Hearing Without Noticing?

Attention-Aware Stealthy Black-Box Adversarial Audio Attacks

Tianyi Xu • Cheng'an Wei • Yue Zhao • Kai Chen

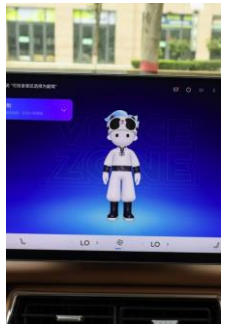
Institute of Information Engineering, University of Chinese Academy of Sciences

ICML 2026

The Problem: ASR Vulnerability & The "Stealthiness" Challenge

Ubiquitous
Integrated into smart home assistants (Google, Alexa) — deeply embedded in daily life.

Always Connected
Voice typing, virtual assistants, and voice search on billions of mobile devices.



In-Car Critical
Core to navigation, infotainment, and hands-free safety controls.

The Effectiveness-Stealthiness Trade-off
To bypass robust commercial ASR systems, attackers must use large perturbations. This often results in audio that sounds noisy, garbled, or contains clearly audible commands.

Poor User Perception (Prior Research)
- Ni-Occam Study: Only **10.78%** of users perceived the audio as "normal".
- KENKU Study: **46%** of participants could identify embedded hidden commands.

THE CORE CHALLENGE

Design attacks that are **Highly Effective** (fool ASR) AND **Highly Stealthy** (undetectable by humans)

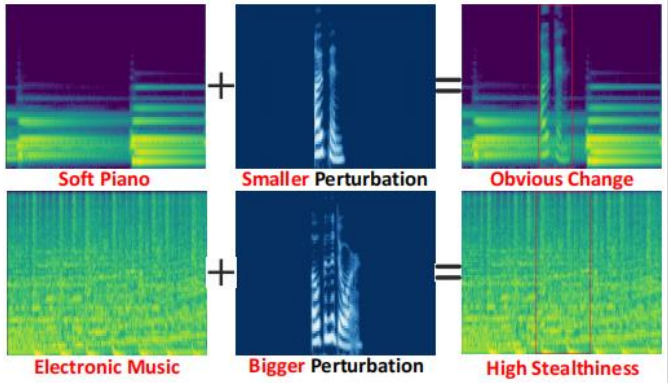
"HEARING WITHOUT NOTICING"

Our Insight: Stealthiness is More Than Just Volume

Stealthiness ≠ Low Magnitude Human auditory perception is a **selective attention process**, not just a physical measurement of sound. A sound can be physically audible ("hearing") but cognitively ignored ("without noticing") if it doesn't trigger our brain's attention mechanisms.

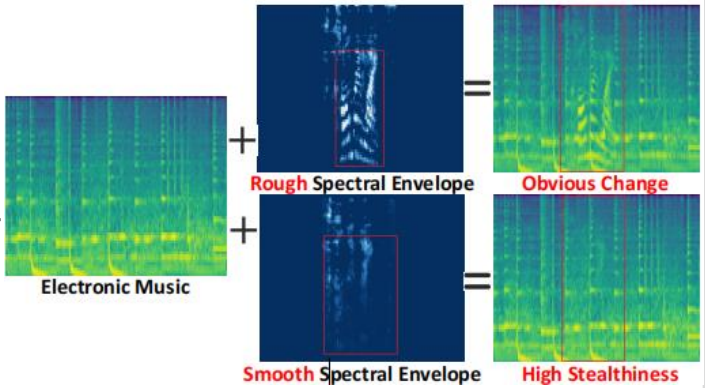
01. Acoustic Context (The Carrier)

- **Sparse carriers** (e.g., soft piano) make small perturbations very obvious.
- **Dense carriers**(e.g., electronic music) mask larger perturbations.



02. Perturbation Texture (The Noise)

- **Irregular peaks** trigger attention.
- **Smooth noise** fades into background.



Conclusion: To achieve true stealthiness, we need to **select the right carrier** and **shape the perturbation's texture** to minimize its saliency to human attention.

Our Method: HWN (Hearing Without Noticing)

