IntentGaze: Task-Aligned Gaze Correction for Stable and Responsive Gaze Interaction in XR

XUNING HU*, The Hong Kong University of Science and Technology, Hong Kong
XINAN YAN*, The Hong Kong University of Science and Technology (Guangzhou), China
YICHUAN ZHANG, The Hong Kong University of Science and Technology (Guangzhou), China
YUSHI WEI, The Hong Kong University of Science and Technology (Guangzhou), China
YUE LI, Xi'an Jiaotong-Liverpool University, China
WOLFGANG STUERZLINGER, Simon Fraser University, Canada
HAI-NING LIANG†, The Hong Kong University of Science and Technology (Guangzhou), China and The Hong Kong University of Science and Technology, Hong Kong

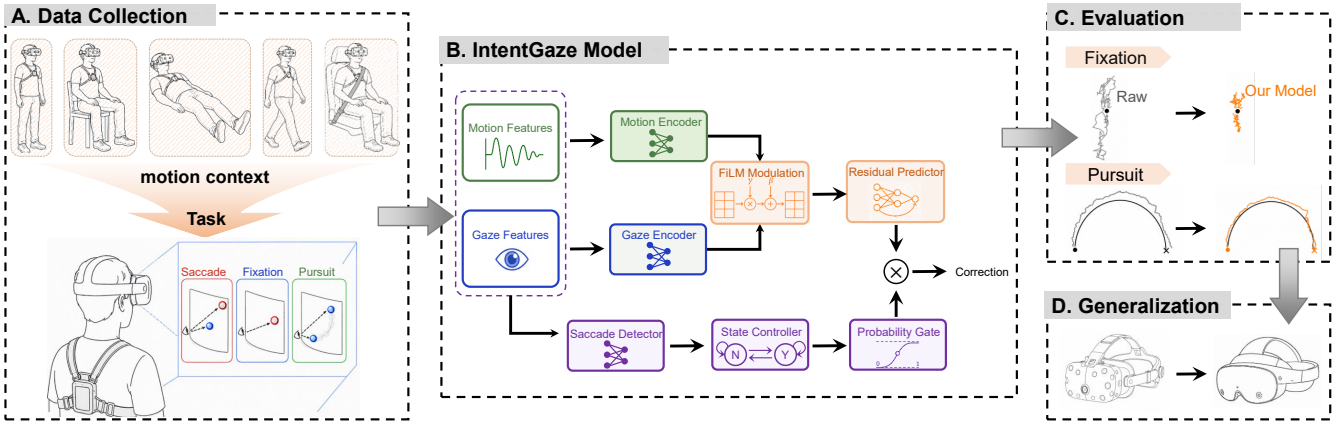

Fig. 1. Overview of IntentGaze. (A) Multi-context data collection covers different posture and motion conditions, together with fixation, pursuit, and saccade-related gaze tasks. (B) The proposed model uses gaze and motion features to estimate event-aware residual corrections through motion-conditioned modulation and saccade-aware control. (C) Our evaluation compares raw gaze and corrected gaze under fixation and pursuit tasks. (D) A generalization study evaluates the transferability of the model across XR devices.

Gaze-contingent XR systems require gaze input that is accurate, responsive, and stable under realistic head, body, and environmental motion. However, raw gaze is affected by fixation jitter, pursuit lag, saccadic transitions, and motion-induced disturbances. Filtering suppresses jitter but often trades responsiveness for stability, while gaze prediction typically estimates future measured samples for latency compensation rather than the task-aligned gaze direction needed for interaction. We formulate online XR gaze correction as causal task-aligned gaze estimation, where intended gaze is treated as the interaction task-defined direction that users are instructed to fixate on or track. To support this formulation, we designed controlled fixation, pursuit, and saccade-related tasks and collected a multi-context gaze dataset from 35 participants, comprising 6.69 million synchronized frames of measured gaze, task-defined target directions, head/body motion signals, and gaze-behavior annotations across sitting, standing, lying down, walking, and simulated vehicle movement. Based on this dataset, we introduce IntentGaze, a motion-FiLM residual correction model that predicts current-frame task-aligned gaze corrections from recent gaze and motion histories. The model uses gaze dynamics for residual correction, motion features for contextual modulation, and saccade-aware control to preserve rapid gaze transitions. In subject-independent evaluation, IntentGaze achieved mean angular errors of 0.57° for fixation and 1.30° for pursuit, reducing errors by 29% and 18% compared with the strongest baseline. It also reduced gaze instability to 0.21° and 0.88°, corresponding to reductions of 33% and 9%. Additional analyses show that motion-conditioned modulation mainly benefits motion-rich conditions, saccade-aware control reduces transition lag

*Both authors contributed equally to this research.
†Corresponding author.

Authors' Contact Information: Xuning Hu, xhuci@connect.ust.hk, The Hong Kong University of Science and Technology, Hong Kong, Hong Kong; Xinan Yan, Xinan.Yan22@student.xjtlu.edu.cn, The Hong Kong University of Science and Technology (Guangzhou), Guangzhou, China; Yichuan Zhang, Yichuan.Zhang22@student.xjtlu.edu.cn, The Hong Kong University of Science and Technology (Guangzhou), Guangzhou, China; Yushi Wei, ywei662@connect.hkust-gz.edu.cn, The Hong Kong University of Science and Technology (Guangzhou), Guangzhou, China; Yue Li, yue.li@xjtlu.edu.cn, Xi'an Jiaotong-Liverpool University, Suzhou, China; Wolfgang Stuerzlinger, w.s@sfu.ca, Simon Fraser University, Burnaby, Canada; Hai-Ning Liang, hainingliang@hkust-gz.edu.cn, The Hong Kong University of Science and Technology (Guangzhou), Guangzhou, China and The Hong Kong University of Science and Technology, Hong Kong, Hong Kong.

2026. ACM 1557-7368/2026/0-ART0
https://doi.org/10.1145/nnnnnnn.nnnnnnn

to approximately one frame, reduced-sensor cross-device evaluation preserves performance on another XR headset, and replay-based dwell selection indicates potential target-acquisition benefits. The compact model further supports online deployment without dedicated acceleration in runtime XR systems. These results suggest that causal task-aligned gaze estimation can serve as a practical runtime correction layer for stable and responsive gaze interaction in XR.

CCS Concepts: • **Human-centered computing** → *Empirical studies in HCI*; **Pointing**; **Virtual reality**; • **Computing methodologies** → *Machine learning*.

Additional Key Words and Phrases: Virtual Reality, Extended Reality, Gaze Interaction, User Studies, Interaction Techniques, Human Factors, Artificial Intelligence/Machine Learning

ACM Reference Format:
Xuning Hu, Xinan Yan, Yichuan Zhang, Yushi Wei, Yue Li, Wolfgang Stuerzlinger, and Hai-Ning Liang. 2026. IntentGaze: Task-Aligned Gaze Correction for Stable and Responsive Gaze Interaction in XR. *ACM Trans. Graph.* 0, 0, Article 0 (2026), 22 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 Introduction

Gaze has become an increasingly important input modality for extended reality (XR) systems. It enables hands-free target selection, object manipulation, and gaze-contingent rendering, where system behavior or visual fidelity is adapted according to where the user is looking [Patney et al. 2016; Pfeuffer et al. 2017; Yu et al. 2021]. In these applications, gaze is not only a perceptual measurement but also a real-time control signal. Its quality directly affects target acquisition, interface stability, perceived controllability, and the placement of high-fidelity regions in gaze-contingent rendering [Albert et al. 2017; Feit et al. 2017; Fernandes et al. 2024; Spjut et al. 2020]. Therefore, gaze-contingent XR requires gaze estimates that are accurate, stable, and responsive under realistic use conditions.

Obtaining such a reliable gaze signal is difficult in realistic XR use. Measured gaze is affected by physiological eye-movement instability, calibration drift, hardware noise, and signal processing latency [Holmqvist et al. 2012; Niehorster et al. 2020b; Stein et al. 2021]. These errors also vary across gaze behaviors. During fixation, high-frequency fluctuations can destabilize dwell-based interaction. During smooth pursuit, temporal lag can make gaze estimates fall behind moving targets. During saccades, aggressive correction can distort rapid transitions and produce misleading intermediate outputs [Kumar et al. 2008; Manakhov et al. 2024]. These challenges become more pronounced in mobile and motion-rich XR contexts, where head, body, and environmental motion further perturb the measured gaze stream [Manakhov et al. 2024]. As a result, XR gaze correction must jointly address spatial accuracy, temporal stability, and responsiveness across gaze behaviors and motion contexts.

Existing methods typically improve gaze streams through online filtering or gaze prediction [Hu et al. 2020; Špakov 2012]. Online filters, such as moving-average and One-Euro filters, are widely used to suppress high-frequency jitter in interactive systems, but they smooth the measured gaze stream rather than correct it toward a task-aligned gaze direction [Casiez et al. 2012; Špakov 2012]. As a result, stronger filtering may improve fixation stability, but it cannot explicitly compensate for systematic gaze errors and can introduce latency during pursuit or rapid gaze transitions. Gaze prediction methods address a different problem by estimating future gaze samples, often to compensate for rendering or display latency [Arabadzhiyska et al. 2017; Hu et al. 2020]. Although such prediction can anticipate where the measured gaze may move next, predicting a future gaze sample does not by itself resolve the current signal-quality problems described above, including fixation jitter, pursuit lag, saccadic distortion, and motion-induced perturbation. Thus, both filtering and prediction remain centered on the measured gaze signal itself, whereas gaze-contingent XR interaction requires a stable and responsive estimate of the gaze direction that should drive the ongoing task.

To address this mismatch, we formulate online XR gaze correction as causal task-aligned gaze estimation. In this formulation, intended gaze for explicit target-driven interaction is operationalized as the task-defined gaze direction that users are instructed to maintain or follow during an ongoing interaction. Because such intention cannot be directly observed in unconstrained use, our controlled XR tasks use the task-defined target direction as an operational proxy for intended gaze. This design provides a supervised reference for learning gaze correction while keeping inference strictly causal, since only recent gaze and motion histories are received as input. To support this formulation, we collected a multi-context XR gaze dataset from **35** participants, containing **6.69 M** synchronized frames of measured gaze, task-defined target directions, head/body motion signals, and gaze-behavior annotations across fixation, pursuit, and saccade-related transitions. Building on this dataset, we introduce IntentGaze, a motion-FiLM residual correction model that predicts current-frame gaze corrections from recent multimodal histories. The model uses gaze dynamics as the primary evidence for residual correction, motion features as contextual modulation, and saccade-aware control to preserve rapid gaze transitions.

We evaluate IntentGaze through a set of analyses that examine both signal-level quality and interaction-oriented utility (section 5). First, we compare IntentGaze with Raw Gaze and optimized online filtering baselines under a subject-independent cross-validation protocol, using spatial accuracy, gaze stability, and saccade responsiveness as primary criteria (sections 5.1.1 and 5.2). Second, we conduct ablation studies to isolate the effects of motion-conditioned FiLM modulation and saccade-aware control, examining whether the model improves gaze quality by adapting to motion context and preserving rapid gaze transitions (section 5.4). Third, we conduct a replay-based dwell-selection evaluation to examine whether frame-wise improvements translate into more reliable target acquisition (section 5.7). Finally, we test reduced-sensor cross-device robustness on an additional XR headset (section 6). Across these evaluations, IntentGaze improves fixation and pursuit accuracy/stability over raw gaze and optimized filtering baselines, reduces saccade lag to within approximately one frame, improves dwell-selection success for small targets, and shows robustness in a reduced-sensor cross-device setting.

