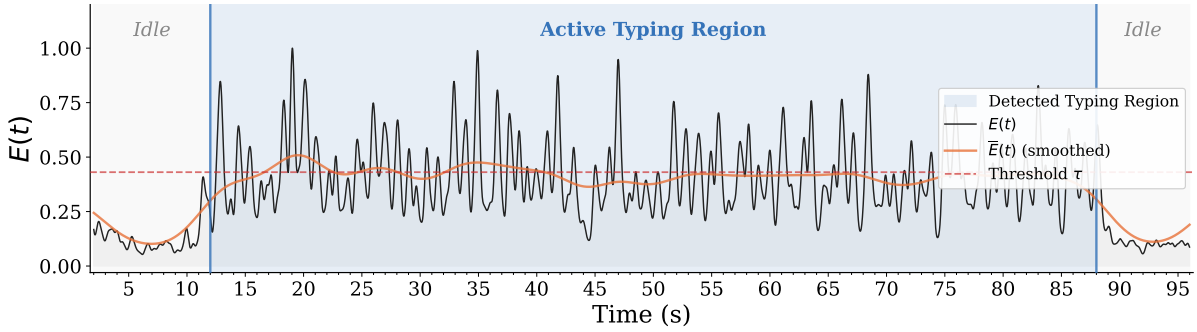
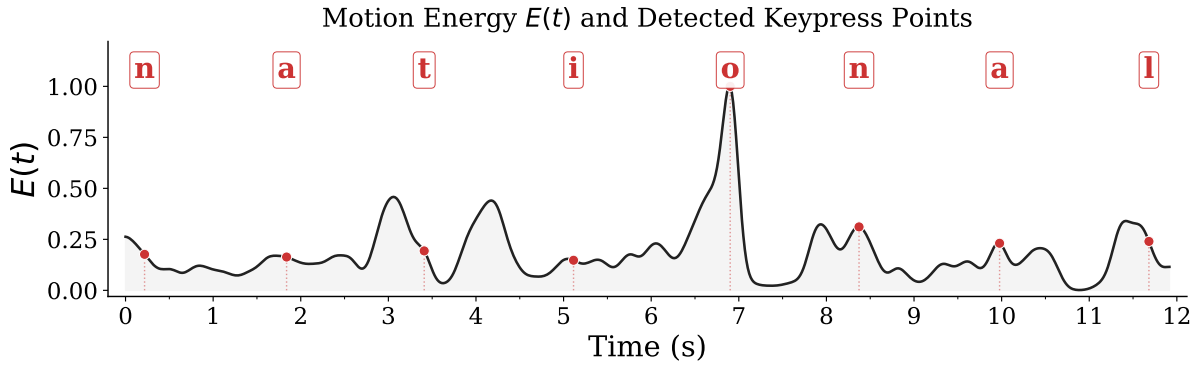


Fig. 4. End-to-end pipeline of the proposed keystroke inference framework. The system processes a back-view video through pre-processing, motion feature extraction, typing activity detection, and inference to decode the victim's typed text.



(a) Macro-level typing region detection. The raw motion energy $E(t)$ is smoothed to derive an adaptive threshold τ, distinguishing active typing sessions from idle intervals.



(b) Micro-level keystroke event localization. Within the active typing region, individual keystrokes (e.g., typing "n-a-t-i-o-n-a-l") are pinpointed by extracting the onset peaks of the energy signal.

Fig. 7. The two-level temporal activity detection scheme. (a) Isolating the continuous typing session from the entire video sequence. (b) Pinpointing the exact temporal anchors of individual keystrokes within the isolated session.