Overall Architecture

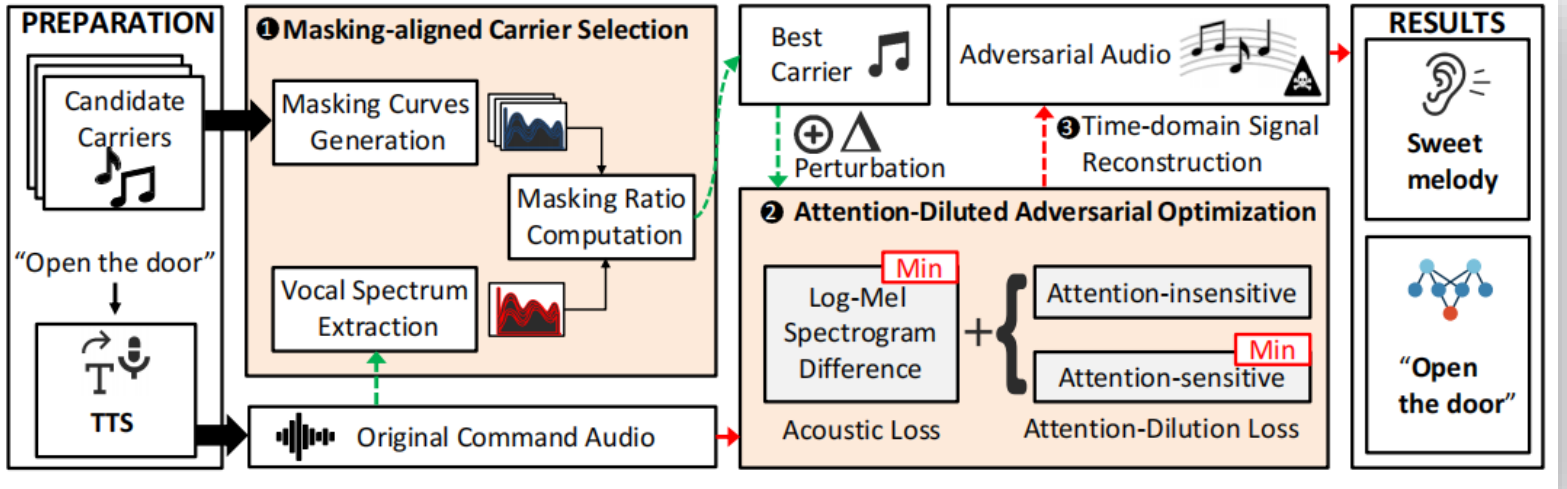


Figure 3: The HWN method combines masking-aware carrier selection and attention-diluted optimization for stealthy command injection.

Component 1: Masking-aligned Carrier Selection

GOAL: Find the music segment that can best mask the target command.

HOW: 1. Calculate masking thresholds using psychoacoustic models.

2. Align with target command by extracting spectral features.

3. Select optimal segment whose thresholds best cover the command.

Component 2: Attention-Diluted Adversarial Optimization

GOAL: Generate a perturbation that is both effective and stealthy to human ears.

HOW: A dual-component loss function:

1. **Acoustic Loss:** Ensures adversarial audio matches target command.

2. **Attention-Dilution Loss:** Penalizes sharp acoustic anomalies to minimize detection.

Key Results: Effectiveness & Stealthiness

Physical Attack (OTA)

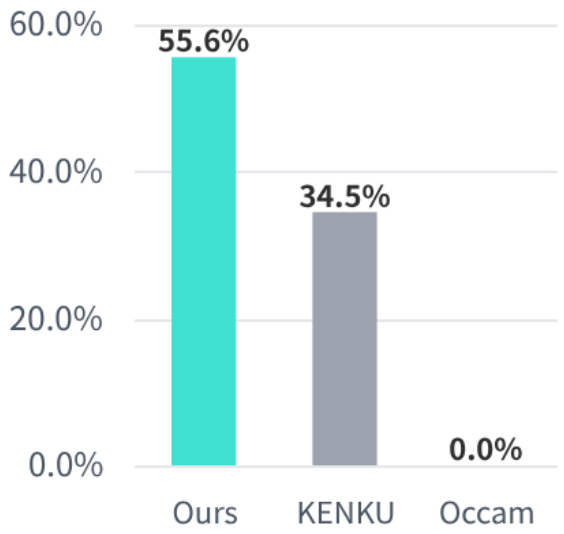
Targets: Gemini, Alexa, Doubao

Attack Success Rate

- Ours: **100% (Perfect)**
- KENKU (SOTA): 100%
- Occam (SOTA): 77.78%

Stealthiness Study

% of participants rating audio as “normal music"



Digital Domain Superiority

Similar **100% effectiveness** observed against 5 major cloud ASR APIs.

Minimal Command Leakage

Only **10.37%** detected speech, vs over **43%** baseline.

Conclusion & Future Work

Conclusion

- Proposed HWN, an attention-aware method for generating stealthy adversarial audio against black-box ASR systems.
- Achieves 100% attack success rate in both digital and physical domains.
- 55.6% of human participants could not distinguish our AEs from benign music.

"Stealthiness is not just about magnitude, but about understanding and manipulating human cognitive processes."

Future Work

- Impact: Exposes a critical vulnerability in widely deployed voice-controlled systems.
- Defenses: Need more robust strategies (e.g., multi-factor auth or AI-based anomaly detection).
- Objective Metrics: Developing better evaluation metrics for perceptual stealthiness.

Thank You!

Code and Audio Demos:`https://github.com/Spa-rkle/HWN-Attack`