In summary, the primary contributions of our work include:

- A formulation of online XR gaze correction as *causal task-aligned gaze estimation*, reframing gaze enhancement from smoothing or future-sample prediction into estimating a stable and responsive

gaze control signal for target-driven gaze interaction in XR (section 4.1).
- A multi-context XR gaze dataset comprising **6.69 million** synchronized frames from **35** participants, including measured gaze, target directions specified by the task, head/body motion signals, and gaze-behavior annotations across fixation, pursuit, and saccade-related transitions. Residual–motion coupling analyses further show that gaze error is conditionally related to motion in motion-rich contexts, motivating motion-aware correction under realistic XR conditions (sections 3 and 3.7).
- IntentGaze is introduced and evaluated as a motion-FiLM residual correction model that combines gaze-based residual prediction, motion-conditioned feature modulation, and saccade-aware control. Subject-independent, ablation, replay-based dwell selection, and reduced-sensor cross-device evaluations show that IntentGaze improves gaze accuracy, stability, and responsiveness over raw gaze and optimized online filtering baselines (sections 4 to 6).

To support reproducibility, the full dataset [Hu et al. 2026] and the IntentGaze reference implementation are publicly available.[1]

2 Related Work

We review prior work from three perspectives: gaze interaction in dynamic and mobile XR, behavior-dependent requirements of gaze quality, and existing approaches for improving gaze signals during interaction. Together, these areas motivate our formulation of online XR gaze correction as causal task-aligned gaze estimation.

2.1 Gaze Interaction in Dynamic and Mobile XR

Gaze is widely used in XR because it provides a fast, natural, and hands-free input channel for target selection, object manipulation, attention-aware interfaces, and gaze-contingent rendering [Chatterjee et al. 2015; Patney et al. 2016; Pfeuffer et al. 2017; Sidenmark et al. 2023; Yu et al. 2021]. At the same time, XR interaction is increasingly expected to operate beyond stationary laboratory settings, including walking, driving, and other mobility-constrained situations [Ghiurău et al. 2020; Li et al. 2025, 2024]. For gaze-based interaction, these dynamic contexts introduce additional reliability challenges: prior studies on gaze interaction under locomotion or vehicle-like movement have shown that motion can reduce selection accuracy, increase response time, and make gaze input less reliable for practical use [Colley et al. 2022; Sasalovici et al. 2025].

The degradation of gaze input in dynamic XR arises from multiple coupled sources rather than a single noise process. During self-motion, visual stability is supported by physiological mechanisms such as the vestibulo-ocular reflex and the optokinetic reflex [Barnes and Lawson 1989; Barnes 1979; Land and Tatler 2009]. Nevertheless, these mechanisms do not guarantee a stable signal for an eye-tracking system. Headset slippage, calibration drift, compensatory eye movements, hardware noise, and body-induced motion can all become mixed in the measured gaze stream [Holmqvist et al. 2022; Hooge et al. 2023; Niehorster et al. 2020a]. These factors suggest that gaze correction for XR should account for motion-varying contexts rather than assuming a stationary user or a context-invariant noise pattern.

2.2 Gaze Behavior and Behavior-Dependent Quality Requirements

Gaze behavior is commonly categorized into fixation, smooth pursuit, and saccade [Kothari et al. 2020]. These behaviors differ in their temporal dynamics and spatial stability, and therefore impose different requirements on gaze correction for XR interaction. Fixation refers to a relatively stable period during which gaze is maintained around a visual target, typically within a low velocity range of approximately 0.5–5°/s [Land and Tatler 2009]. In XR interaction, fixation supports dwell-based selection, target confirmation, and attention-aware interface adaptation [Hessels et al. 2018; Mutasim et al. 2025]. Its main quality requirement is stability around the interaction target, since high-frequency jitter can make a gaze cursor fluctuate, reduce target-acquisition reliability, and increase the risk of false activations. Smooth pursuit refers to continuous eye movement that follows a moving visual target, usually below approximately 30°/s [Barnes 2008]. It has been used for moving-target selection, pursuit-based interaction, menu navigation, and gaze-based steering tasks [Hu et al. 2025a,b; Pfeuffer et al. 2013; Vidal et al. 2013]. Unlike fixation, smooth pursuit requires both spatial alignment and temporal responsiveness. A correction method that strongly suppresses local fluctuations may appear to make the signal smoother, but can also make the estimated gaze trajectory lag behind the moving target, thereby increasing target-referenced error. Saccades are rapid ballistic eye movements that shift gaze between visual targets, typically reaching velocities of 30–500°/s and lasting approximately 20–200 ms [Holmqvist et al. 2011]. In XR interaction, target acquisition often involves a saccadic transition followed by fixation or dwell confirmation [Holmqvist et al. 2011]. During these rapid transitions, gaze correction must preserve responsiveness rather than smoothness. Applying strong smoothing or residual correction uniformly across saccades can delay the transition, distort the landing trajectory, or generate misleading intermediate gaze outputs [Kumar et al. 2008; Manakhov et al. 2024].

These behaviors rarely occur in isolation during practical interaction. Users often alternate between saccades and fixation during target acquisition, while continuous tracking may combine smooth pursuit with corrective saccades [Holmqvist et al. 2011; Hu et al. 2025b]. Therefore, gaze quality cannot be fully characterized by a single static criterion. A useful gaze signal for XR should be accurate with respect to the interaction target, stable enough to support control, and responsive enough to preserve rapid gaze transitions. This behavior-dependent view is consistent with standard eye-tracking quality measures such as spatial accuracy, spatial precision, data loss, and sampling frequency [Holmqvist et al. 2011; Niehorster et al. 2020b], but it emphasizes that these measures must be interpreted in relation to the interaction task and the current eye movement state.

[1]Dataset: https://doi.org/10.5281/zenodo.21933665; source code: https://github.com/W-YXN/IntentGaze.

2.3 Improving Gaze Input for Interactive XR Systems

Prior work has improved eye-tracking quality through calibration, runtime filtering, and data-driven gaze estimation. Calibration and recalibration methods address geometric alignment between eye measurements and display coordinates. For head-worn eye trackers, gaze quality can degrade after calibration because the headset may shift during use, even due to speaking, facial expressions, or manual adjustment [Niehorster et al. 2020a]. Implicit and background calibration methods, therefore, aim to maintain alignment during interaction. For example, Hou et al. [2025] proposed a multimodal implicit calibration method for XR that refines calibration using gaze and interaction signals. Such methods are important for reducing calibration drift, but they mainly address upstream alignment rather than the runtime correction of noisy gaze streams during ongoing interaction.

Runtime filters operate directly on gaze streams and are therefore widely used in interactive systems. Common methods include moving-average filters, Kalman-style filters, and the One-Euro filter [Casiez et al. 2012; Feit et al. 2017; Koh et al. 2009]. These filters can suppress high-frequency jitter, but they expose a fundamental stability–latency trade-off: stronger smoothing can improve fixation stability, yet it can also delay smooth pursuit and saccadic transitions [Špakov 2012]. This trade-off becomes more problematic in dynamic XR, where walking and vehicle-like movement introduce context-dependent motion patterns and temporal noise characteristics [Sasalovici et al. 2025]. Saccade-aware filtering can reduce some of these artifacts by relaxing filtering around rapid transitions [Manakhov et al. 2024], but such methods still primarily smooth the measured gaze stream rather than estimating the task-aligned gaze direction needed for interaction.

Gaze prediction addresses a related but different problem. Predictive methods estimate future gaze positions or saccade landing locations, often to compensate for rendering or display latency in gaze-contingent rendering [Arabadzhiyska et al. 2017; Hu et al. 2020]. These methods can be effective when the goal is to anticipate where the gaze will move next. However, predicting a future measured gaze sample does not directly solve the current-frame signal-quality problem faced by gaze interaction, where the system needs a stable and responsive control signal during fixation, pursuit, and saccadic transitions. In addition, many data-driven gaze estimation methods operate upstream of runtime interaction by regressing gaze direction from eye images, face images, videos, or head-pose cues, with emphasis on appearance robustness, personal calibration, or cross-user generalization [Cheng et al. 2024; Fischer et al. 2018; Zhang et al. 2020]. These advances improve gaze estimation itself, but they do not directly learn how to transform a measured runtime gaze stream into an interaction-ready signal under motion-varying XR conditions.

Overall, existing work has improved gaze input by maintaining calibration, filtering noisy measurements, predicting future gaze samples, or enhancing upstream gaze estimation. However, these approaches remain centered on the measured gaze signal. They rarely learn runtime gaze correction from synchronized gaze and body-motion histories, adapt correction behavior to motion context, or explicitly preserve the different requirements of fixation, pursuit, and saccadic transitions. This motivates our formulation of online XR gaze correction as causal task-aligned gaze estimation, where the goal is to produce a gaze control signal that is accurate, stable, and responsive for ongoing gaze-contingent interaction.

3 Data Collection and Processing

We conducted a within-subjects study to collect synchronized gaze and motion data for training and evaluating causal task-aligned gaze correction. The study included 35 participants, seven movement conditions derived from five movement contexts, and two interaction tasks: Saccade–Fixation and Pursuit. These tasks were designed to elicit fixation, smooth pursuit, and saccade-related transitions in response to explicit visual targets. In total, the study produced 36,750 trials and 6,699,749 synchronized frames of gaze, task-defined target-direction, and head/body motion data. The study protocol was approved by the local Institutional Review Board.

3.1 Participants, Apparatus, and Recorded Signals

Thirty-five participants were recruited from a local university (21 men and 14 women; age range 19–42 years, $M = 25.09$, $SD = 4.47$). All participants had normal or corrected-to-normal vision.

The experiment was implemented in Unity *2022.3.12f1* and run on a PC equipped with an AMD Ryzen 7 processor and an NVIDIA RTX 5070 GPU. Participants wore an HTC Vive Pro Eye headset, connected through the SteamVR runtime. Binocular gaze data were recorded through the SRanipal SDK callback *WrapperRegisterEyeDataCallback* with a nominal sampling rate of 90 Hz. To preserve minimally processed gaze measurements, the SRanipal parameter `sensitive_factor` was set to 1, where 0 indicates stronger SDK-level filtering and 1 indicates no filtering[2].

The SDK provided a combined binocular gaze direction, represented as a 3D vector originating from the midpoint between the two eyes. This binocular gaze direction was used as the measured gaze signal for subsequent modeling and evaluation. During the experiment, participants saw a semi-transparent gaze cursor rendered at the intersection between the gaze ray and a virtual plane placed at a depth of 1 m. The cursor subtended approximately 0.5° of visual angle and was used only as online interaction feedback. The recorded gaze signal, model input, and evaluation metrics were based on the binocular gaze direction rather than the rendered cursor position.