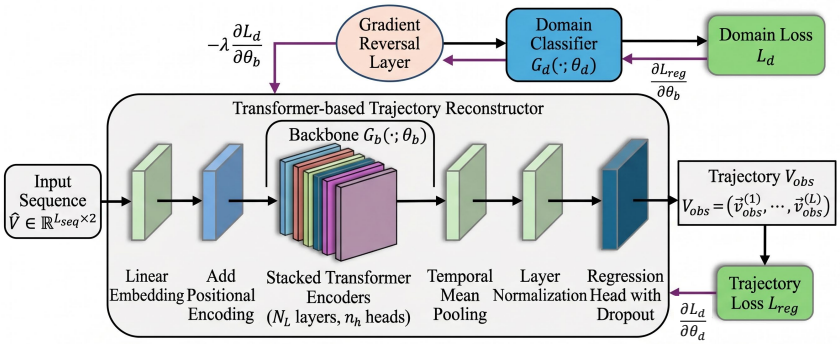
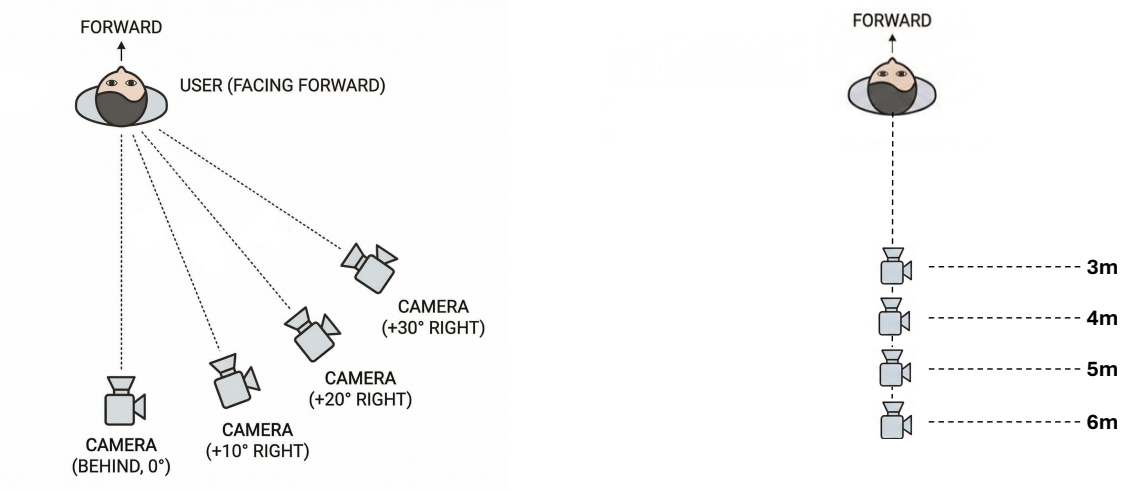


Fig. 8. Overview of the Transformer-based trajectory reconstructor. The network utilizes a stacked Transformer backbone to extract holistic kinematic representations, while a Domain Adversarial Neural Network (DANN) is integrated via a Gradient Reversal Layer (GRL) to decouple user-specific spatial variations from content-intrinsic relative motion patterns.



(a) Angle evaluation setup

(b) Distance evaluation setup

Fig. 11. Illustration of the experimental setup for evaluating BackSnoop's robustness under diverse physical viewing conditions. (a) depicts the variations in camera offset angles, and (b) depicts the variations in recording distances with corresponding optical zoom applied.

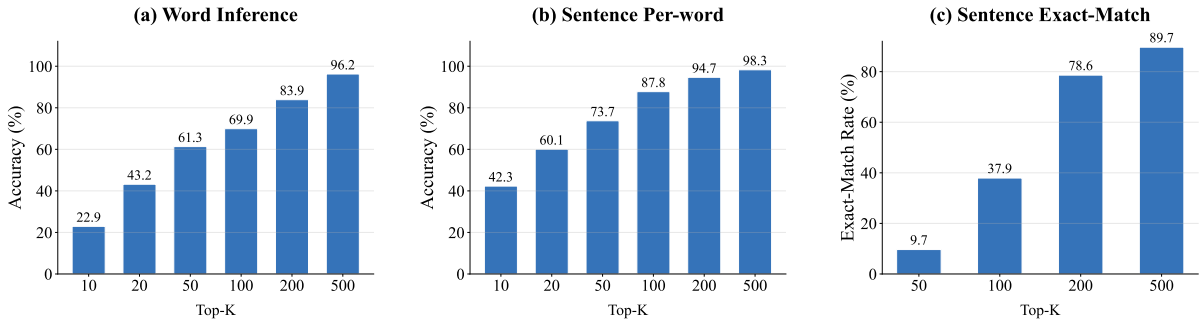


Fig. 12. Overall inference performance across 29 users. (a) Word-level Top-K accuracy (1,450 words, ~10,000-word open vocabulary). (b) Per-word Top-K accuracy within sentences (1,740 words from 290 sentence instances). (c) Sentence-level exact-match theoretical upper bound, starting from Top-50.