Motion signals were recorded from two tracked devices: the HMD and a chest-mounted Vive Tracker 3.0. For both devices, we recorded SDK-reported linear velocity $\mathbf{v} = (v_x, v_y, v_z)$ and angular velocity $\boldsymbol{\omega} = (\omega_x, \omega_y, \omega_z)$ using *UnityEngine.XR XRNodeState*, with a nominal sampling rate of 90 Hz.[3]. The HMD signals captured head motion, while the chest-mounted Vive Tracker 3.0 captured upper-body motion through SteamVR[4]. Together, these measurements provided synchronized gaze, head-motion, and body-motion streams for motion-aware gaze correction.

[2]https://forum.htc.com/topic/9119-questions-i-have-again-after-using-eye-tracking-for-few-months [Accessed: 2025-07-11]
[3]https://docs.unity3d.com/2022.3/Documentation/ScriptReference/XR.XRNodeState.html [Accessed: 2025-07-13]
[4]https://www.vive.com/us/accessory/tracker3 [Accessed: 2025-07-13]

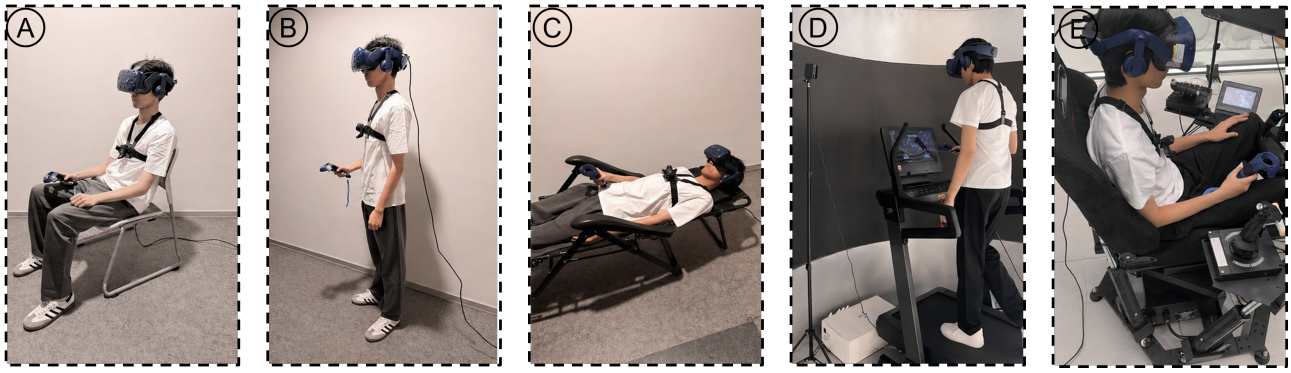

Fig. 2. Representative examples of the five movement contexts: (a) *Sitting*, (b) *Standing*, (c) *Lying Down*, (d) *Walking* (at two different speeds), and (e) *Simulated Vehicle Movement* (at two different motion intensities). Overall, this yielded a total of seven movement conditions.

3.2 Movement Conditions

To capture gaze behavior under representative XR use conditions, we included seven movement conditions grouped into five everyday contexts: *Sitting*, *Lying Down*, *Standing*, *Walking* at two speed levels, and *Simulated Vehicle Movement* at two motion-intensity levels (see fig. 2). Together, these conditions cover stable postures, self-induced locomotion, and externally induced vehicle-like motion.

Sitting. Participants sat upright on a chair with natural head movement, providing a relatively stable seated condition.

Lying Down. Participants performed the tasks in a supine position on a bed, providing a low-motion but biomechanically constrained condition.

Standing. Participants stood upright with minimal voluntary movement, serving as an upright baseline with natural postural sway.

Walking. Participants walked on a treadmill at 60% and 80% of their preferred walking speed (PWS), defined as their self-selected walking speed in daily living [Bohannon 1997]. These two levels were chosen to represent realistic locomotion during interactive XR use, ranging from a cautious interaction-oriented pace to a more comfortable exploratory gait [Bergstrom-Lehtovirta et al. 2011]. The treadmill remained flat, and participants were instructed to maintain a steady gait so that the primary disturbance arose from gait-related head and body motion.

Simulated Vehicle Movement. Participants were seated in a six-degree-of-freedom (6-DoF) motion simulator that generated vehicle-like motion at two intensity levels. The simulator was driven by a washout-based motion cueing algorithm using real-world in-vehicle inertial traces from a smartphone-sensor dataset [Khandakar et al. 2025]. Rather than manually specifying fixed translational or rotational amplitudes, we defined the two intensity levels using RMS acceleration after frequency weighting to align with ISO standards. The low-intensity condition corresponded to an RMSA of 0.58 m/s^2, while the medium-intensity condition corresponded to an RMSA of 0.83 m/s^2. According to the comfort guidance in ISO 2631-1:1997 [International Organization for Standardization 1997], weighted RMS acceleration values of 0.315–0.63 m/s^2 are associated with "a little uncomfortable", whereas values of 0.5–1.0 m/s^2 are associated with "fairly uncomfortable". Therefore, the low-intensity condition lies within the overlapping region of these two categories, while the medium-intensity condition falls within the "fairly uncomfortable" range. These settings were intended to approximate mild-to-moderate urban in-vehicle movement rather than extreme motion exposure.

3.3 Experimental Tasks

We designed two gaze-interaction tasks to elicit the eye-movement behaviors most relevant to XR gaze correction: Saccade–Fixation and Pursuit. Both tasks used explicit visual targets, allowing the task-defined target direction to serve as the supervised reference for task-aligned gaze correction. As shown in fig. 3, each trial began with the same preparation phase. Participants first aligned the gaze cursor with an initial white sphere and pressed the controller trigger to confirm readiness. An auditory cue then indicated trial onset. After this phase, the trial proceeded to either the Saccade–Fixation task or the Pursuit task.

3.3.1 Saccade–Fixation Task. This task was designed to capture target acquisition as a rapid gaze shift followed by steady fixation. In each trial, the initial white sphere appeared at a random location within a head-centered layout spanning ±25° horizontally and ±20° vertically at a visual depth of 1 m. After participants confirmed readiness, the initial sphere disappeared, and a blue target sphere appeared at a different angular location. The angular separation between the two spheres was randomly sampled from { 5°, 10°, 20°, 30°, 40° }. Participants were instructed to shift their gaze to the new target as quickly and accurately as possible. Once the gaze cursor reached the target, the sphere was highlighted, and participants maintained fixation on it for approximately 2 s. This task provided a saccadic transition followed by a controlled fixation segment.

3.3.2 Pursuit Task. This task was designed to elicit continuous smooth pursuit under explicit target motion. In each trial, the initial white sphere appeared at a depth of 1 m. After participants confirmed readiness, the target moved smoothly for 2 s along an invisible predefined trajectory. To increase movement diversity, the trajectory was randomized across trials. The trajectory shape was either circular or square, both with a visual radius of 10°. The moving direction was clockwise or counterclockwise, and the starting phase and inclination angle were randomized. The target angular

velocity was selected from { 10°/s, 15°/s, 20°/s, 25°/s, 30°/s }, covering a representative range of smooth-pursuit speeds used in prior work [Wei et al. 2025].

3.4 Procedure

Participants first watched a demonstration video explaining the study purpose, the recorded gaze and motion signals, and the two experimental tasks. They were informed that they could stop the experiment at any time if they experienced eye fatigue, cybersickness, or discomfort. After providing informed consent, participants completed a brief demographic questionnaire, wore the HMD, and completed the built-in eye-tracking calibration procedure of the HTC Vive Pro Eye. A short validation session was then conducted to confirm tracking quality and task understanding before formal data collection.

The study used a within-subjects design. Each participant completed both Saccade–Fixation and Pursuit tasks under all seven movement conditions. For each movement condition, each task contained five parameter levels, and each level was repeated 15 times, yielding 75 trials per task per movement condition. Thus, each participant completed 150 trials in each movement condition and 1,050 trials in total (2 tasks × 7 movement conditions × 75 trials), resulting in 36,750 trials across all 35 participants.

To reduce physical and visual fatigue, data collection was distributed across seven sessions, with one movement condition completed per session. The order of movement conditions was counterbalanced across participants. Within each session, task order, block order, and parameter levels were randomized. Before each block, participants received brief task instructions, followed by eye-tracker calibration and validation. Participants were allowed to rest between trials and blocks whenever needed. Each session lasted approximately 30 minutes. Participants received compensation equivalent to 35.7 USD in the local currency for completing all seven sessions.

3.5 Data Cleaning and Segmentation

We defined the supervised reference at each frame as the unit direction from the binocular gaze origin to the current task target in Unity world coordinates. This definition matched the recorded binocular gaze representation described in section 3.1, allowing target-referenced angular error to be computed in a common 3D coordinate frame. The target direction was used only for supervision, segmentation, and evaluation, but not as an input feature.

During preprocessing, 31,989 raw gaze-direction frames (0.48%) with negligible vector magnitude ($\leq 1\times10^{-5}$) were identified as signal dropouts. These frames were retained in the timeline to preserve temporal alignment, but were masked from loss computation and excluded as prediction target frames during training and evaluation. Because all tracking channels (gaze, head, and body) were recorded concurrently within a unified Unity pipeline, the data streams inherently shared a common system timeline, mitigating the need for external temporal alignment. Based on raw timestamps, the collected signals had a mean effective sampling rate of 88.17 Hz, ranging from 81.40 to 89.85 Hz across recordings. All synchronized streams were then linearly resampled to 90 Hz.

We refined gaze-behavior segments to improve label quality. Because the task-imposed maximum saccade amplitude was 40°, saccade runs longer than 27 frames (~300 ms at 90 Hz) were removed as implausibly long saccadic segments [Holmqvist et al. 2011]. This step excluded 1,773 saccade segments (11.19%), leaving 14,069 valid saccade segments. For the Saccade–Fixation task, the early portion of nominal fixation periods often contained post-saccadic settling. Based on an acceleration-inflection analysis, we masked the first 10 frames (≈ 111 ms) of each nominal fixation period, accounting for 5.51% of all fixation frames.

Finally, trial-level quality control was applied. Within each cross-validation fold, the upper IQR threshold, Q_3 + 1.5 IQR, was estimated exclusively from the target-referenced errors of the training-participant trials and applied only to the training set. Validation and test trials were not excluded based on target-referenced error and did not contribute to threshold estimation. The remaining training trials, together with the untouched validation and test trials, were used for model training, validation, and evaluation under the subject-independent protocol described in section 5.

3.6 Feature Formulation and Representation

After preprocessing, each frame was represented by an observable 18-dimensional feature vector $\mathbf{x}_t \in \mathbb{R}^{18}$, consisting of 6 gaze-related features and 12 motion-related features. The raw binocular gaze direction was converted from a 3D unit vector to yaw–pitch coordinates,

$$\mathbf{g}_t^{\text{raw}} = (\theta_t^{\text{raw}}, \phi_t^{\text{raw}})^\top.$$

The task-defined target direction was converted using the same representation,

$$\mathbf{g}_t^{*} = (\theta_t^{*}, \phi_t^{*})^\top,$$

but was used only to define the supervised residual and evaluation metrics.

The gaze-related features encoded raw-gaze kinematics:

$$\mathbf{x}_t^{\text{gaze}} = [\mathbf{g}_t^{\text{raw}}, \dot{\mathbf{g}}_t^{\text{raw}}, \ddot{\mathbf{g}}_t^{\text{raw}}] \in \mathbb{R}^6,$$

where the three terms denote yaw–pitch orientation, angular velocity, and angular acceleration, respectively. The motion-related features encoded head and upper-body motion:

$$\mathbf{x}_t^{\text{motion}} = [\mathbf{v}_{\text{head},t}, \boldsymbol{\omega}_{\text{head},t}, \mathbf{v}_{\text{chest},t}, \boldsymbol{\omega}_{\text{chest},t}] \in \mathbb{R}^{12},$$

where $\mathbf{v}_{\text{head},t}$ and $\mathbf{v}_{\text{chest},t}$ denote SDK-reported linear velocities, and $\boldsymbol{\omega}_{\text{head},t}$ and $\boldsymbol{\omega}_{\text{chest},t}$ denote SDK-reported angular velocities. The final frame-level input was defined as

$$\mathbf{x}_t = [\mathbf{x}_t^{\text{gaze}}, \mathbf{x}_t^{\text{motion}}] \in \mathbb{R}^{18}.$$

All gaze, target-direction, and motion variables were expressed in the Unity world coordinate system.

3.7 Residual–Motion Coupling Analysis and Design Implications

As an exploratory post-hoc analysis, we examined whether target-referenced gaze residuals contain systematic structure related to head and body motion. Let

$$\mathbf{r}_t = (\theta_{r,t}, \phi_{r,t})^\top = \mathbf{g}_t^{*} - \mathbf{g}_t^{\text{raw}}$$

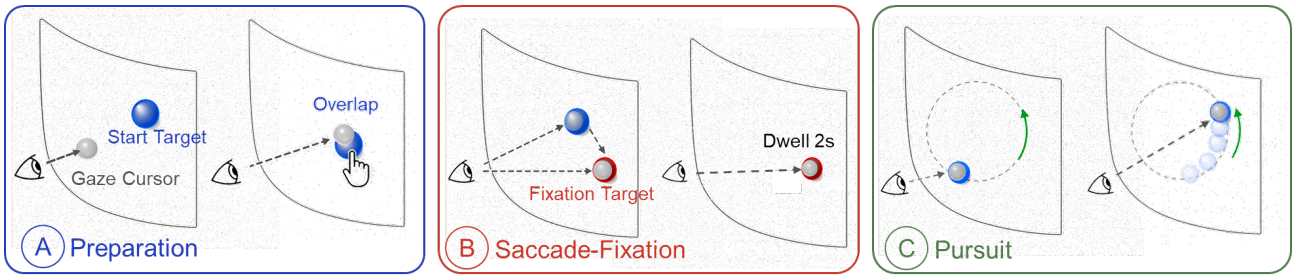

Fig. 3. Overview of the experimental tasks. (a) Shared preparation phase. Participants first aligned the gaze cursor with the initial sphere and confirmed trial onset. (b) Saccade–Fixation task. A new target appeared at a different angular location, prompting a rapid gaze shift followed by a controlled fixation period. (c) Pursuit task. The target moved continuously along a predefined trajectory, and participants tracked it with their gaze. The dashed lines and arrows indicating trajectories are for illustrative purposes only and were not visible to the participants.

denote the yaw–pitch residual between the task-defined target direction and the measured raw gaze direction. We analyzed the coupling between $\mathbf{r}_t$ and the 12 motion-related features from both time- and frequency-domain perspectives, stratified by movement context and gaze behavior. Detailed definitions of the diagnostic quantities and analysis settings are provided in appendix A.

A first observation is that gaze residuals are not fully independent of motion signals. Using normalized cross-correlation within a lag window of ±0.3 s, we found the strongest associations in the two motion-rich contexts, *Walking* and *Vehicle Movement*. These associations consistently appeared in physically interpretable direct pairs: the pitch residual ϕ_r with head angular velocity $\omega_{\text{head},x}$, and the yaw residual θ_r with head angular velocity $\omega_{\text{head},y}$. As summarized in table 1, peak correlations under *Fixation* were strong across both contexts. However, the corresponding peak lags varied across pairs and conditions, ranging from 0 to −44.4 ms.

This pattern is consistent with a motion-coupled component in the residual, while providing no support for a rigid sample-wise alignment assumption. It therefore offers exploratory motivation for allowing temporal lag and phase variability in motion-aware correction rather than imposing a universal zero-lag relationship.

Table 1. Peak cross-correlation (CC) for dominant residual–motion pairs under *Fixation*.

Movement Context	Pair	Peak CC	Lag (ms)
Simulated Vehicle Movement	$\phi_r \leftrightarrow \omega_{\text{head},x}$	0.72	−33.3
	$\theta_r \leftrightarrow \omega_{\text{head},y}$	0.53	−11.1
Walking	$\phi_r \leftrightarrow \omega_{\text{head},x}$	0.62	−44.4
	$\theta_r \leftrightarrow \omega_{\text{head},y}$	0.55	0.0

We next examined whether this coupling was also visible in the frequency domain. Using an empirical coherence threshold of $\gamma^2 > 0.3$ to mark supported frequency bins, magnitude-squared coherence analysis showed that stable residual–motion coupling was selective rather than ubiquitous. It was concentrated primarily in *Fixation*, especially under *Simulated Vehicle Movement* and *Walking*, which yielded 7 and 2 above-threshold motion–residual pairs, respectively. In contrast, pursuit conditions yielded at most 2 above-threshold pairs, and the remaining strata yielded none. A possible physiological interpretation is that fixation-related gaze stabilization depends more directly on vestibulo-ocular compensation, whereas during smooth pursuit, this coupling may be attenuated by pursuit-associated suppression of such reflex [Dietrich and Wuehr 2019; Leigh and Zee 2015].

As illustrated in fig. 4, the same direct pairs identified in the time-domain analysis also showed the strongest coherence peaks. For example, $\phi_r \leftrightarrow \omega_{\text{head},x}$ reached coherence values of 0.81 and 0.60 in *Simulated Vehicle Movement* and *Walking*, respectively, while $\theta_r \leftrightarrow \omega_{\text{head},y}$ reached 0.45 and 0.51. Notably, the dominant walking-related peaks fell within the typical adult step-frequency range [Bohannon 1997; MacDougall and Moore 2005], whereas higher peaks are more conservatively interpreted as gait-related harmonics.

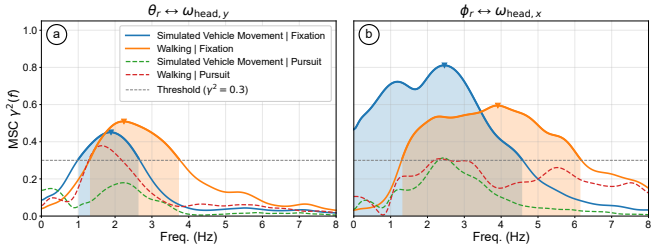

Fig. 4. Representative frequency-domain evidence of residual–motion coupling for the two dominant direct pairs. (a) Magnitude-squared coherence spectra for the yaw-related pair $\theta_r \leftrightarrow \omega_{\text{head},y}$. (b) Magnitude-squared coherence spectra for the pitch-related pair $\phi_r \leftrightarrow \omega_{\text{head},x}$. Solid lines denote *Fixation* and dashed lines denote *Pursuit* under *Simulated Vehicle Movement* and *Walking*. The horizontal dashed line indicates the empirical coherence threshold ($\gamma^2 = 0.3$). Shaded regions highlight coherence-supported frequency ranges above the threshold, and triangle markers indicate peak coherence values for the fixation curves.

The coupling was also band-limited rather than broadband. After aggregating the coherence-supported bands across the two dominant fixation conditions and excluding low-frequency drift below 0.2 Hz, the yaw-related structure was concentrated in a narrow band of 1.055–3.516 Hz, whereas the pitch-related structure occupied a

broader and more multi-channel range of approximately 0.35–6.0 Hz. Additional channels occasionally showed above-threshold coherence, but they remained secondary to the primary head angular-velocity coupling.

Taken together, these analyses suggest that residual–motion coupling is conditional on both gaze behavior and movement context. It is strongest in motion-rich fixation, appears in physically interpretable head-motion channels, varies in temporal lag, and occupies selective frequency bands. These findings motivate using motion as a contextual conditioning signal rather than as a direct residual source. Accordingly, our model encodes recent head/body motion history and uses it to modulate the gaze-based residual representation, enabling motion-adaptive correction without assuming that the full residual is uniformly motion-driven.

4 IntentGaze: Motion-FiLM Residual Gaze Correction Model

We formulate online gaze correction as causal task-aligned gaze estimation and introduce IntentGaze, a motion-FiLM residual correction model. Given a temporal history of gaze and motion observations, IntentGaze predicts a current-frame yaw–pitch correction that maps a short-term raw-gaze reference toward the task-defined gaze direction. As shown in fig. 5, the framework consists of a gaze dynamics encoder, a motion context encoder, a motion-FiLM modulation layer, and a saccade-aware correction controller. The design uses gaze dynamics as the primary basis for residual prediction, motion features for contextual modulation, and saccade-aware gating to regulate correction during rapid eye movements.

4.1 Problem Formulation

Building on the cleaned recordings described in section 3.5 and the observable multimodal representation defined in section 3.6, the model predicts a current-frame gaze correction from a causal temporal history. At time t, the input is a sliding window of observable features before t:

$$\mathbf{X}_t = [\mathbf{x}_{t-W}, \ldots, \mathbf{x}_{t-1}] \in \mathbb{R}^{W\times 18},$$

where $\mathbf{X}_t^{\text{gaze}} \in \mathbb{R}^{W\times 6}$ denotes the gaze feature window and $\mathbf{X}_t^{\text{motion}} \in \mathbb{R}^{W\times 12}$ denotes the motion feature window. This formulation ensures that inference uses only observations available before time t.

To reduce direct propagation of instantaneous high-frequency jitter, we define a short-term causal raw-gaze reference from recent raw gaze directions. The recent directions $\mathbf{g}_{t-3:t-1}^{\text{raw}}$ are converted from yaw–pitch coordinates to three-dimensional unit vectors, averaged in direction-vector space, normalized, and converted back to yaw–pitch coordinates. The resulting reference is denoted as $\bar{\mathbf{g}}_t^{\text{raw}}$.

The supervised residual target is defined as:

$$\mathbf{r}_t^* = \mathbf{g}_t^* - \bar{\mathbf{g}}_t^{\text{raw}},$$

where $\mathbf{g}_t^*$ is the task-defined target direction used as the supervised proxy for intended gaze. The target direction is used only for defining supervision and evaluation, not as an input feature. Given the causal input window, the model predicts a pre-control residual $\hat{\mathbf{r}}_t \in \mathbb{R}^2$. A saccade-aware controller then computes a continuous gate w_t and produces the corrected gaze estimate:

$$\hat{\mathbf{g}}_t^{\text{corr}} = \bar{\mathbf{g}}_t^{\text{raw}} + w_t \hat{\mathbf{r}}_t.$$

4.2 Motion-Conditioned Gaze Residual Regression Network

Motivated by the residual–motion coupling analysis in section 3.7, the residual regression network predicts the pre-control residual $\hat{\mathbf{r}}_t$ using a gaze branch and a motion-FiLM branch.