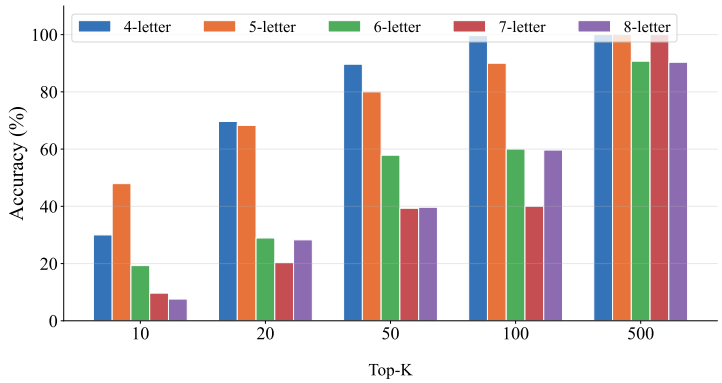


Fig. 13. Word inference Top-K accuracy stratified by word length (4–8 characters). Shorter words consistently achieve higher accuracy due to fewer cumulative transition errors and lower dictionary density.

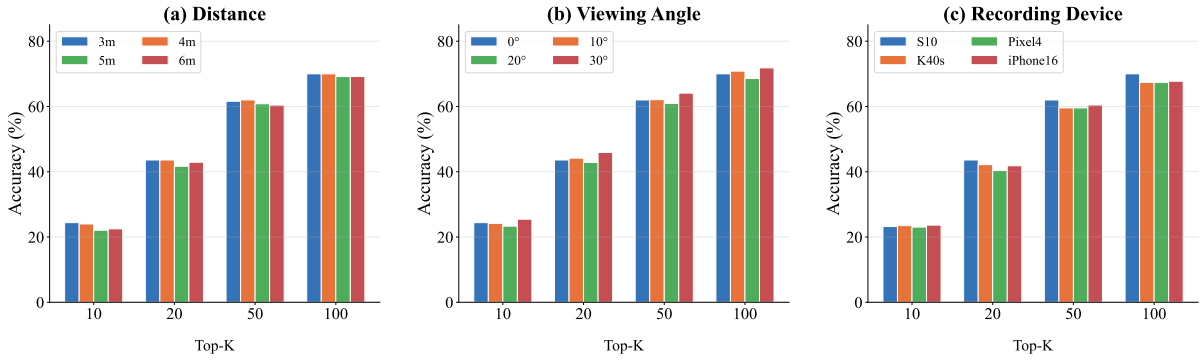


Fig. 14. Robustness evaluation across four environmental factors, each tested with 5 participants. (a) Camera distance (3–6 m with proportional zoom). (b) Camera viewing angle (0°–30°). (c) Recording device (four commodity smartphones). (d) Typing posture and handedness.

6.3 Impact of User-Specific Factors

Beyond external attack configurations, the system's robustness inherently depends on the physical and behavioral characteristics of the victims. In this section, we evaluate the influence of diverse individual variations, specifically examining the impact of typing postures, handedness, clothing types, hair lengths, and natural typing speeds.

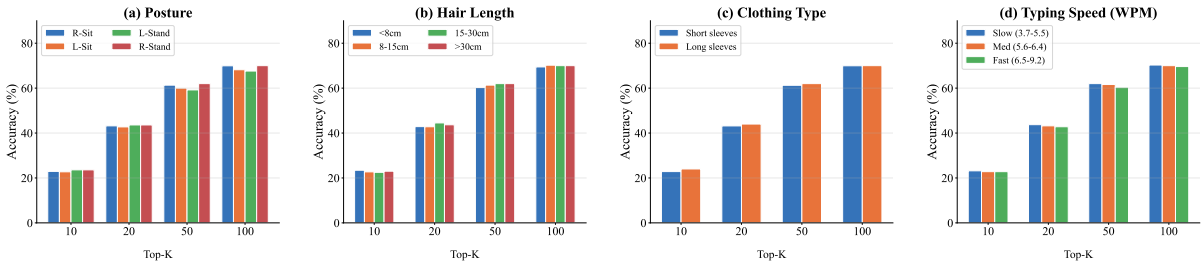


Fig. 15. Impact of user physical demographics on word inference accuracy. (a) Hair length across four groups. (b) Clothing type (long sleeve vs. coat/jacket). (c) Natural typing speed partitioned into three WPM tiers. Error bars indicate standard deviation across users within each group.