Gaze Branch. The gaze branch takes $\mathbf{X}_t^{\text{gaze}}$ as input and encodes recent gaze orientation, angular velocity, and angular acceleration. A stacked gated recurrent unit (GRU [Chung et al. 2014]) produces the temporal gaze representation:

$$\mathbf{h}_t^{\text{gaze}} = \text{GRU}\left(\mathbf{X}_t^{\text{gaze}}\right) \in \mathbb{R}^{\dim_{\text{gaze}}},$$

where $\mathbf{h}_t^{\text{gaze}}$ denotes the last-step hidden state of the stacked GRU and serves as the primary basis for residual prediction.

Motion Branch. The motion branch takes the motion feature window $\mathbf{X}_t^{\text{motion}}$ as input and encodes recent head and upper-body movement context. The motion sequence is processed by a causal temporal convolutional network (TCN [Bai et al. 2018]) to capture multi-scale temporal motion patterns:

$$\mathbf{h}_t^{\text{motion}} = \text{TCN}\left(\mathbf{X}_t^{\text{motion}}\right) \in \mathbb{R}^{\dim_{\text{motion}}},$$

The resulting motion representation is projected to generate Feature-wise Linear Modulation (FiLM [Perez et al. 2018]) parameters:

$$[\boldsymbol{\gamma}_t^{\text{motion}} \parallel \boldsymbol{\beta}_t^{\text{motion}}] = \text{MLP}_{\text{FiLM}}\left(\mathbf{h}_t^{\text{motion}}\right) \in \mathbb{R}^{2\cdot\dim_{\text{gaze}}},$$

where $\text{MLP}_{\text{FiLM}} : \mathbb{R}^{\dim_{\text{motion}}} \to \mathbb{R}^{2\cdot\dim_{\text{gaze}}}$ transforms the motion representation into a concatenated vector of modulation parameters, which is then partitioned into $\boldsymbol{\gamma}_t^{\text{motion}}, \boldsymbol{\beta}_t^{\text{motion}} \in \mathbb{R}^{\dim_{\text{gaze}}}$. Here $\boldsymbol{\gamma}_t^{\text{motion}}$ controls scaling and $\boldsymbol{\beta}_t^{\text{motion}}$ controls shifting of the gaze representation.

The modulated gaze representation is computed as:

$$\tilde{\mathbf{h}}_t^{\text{gaze}} = \text{LN}\left(\boldsymbol{\gamma}_t^{\text{motion}} \odot \mathbf{h}_t^{\text{gaze}} + \boldsymbol{\beta}_t^{\text{motion}}\right) \in \mathbb{R}^{\dim_{\text{gaze}}},$$

where $\text{LN}(\cdot)$ denotes layer normalization and $\odot$ denotes element-wise multiplication.

Residual Head. The residual head maps the modulated gaze representation to the predicted yaw–pitch residual:

$$\hat{\mathbf{r}}_t = \text{MLP}_{\text{head}}\left(\tilde{\mathbf{h}}_t^{\text{gaze}}\right) \in \mathbb{R}^2.$$

4.3 Saccade-Aware Control Module

To preserve responsiveness during rapid eye movements [Kumar et al. 2008], we use a saccade-aware control module as a terminal controller that regulates how strongly the predicted residual should be applied. This module operates independently of the residual regressor and only controls the magnitude of the residual output. IntentGaze applies learned residual correction during fixation and pursuit, while the saccade-aware controller suppresses the correction during detected saccadic transitions.

Design Rationale. The choice of a learned probabilistic detector, rather than a heuristic kinematic threshold on instantaneous gaze

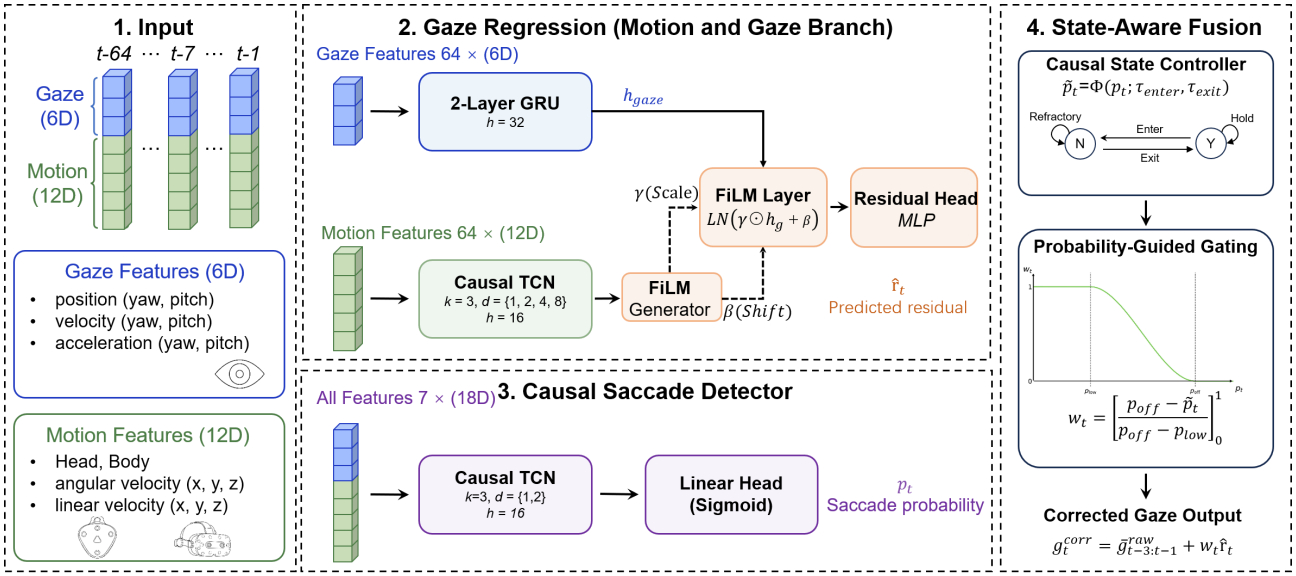

Fig. 5. The model predicts a current-frame yaw–pitch correction from recent gaze and motion histories. A GRU encodes gaze dynamics, while a causal TCN encodes head/body motion and generates FiLM parameters to modulate the gaze representation before residual prediction. A causal saccade detector and state-aware fusion module produce a continuous gate that attenuates the residual during saccadic transitions. The gated residual is then applied to the short-term raw-gaze reference to obtain the corrected gaze output.

velocity or acceleration, is dictated by two requirements of the downstream control strategy. First, the downstream gating requires a calibrated likelihood whose magnitude is informative within a transition band; a heuristic binary indicator would collapse this soft transition and bypass the refractory logic. Second, saccadic events manifest as joint temporal patterns across multiple coupled channels (orientation, angular velocity, and acceleration) rather than isolated magnitude spikes [Kothari et al. 2020; Startsev et al. 2019]. A lightweight causal network can integrate these multi-dimensional dynamics over a short temporal window, producing a stable probability for the downstream timing and gating logic.

Causal Saccade Probability Estimation. The saccade detector uses a short localized window taken from the terminal portion of the causal input history:

$$\mathbf{X}'_t = [\mathbf{x}_{t-W'}, \ldots, \mathbf{x}_{t-1}] \subset \mathbf{X}_t.$$

A lightweight TCN with dilated causal convolutions extracts a hidden representation from $\mathbf{X}'_t$. A linear classifier applied to the last temporal state produces the saccade probability:

$$p_t^{\text{sac}} = \sigma\left(\mathbf{W}_d \mathbf{h}'_t + b_d\right),$$

where $\mathbf{h}'_t$ denotes the detector hidden state and $\sigma(\cdot)$ is the sigmoid function.

State-Aware Fusion Control. To prevent rapid state toggling and false detections, the raw probability p_t^{sac} is processed by a causal state machine Φ:

$$(\tilde{p}_t, \mathbf{z}_t) = \Phi(p_t^{\text{sac}}, \mathbf{z}_{t-1}; \tau_{\text{enter}}, \tau_{\text{exit}}, d_{\text{hold}}, d_{\text{ref}}).$$

where $\mathbf{z}_t = (S_t, C_t)$ denotes the internal state, comprising a binary phase indicator $S_t \in \{0, 1\}$ and a frame counter C_t. The machine partitions the eye-movement history into three operational phases. In the **Saccadic Phase**, Φ enforces a binary high output ($\tilde{p}_t = 1$) to fully attenuate the residual during rapid gaze shifts. Upon detecting a saccade termination (stable $p_t^{\text{sac}} < \tau_{\text{exit}}$ for d_{hold} frames), the system enters a **Recovery Phase**. During this refractory period (d_{ref}), Φ passes the raw probability p_t^{sac} directly to allow for a soft, data-driven transition. Finally, the system enters the **Stable Phase** when the refractory counter exceeds d_{ref}, where a binary low output ($\tilde{p}_t = 0$) is enforced to ensure maximum transparency for gaze correction. This three-phase logic is designed to suppress post-saccadic oscillations while preserving the responsiveness of the FiLM-based correction model. Detailed algorithmic transitions are provided in appendix B.

To avoid a hard binary switch, the filtered saccade probability is converted into a continuous residual gate:

$$w_t = \left[\frac{p_{\text{off}} - \tilde{p}_t}{p_{\text{off}} - p_{\text{low}}}\right]_0^1,$$

where $[\cdot]_0^1$ denotes clamping onto the unit interval, and p_{low} and p_{off} define the lower and upper bounds of the transition band, with

$p_{\text{low}} < p_{\text{off}}$. This gate suppresses the residual correction when the saccade likelihood is high and gradually restores it as the likelihood decreases. The gated residual used by the final correction rule is:

$$\hat{\mathbf{r}}_t^{\text{out}} = w_t \hat{\mathbf{r}}_t.$$

By attenuating the residual during likely saccades and restoring it during stable gaze behavior, the controller reduces undesirable delays or distortions while preserving correction benefits during fixation and pursuit.

4.4 Training Strategy and Objectives

The residual regressor and the saccade detector are trained separately. The regressor learns the pre-control yaw–pitch residual, while the detector estimates the saccade likelihood used to control residual attenuation.

Residual Regression Objective. The residual regression network is optimized with a SmoothL1 loss:

$$\mathcal{L}_{\text{reg}} = \left\langle \text{SmoothL1}\left(s_{\text{deg}}\hat{\mathbf{r}}_t, s_{\text{deg}}\mathbf{r}_t^*\right)\right\rangle,$$

where s_{deg} converts angular residuals into degree units and $\langle\cdot\rangle$ denotes averaging over valid regression frames.

Saccade Detector Objective. The saccade detector is trained using frame-level saccade labels aligned with time t. Since saccadic frames are much less frequent than non-saccadic frames, the detector is optimized in two steps, inspired by Kang et al. [2020]. First, the full detector is trained on the natural data distribution using Binary Cross-Entropy (BCE). Second, the detector backbone is frozen, and only the final classifier is recalibrated using Focal Loss [Lin et al. 2017]. This recalibration improves sensitivity to the minority saccade class without changing the temporal feature extractor [Kang et al. 2020].

4.5 Implementation Details

Network Architecture. The residual regression network uses a causal history window of $W = 64$ frames ($\approx$ 711 ms). The gaze branch is a 2-layer GRU with hidden size 32. The motion branch is a 4-layer causal TCN with hidden size 16, kernel size 3, and dilation rates $[1, 2, 4, 8]$. The motion representation is projected into FiLM parameters before residual prediction. Dropout is set to 0.15 for both branches. The saccade detector uses a terminal window of $W' = 7$ frames ($\approx$ 78 ms) and is implemented as a 2-layer causal TCN with residual connections, hidden size 16, kernel size 3, dilation rates $[1, 2]$, and dropout 0.1.

Optimization. The proposed model is implemented using `PyTorch` and trained on a single `NVIDIA GeForce RTX 5090 GPU`. We use the `AdamW` optimizer for all training stages. To maximize data utilization, each sliding window is extracted with a stride of 1 and treated as an independent sample, followed by global frame-wise shuffling. The residual regression network is trained for 160 epochs with a batch size of 32768, an initial learning rate of 1.6×10^{-3}, and a weight decay of 2×10^{-4}. For the saccade detector, the BCE training stage runs for 80 epochs with an initial learning rate of 1×10^{-3}, and the Focal Loss recalibration stage runs for 40 epochs with a learning rate of 1×10^{-4}, adopting the default $\gamma = 2.0$ and $\alpha = 0.5$ following Lin et al. [2017]. We employ a cosine annealing scheduler with warm restarts to dynamically adjust the learning rate. Throughout the training process, no early stopping is applied; instead, the final model checkpoint is selected based on the minimum validation loss.

Hyperparameters. The causal state machine parameters (τ_{enter}, τ_{exit}, d_{hold}, d_{ref}) and the fusion-gate boundaries ($p_{\text{low}}, p_{\text{off}}$) are selected on the validation set using the operating-point selection protocol described in section 5.1.3. The selected configuration is fixed before the final test evaluation.

5 Evaluation

5.1 Evaluation Settings

We evaluated IntentGaze using a subject-independent 7-fold cross-validation protocol. The 35 participants were randomly shuffled and partitioned into seven mutually exclusive groups, each containing five participants. In each fold, one group was used as the test set, the next group was used as the validation set, and the remaining five groups were used for training. The assignment was rotated across all seven folds, ensuring that each participant appeared in the test set once and in the validation set once. This protocol prevents subject-level data leakage and evaluates whether the model generalizes to unseen users. We conducted participant-level comparisons using two-sided paired-samples (t)-tests across the 35 participants. Holm correction was applied within each task-by-metric family, and linear mixed-effects models were used as a robustness check, yielding the same significance pattern.

5.1.1 Metrics. Given the output yaw–pitch pair $\hat{\mathbf{g}}_t$ from a gaze-processing method and the task-defined target yaw–pitch pair $\mathbf{g}_t^*$ at frame t, we first convert both to 3D unit gaze vectors and compute all angular metrics in 3D space. The frame-wise target-referenced angular error is defined as

$$a_t = \arccos\left(\hat{\mathbf{g}}_t^\top \mathbf{g}_t^*\right), \tag{1}$$

where $\hat{\mathbf{g}}_t$ is the output 3D unit gaze vector and $\mathbf{g}_t^*$ is the task-defined target 3D unit vector.

Accuracy. Accuracy measures the mean angular deviation between the output gaze direction and the task-defined target direction. A lower value indicates that the gaze output is closer to the task-defined target.

$$\text{Accuracy} = \frac{1}{N}\sum_{t=1}^{N} a_t. \tag{2}$$

Stability. Stability measures the temporal variability of the target-referenced angular error. Unlike Accuracy, which reflects the mean deviation from the task-defined target direction, Stability captures how consistently the gaze output remains aligned with the target over time. A lower value indicates a more stable gaze output.

$$\text{Stability} = \sqrt{\frac{1}{N-1}\sum_{t=1}^{N}(a_t - \bar{a})^2}, \quad \bar{a} = \frac{1}{N}\sum_{t=1}^{N} a_t. \tag{3}$$

Saccade Lag. For saccadic transitions, we measured the delay introduced by each gaze-processing method using a midpoint-crossing criterion, similar to Manakhov et al. [2024], but adapted to 3D gaze vectors. For each saccade segment with raw 3D gaze start and end vectors $\mathbf{g}_s^{\text{raw}}$ and $\mathbf{g}_e^{\text{raw}}$, we define the unit saccade direction $\mathbf{u}$ and

midpoint distance m as

$$\mathbf{u} = \frac{\mathbf{g}_e^{\text{raw}} - \mathbf{g}_s^{\text{raw}}}{\|\mathbf{g}_e^{\text{raw}} - \mathbf{g}_s^{\text{raw}}\|}, \quad m = \frac{\|\mathbf{g}_e^{\text{raw}} - \mathbf{g}_s^{\text{raw}}\|}{2}. \tag{4}$$

The raw and processed gaze vectors are then projected onto $\mathbf{u}$ relative to the start position:

$$d_t = (\mathbf{g}_t^{\text{raw}} - \mathbf{g}_s^{\text{raw}})^\top \mathbf{u}, \quad \hat{d}_t = (\hat{\mathbf{g}}_t - \mathbf{g}_s^{\text{raw}})^\top \mathbf{u}. \tag{5}$$

Let t_{raw} and t_{out} denote the linearly interpolated times at which d_t and $\hat{d}_t$ first cross the midpoint m. Saccade Lag is then defined as

$$\text{Lag}_{\text{sac}} = t_{\text{out}} - t_{\text{raw}}. \tag{6}$$

5.1.2 Online Filtering Baselines. Our primary baselines are online filters, as filtering is the most widely deployed strategy for handling noisy gaze streams in real-time gaze-contingent XR systems. We do not compare against future-sample gaze prediction methods, as they address a different problem: predicting upcoming measured samples for latency compensation, rather than correcting the current stream toward a task-defined control signal. Accordingly, in our target-referenced evaluation, Raw Gaze serves as the measured-signal anchor, and online filters as the runtime correction baselines.

Raw Gaze. Raw Gaze directly uses the measured eye-tracking signal without temporal filtering or task-aligned correction. It reflects the unprocessed gaze quality under different gaze behaviors and motion contexts, and serves as the measured-signal reference for all comparisons.

Reference-only. Reference-only outputs the same three-frame causal average of raw gaze used as the reference signal in IntentGaze, but without applying the learned residual or the state-dependent control module. This baseline isolates the contribution of the short-term causal averaging operation from that of the learned task-aligned correction.

One Euro. The One Euro filter [Casiez et al. 2012] is a first-order low-pass filter with an adaptive cutoff frequency. It was originally proposed for noisy real-time interactive input and is widely used because it provides a practical trade-off between jitter reduction and responsiveness. We include it as a representative general-purpose online filtering baseline.

LP-SD*n*. We also include the low-pass filter with n-frame saccade detection, denoted as LP-SDn, proposed by Manakhov et al. [2024]. This method combines naïve first-order low-pass filtering with an n-frame saccade detection mechanism, allowing filtering to be relaxed around detected saccades to reduce delay. Because saccade-aware filtering has been shown to improve gaze stabilization under physical locomotion, LP-SDn is a particularly relevant gaze-specific online filtering baseline for our setting.

UKF-SD*n*. We further include an Unscented Kalman Filter [Wan and Van Der Merwe 2000] with n-frame saccade detection, denoted as UKF-SDn, based on the probabilistic gaze-tracking formulation of Koh et al. [2009]. The UKF estimates the latent gaze state by propagating sigma points through state-transition and observation models without explicit linearization. For a fair comparison with LP-SDn, we apply the same n-frame saccade-detection mechanism and relax filtering around detected saccades. This prevents rapid gaze transitions from disproportionately degrading the UKF and allows the comparison to focus on differences between the underlying low-pass and probabilistic filtering strategies.

5.1.3 Validation-Based Parameter Selection. To ensure a fair comparison, all hyperparameters affecting the final operating point were selected strictly on the validation set and fixed before test-set evaluation. For the filtering baselines, parameters were tuned separately for Fixation and Pursuit to select their best validation-set operating points for each task. In contrast, the hyperparameters of IntentGaze were jointly tuned across both tasks, reflecting its intended use as a unified runtime correction method.

Search Space. For IntentGaze, we grid-searched $\tau_{\text{enter}} \in [0.2, 0.9]$, $\tau_{\text{exit}} \in [0.05, 0.7]$ with $\tau_{\text{exit}} < \tau_{\text{enter}}$, and $p_{\text{low}} \in [0.0, 0.7]$, $p_{\text{off}} \in [0.3, 1.0]$ with $p_{\text{low}} < p_{\text{off}}$. The frame-duration parameters were searched over $d_{\text{hold}} \in \{0, \ldots, 20\}$ and $d_{\text{ref}} \in \{0, \ldots, 40\}$. For One Euro, we swept $f_{\text{min}} \in [0.01, 45.0]$, $\beta \in [0.0, 4.5]$, and $f_{\text{d}_{\text{cutoff}}} \in [0.5, 10.0]$. For LP-SD$n$, we swept $f_{\text{cutoff}} \in [0.01, 45.0]$, $\delta_{\text{rst}} \in [0.2, 5.0]$, and $n \in \{1, 2, 3\}$. The UKF process and measurement noise covariances were parameterized as $Q = q_{\text{UKF}} \cdot Q_0$ and $R = r_{\text{UKF}} \cdot R_0$. For UKF-SD$n$, we searched $q_{\text{UKF}} \in [0.3, 6.5]$, $r_{\text{UKF}} \in [0.2, 6.5]$, $\delta_{\text{rst}} \in [0.2, 5.0]$, and $n \in \{1, 2, 3\}$.

Selection Criteria. We applied the same operating-point selection protocol to all methods. First, to avoid corrections that introduced excessive delay, candidates with a Saccade Lag greater than 11 ms, approximately one frame at 90 Hz, were discarded. Among the remaining candidates, we identified the Pareto-optimal set with respect to Accuracy and Stability. The final operating point was selected as the configuration with the smallest Euclidean distance to the ideal point $(0, 0)$ after min–max normalization of both metrics within the feasible candidate set. The selected parameters are reported in appendix C.

5.2 Comparison with Online Baselines

As shown in fig. 6, IntentGaze achieved the best overall trade-off between Accuracy and Stability compared with the optimized online baselines. This comparison evaluates whether a task-aligned correction model provides benefits beyond conventional runtime smoothing. The two tasks expose different limitations of filtering-based correction. Fixation primarily tests whether a method can suppress local fluctuations around a stationary interaction target. Pursuit, in contrast, requires the gaze output to remain temporally aligned with a continuously moving target, making excessive smoothing more likely to introduce spatial lag.

Fixation. In Fixation, baselines improved some aspects of Raw Gaze, but their benefits were limited. One Euro slightly reduced both Accuracy and Stability errors, while LP-SDn and UKF-SDn achieved stronger reductions in Stability. Reference-only remained close to Raw Gaze, whereas UKF-SDn achieved the lowest Accuracy and Stability errors among the baseline methods. IntentGaze reduced the overall Accuracy error from 0.842° to 0.568° and the overall Stability error from 0.376° to 0.207°, corresponding to relative reductions of 32.5% and 44.9% over Raw Gaze (both Holm-adjusted ($p<.001$)). Compared with the strongest filtering baseline in each metric, IntentGaze further reduced Accuracy error by 28.7% relative to UKF-SDn and Stability error by 33.2% relative to UKF-SDn

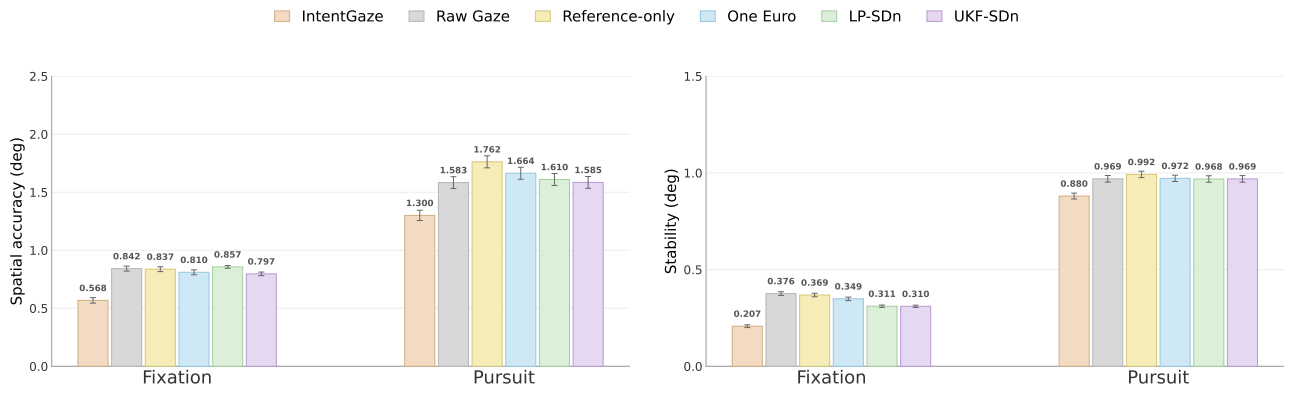

Fig. 6. Overall comparison of IntentGaze and baseline methods on the Fixation and Pursuit tasks. The left panel reports Accuracy, and the right panel reports Stability. *Lower* values indicate *better* performance. Error bars denote descriptive 95% confidence intervals across folds.

(both Holm-adjusted ($p<.001$)). These results indicate that the improvement is not simply due to stronger temporal smoothing. Instead, IntentGaze better aligns the output with the task-defined target direction while also reducing temporal instability.

Pursuit. The difference between task-aligned correction and filtering becomes more pronounced in Pursuit. In this task, the baselines did not improve spatial alignment: all baselines produced higher Accuracy errors than Raw Gaze, despite yielding nearly unchanged Stability. Reference-only produced the largest Accuracy error among the baseline methods and also slightly increased Stability error relative to Raw Gaze. This pattern is consistent with the moving-target setting, where smoothing can delay the gaze output and thereby increase target-referenced error. By contrast, IntentGaze reduced the overall Accuracy error from 1.583° to 1.300° and the overall Stability error from 0.969° to 0.880°, corresponding to reductions of 17.9% and 9.2% relative to Raw Gaze (both Holm-adjusted ($p<.001$)). Although the relative gain in Pursuit is smaller than in Fixation, IntentGaze remains the only method that improves both spatial alignment and temporal stability under continuous target motion.

Taken together, these results show that online filtering and task-aligned correction address different aspects of gaze processing. Filtering can reduce local fluctuation, especially during Fixation, but it remains constrained by the stability–latency trade-off and can harm target alignment during Pursuit. IntentGaze avoids treating gaze correction as smoothing alone. Instead, it predicts a residual correction toward the task-defined gaze direction, leading to more consistent improvements across both stationary and moving-target tasks.

5.3 Scenario-wise Analysis

Table 2 provides a scenario-wise breakdown of the same comparison. The main trend is that IntentGaze achieves the lowest Accuracy error in every scenario for both Fixation and Pursuit. This result is important because Accuracy is measured with respect to the task-defined target direction, rather than only the smoothness of the output signal. It therefore suggests that IntentGaze does not merely make the gaze trace smoother, but more consistently moves the output toward the task-defined target direction.

For Fixation, the advantage of IntentGaze is clearest in motion-rich scenarios. In Vehicle Movement and Walking, where Raw Gaze exhibits larger target-referenced errors, IntentGaze improves both Accuracy and Stability over all baselines. This supports the motivation for motion-aware task-aligned correction: when body motion or environmental motion introduces stronger gaze disturbance, predicting a task-aligned residual correction provides more effective correction than applying a generic smoothing rule.

The stable scenarios show a more nuanced pattern. In Lying Down, Sitting, and Standing, LP-SD*n* sometimes achieves slightly lower Stability than IntentGaze. This is expected because a low-pass filter with a saccade-aware reset can aggressively suppress local fluctuation when the target is stationary. However, this lower variability comes with weaker spatial alignment, whereas IntentGaze maintains the lowest Accuracy error in all stable scenarios. This distinction matters for gaze interaction: a signal can be temporally stable while still being biased away from the interaction target. IntentGaze therefore provides a more balanced correction by improving target-centered accuracy while retaining competitive stability.

For Pursuit, the scenario-wise trend is more uniform. IntentGaze outperforms all baselines in both Accuracy and Stability across all movement scenarios. In contrast, One Euro and LP-SD*n* remain close to Raw Gaze in Stability but generally increase the Accuracy error. This suggests that the main challenge in Pursuit is not only suppressing noise, but also avoiding temporal lag introduced by smoothing. Because IntentGaze predicts a task-aligned correction under the current task context, it better preserves alignment with the moving target.

5.4 Ablation Study

To examine the contribution of the main architectural components, we evaluated two reduced variants of IntentGaze: (1) *w/o Motion Branch*, which removes the motion-processing branch and FiLM modulation while relying only on gaze-derived features for residual prediction; and (2) *w/o Saccade-Aware Control*, which removes

Table 2. Performance comparison across tasks and scenarios. Each cell reports Accuracy / Stability. *Lower* values indicate *better* performance. Filtering baselines use task-specific Pareto-selected settings: fixation columns use fixation-tuned parameters and pursuit columns use pursuit-tuned parameters.

Method	Fixation					Pursuit				
	Lying Down	Vehicle Move.	Sitting	Standing	Walking	Lying Down	Vehicle Move.	Sitting	Standing	Walking
IntentGaze	**0.620** / 0.182	**0.486** / **0.198**	**0.502** / 0.150	**0.513** / 0.160	**0.712** / **0.291**	**1.427** / **0.912**	**1.109** / **0.839**	**1.305** / **0.870**	**1.333** / **0.891**	**1.456** / **0.921**
Raw Gaze	0.763 / 0.266	0.879 / 0.431	0.633 / 0.216	0.630 / 0.239	1.076 / 0.532	1.784 / 1.004	1.372 / 0.930	1.583 / 0.935	1.613 / 0.987	1.730 / 1.017
Reference-only	0.757 / 0.258	0.881 / 0.429	0.628 / 0.212	0.626 / 0.234	1.064 / 0.516	1.987 / 1.022	1.542 / 0.958	1.800 / 0.953	1.822 / 1.007	1.870 / 1.039
One Euro	0.748 / 0.247	0.847 / 0.407	0.620 / 0.202	0.615 / 0.222	1.020 / 0.484	1.885 / 1.004	1.443 / 0.938	1.696 / 0.936	1.714 / 0.987	1.781 / 1.015
LP-SDn	0.625 / **0.144**	0.873 / 0.313	0.512 / **0.118**	0.547 / **0.149**	1.296 / 0.573	1.821 / 1.002	1.394 / 0.932	1.624 / 0.934	1.649 / 0.985	1.744 / 1.013
UKF-SDn	0.685 / 0.187	0.738 / 0.319	0.566 / 0.151	0.572 / 0.173	1.160 / 0.517	1.786 / 1.003	1.373 / 0.930	1.586 / 0.935	1.616 / 0.986	1.730 / 1.016

the saccade-aware control mechanism and applies correction uniformly across fixation, pursuit, and saccadic intervals. The first variant also serves as a learned gaze-only correction model under the same saccade-aware control strategy, allowing us to examine whether motion-conditioned modulation provides additional benefits beyond gaze-history-based learning. All variants were evaluated under the same 7-fold subject-independent protocol.

Table 3 shows that the motion branch mainly benefits motion-rich scenarios. Under Vehicle Movement, removing the motion branch increased fixation Accuracy error from 0.486° to 0.633° and Stability error from 0.198° to 0.281°. During Walking, the corresponding errors increased from 0.712° to 0.851° and from 0.291° to 0.383°, respectively. In relatively static scenarios, such as Lying Down, Sitting, and Standing, the differences were smaller. This pattern indicates that the motion branch is most useful when head/body motion or external disturbance becomes a major source of gaze degradation.

The Saccade-Aware Control module primarily decouples stabilized correction from temporal responsiveness. As shown in table 3, removing this module results in nearly identical Accuracy and Stability for fixation and pursuit, suggesting that the controller remains largely transparent during non-saccadic intervals. Its main contribution appears during rapid gaze transitions. With Saccade-Aware Control, Saccade Lag is reduced from 31.36 ms, approximately three frames at 90 Hz, to 8.35 ms, within a single frame. This improvement is supported by the causal saccade detector and state machine, which achieve a balanced accuracy of 83.38% ± 1.11%. The high non-saccadic recall of 91.40% helps ensure that residual correction remains active during stable gaze states, while the saccadic recall of 75.36% is sufficient to trigger attenuation during rapid transitions.

5.5 Held-Out Trajectory-Family Transfer

To assess reliance on trajectory-specific priors, we conducted bidirectional held-out transfer between circular and square pursuit trajectories. Circle-to-square transfer achieved comparable performance to training on both trajectory families (Accuracy: 1.347°; Stability: 0.899°), with marginal increases of only 0.008° and 0.005°, respectively. Similar results were observed for square-to-circle transfer (Accuracy: 1.323°; Stability: 0.888°), with corresponding increases of 0.016° and 0.005°. These small degradations indicate limited reliance on trajectory-family-specific patterns.

5.6 Efficiency and Real-Time Feasibility

We profiled the computational footprint of IntentGaze to assess its suitability for online inference. The full model is compact, containing only **25.4 K parameters**: 21.1 K in the regression branch and 4.3 K in the saccade detector. The per-frame arithmetic complexity is approximately **2.31 MFLOPs**, consisting of 2.25 MFLOPs for the regression path and 0.06 MFLOPs for the detector.

We benchmarked per-frame latency under streaming conditions with batch size 1 using the ONNX Runtime[5], which avoids Python-level overhead and better approximates a deployable configuration. On a laptop-class CPU, Intel Core i7-14650HX, the full step takes 0.334 ms on average, with p90 = 0.343 ms, corresponding to a throughput of roughly 3,000 FPS on this platform.

Dispatching the same graph to a discrete GPU does not reduce latency: ONNX Runtime measures 7.16 ms, with p90 = 7.51 ms, on an RTX 5070 Laptop GPU.[6] This pattern is consistent with a launch- and transfer-bound regime, where host–device synchronization dominates the actual computation for a small model. Although we do not directly benchmark on embedded ARM hardware, the combination of sub-millisecond latency on commodity x86 CPUs and a per-frame budget of approximately 2.3 MFLOPs suggests that the pipeline is suitable for real-time wearable deployment without requiring a dedicated accelerator.

5.7 Interaction-Level Evaluation: Replay-Based Dwell Selection

To examine whether signal-level improvements can translate into downstream interaction benefits, we conducted a replay-based dwell-selection evaluation using fixation segments from the Saccade–Fixation task. The evaluation followed the same 7-fold subject-independent protocol and was performed on the corresponding test sets.

In the original task, participants shifted their gaze toward a newly appeared target and then maintained fixation on it for approximately 2 s. We used this fixation period as the dwell-evaluation window for simulating dwell-based target selection. To provide sufficient temporal context for IntentGaze and the filter-based baselines, the replay segment additionally included 200 frames preceding fixation onset; these frames served as warm-up only and were excluded

[5] https://onnxruntime.ai [Accessed: 2025-10-09]

[6] The same pattern holds on server-class hardware: 0.151 ms, p90 = 0.155 ms, on an Intel Xeon Platinum 8470Q CPU versus 1.57 ms, p90 = 1.60 ms, on an RTX 5090 GPU.

Table 3. Ablation study across tasks and scenarios. Each task-scenario cell reports Accuracy / Stability, and Saccade Lag reports absolute lag in ms. *Lower* values indicate *better* performance. Scene-level values are computed as participant-averaged scenario means; Vehicle Move. and Walking aggregate their low/medium motion conditions. Shaded cells highlight the comparison between IntentGaze and the variant without the motion branch under motion-rich scenarios.

Method	Fixation					Pursuit					Saccade Lag (ms)
	Lying Down	Vehicle Move.	Sitting	Standing	Walking	Lying Down	Vehicle Move.	Sitting	Standing	Walking	
IntentGaze	0.620 / 0.182	**0.486 / 0.198**	0.502 / 0.150	0.513 / 0.160	**0.712 / 0.291**	1.427 / 0.912	**1.109 / 0.839**	1.305 / 0.870	1.333 / 0.891	**1.456 / 0.921**	8.352
w/o Motion Branch	0.656 / 0.190	0.633 / 0.281	0.545 / 0.154	0.557 / 0.172	0.851 / 0.383	1.478 / 0.916	1.163 / 0.857	1.331 / 0.872	1.347 / 0.899	1.557 / 0.959	8.268
w/o Saccade-Aware Control	0.619 / 0.178	0.486 / 0.198	0.502 / 0.149	0.512 / 0.159	0.709 / 0.286	1.423 / 0.908	1.109 / 0.839	1.306 / 0.868	1.332 / 0.890	1.451 / 0.916	31.363

from the dwell-selection evaluation. For each trial, the gaze output of each method was treated as a virtual gaze cursor, and the same dwell-selection rule was applied to Raw Gaze, One Euro, LP-SD*n*, UKF-SD*n*, and IntentGaze.

Following prior VR gaze-selection settings, the dwell threshold was set to 0.6 s [Lu et al. 2021]. We tested two target widths, $W = 1.5°$ and 2.75°, corresponding to difficult and medium selection settings [Wei et al. 2025]. The simulated gaze cursor size was set to 0.5° [Fernandes et al. 2025]. A trial was counted as successful if the gaze cursor remained within the target boundary continuously for 0.6 s during the 2 s fixation window. We report Selection Success Rate, defined as the proportion of trials in which a successful dwell selection was completed. This metric provides an interaction-level indication of whether a gaze-processing method produces output that is sufficiently accurate and stable for dwell-based target acquisition.

Results. As shown in fig. 7, IntentGaze achieved the highest Selection Success Rate under both target-width conditions. For the difficult 1.5° target, IntentGaze reached a success rate of 73.8%, compared with 59.2% for Raw Gaze, 68.4% for One Euro, 72.7% for LP-SD*n*, and 71.8% for UKF-SD*n*. This corresponds to an improvement of 14.6 percentage points over Raw Gaze and 5.4 percentage points over One Euro, while remaining slightly higher than the strongest smoothing baseline, LP-SD*n*, and 2.0 percentage points higher than UKF-SD*n*. For the medium 2.75° target, IntentGaze again achieved the highest success rate of 96.2%, compared with 90.7% for Raw Gaze, 95.5% for One Euro, 90.2% for LP-SD*n*, and 94.4% for UKF-SD*n*. The smaller margin in this condition reflects the easier target-acquisition requirement, where several methods including One Euro and UKF-SD*n* approach near-ceiling performance. Participant-level paired tests confirmed that the improvement of IntentGaze over Raw Gaze was significant ($p < .001$), as was the improvement over One Euro ($p = .022$). In contrast, the differences relative to LP-SD*n* ($p = .194$) and UKF-SD*n* ($p = .145$) were not statistically significant.

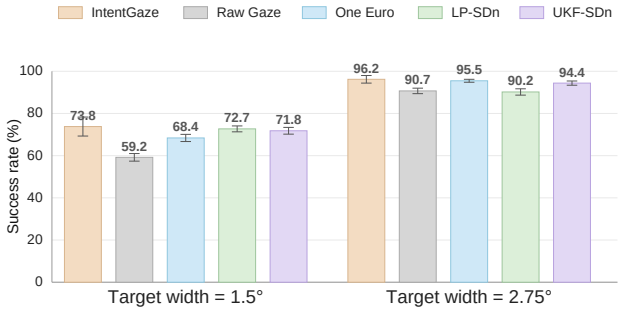

Fig. 7. Replay-based dwell-selection success rate with a 0.6 s dwell threshold. *Higher* values indicate *better* performance. Results are reported for two target widths, 1.5° and 2.75°. Bars show the mean success rate across seven folds, and error bars denote 95% confidence intervals.

Overall, the replay-based evaluation suggests that the frame-wise improvements produced by IntentGaze can improve downstream dwell-selection reliability under controlled replay conditions. The benefit is most evident for the smaller 1.5° target, where successful selection requires both accurate target alignment and sufficient temporal stability over the dwell interval. This result indicates that the proposed correction method is particularly useful under demanding gaze-selection settings, where small gaze errors or instability can prevent dwell completion.

6 Cross-Device Transfer under Reduced Sensing

6.1 Participants, Apparatus, Task, and Procedure

To further examine whether the proposed gaze correction framework can transfer across headset and sensing configurations, we conducted an additional study using a Meta Quest Pro with seven new participants (4 males, 3 females; age range 19–23, $M = 22.14$, $SD = 1.57$). The experimental system used the same Unity-based pipeline as the main data collection, but replaced the HTC Vive Pro Eye with the Meta Quest Pro. Eye tracking was obtained through the built-in eye-tracking interface provided by the Meta All-in-One SDK[7]. The `OVREyeGaze` confidence threshold was set to 0, where 0 accepts all data and 1 filters for high-confidence samples, so that

[7]https://assetstore.unity.com/packages/tools/integration/meta-xr-all-in-one-sdk-269657 [Accessed: 2025-09-22]

the recorded eye-tracking stream underwent minimal additional confidence-based filtering[8].

Unlike the main data collection, which used both head-mounted and body-worn motion sensing, this additional dataset adopted a reduced-sensor configuration using only gaze and head-motion signals from the integrated Quest Pro sensors. We did not add a body-worn motion sensor, to reflect a more broadly deployable XR setup. Both gaze and head-motion streams were recorded at a nominal sampling rate of 90 Hz within the same Unity system timeline and were linearly resampled to 90 Hz using the same preprocessing procedure as in the main dataset. The gaze representation, target definition, and target-relative processing pipeline otherwise followed the main data-collection protocol (see sections 3.1, 3.5 and 3.6). This setting allowed us to evaluate whether the proposed correction mechanism remains beneficial when both the headset and available sensing modalities change.

The additional dataset retained the same movement conditions, experimental tasks, and overall procedure as the main data collection, including the seven movement conditions and the two task types, Saccade–Fixation and Pursuit (see sections 3.2 and 3.3). In total, the dataset contained 7 participants × 2 tasks × 7 movement conditions × 75 trials, resulting in 7,350 trials. Following the same procedure as the main data collection (see section 3.4), participants completed the experiment over seven sessions, filled out a demographic questionnaire, and proceeded to the formal tasks only after the built-in eye-tracking calibration was completed and verified.

6.2 Zero-Shot Cross-Device Transfer without Body-Motion Sensing

We evaluated whether IntentGaze remains beneficial on Quest Pro under a zero-shot, reduced-sensor cross-device setting. The model trained on the main dataset was directly applied to the Quest Pro dataset without retraining, fine-tuning, or device-specific adaptation. To match the reduced-sensor configuration, the body-motion input channels were zero-masked, while the gaze and head-motion inputs were retained. The network architecture and learned parameters were otherwise kept unchanged from the main model (see section 4). This setting jointly tests two forms of generalization: transfer to a different headset and operation without body-motion sensing. We also compare against online filtering baselines under the same reduced-sensor test data. To keep the comparison adaptation-free, the filtering uses the same Fixation-optimized and Pursuit-optimized Pareto settings selected in the main dataset (appendix C).

As shown in table 4, IntentGaze consistently improved over Raw Gaze across both tasks and all scenario groups. At the event level, IntentGaze reduced Fixation Accuracy from 1.093° to 0.929° and Stability from 0.536° to 0.371°, corresponding to relative reductions of 14.94% and 30.78%, respectively. For Pursuit, it reduced Accuracy from 2.818° to 2.400° and Stability from 1.381° to 1.249°, corresponding to relative reductions of 14.83% and 9.53%.

Compared with One Euro and LP-SD*n*, IntentGaze showed a stronger overall trade-off between Accuracy and Stability across the reduced-sensor Quest Pro evaluation. The advantage was especially clear for Pursuit, where both filters reduced local fluctuation but increased target-referenced Accuracy error in all scenario groups. UKF-SD*n* showed a similar pattern in Pursuit, with Accuracy errors slightly higher than Raw Gaze and Stability remaining nearly unchanged across all scenario groups; IntentGaze achieved lower values for both metrics in every case. In Fixation, LP-SD*n* achieved slightly lower errors in some low-motion scenarios such as Lying Down and Sitting, reflecting the benefit of aggressive smoothing when responsiveness demands are limited. However, its performance degraded in motion-rich scenarios, particularly under Vehicle Movement and Walking. UKF-SD*n* improved both metrics over Raw Gaze in four of the five Fixation scenario groups, but degraded under Vehicle Movement; IntentGaze nevertheless achieved lower Accuracy in all five groups and lower Stability in four of five. In contrast, IntentGaze maintained lower errors in these dynamic conditions, suggesting that learned task-aligned correction transfers better than fixed filtering when the sensing device and motion context change simultaneously. Overall, these results suggest that IntentGaze is not tightly coupled to the original headset or full sensing configuration, and can still improve gaze quality relative to both Raw Gaze and optimized online filtering baselines in a sensor-limited XR setting without additional adaptation.

7 Discussion

7.1 From Gaze Measurement to Interaction-Ready Control

Our results suggest that XR gaze correction should be understood as constructing an interaction-ready control signal, rather than merely improving an eye-tracking measurement. Raw gaze contains task-relevant eye movements, fixation jitter, pursuit delay, saccadic transitions, and motion-induced perturbations. These components have different implications for interaction: some should be preserved as intentional visual behavior, whereas others should be attenuated because they reduce controllability. This distinction can also be seen qualitatively in the representative trajectories in fig. 8. In the fixation examples, Raw Gaze fluctuates around the target with visible drift and dispersion, especially under walking and vehicle-movement conditions, whereas IntentGaze remains more tightly concentrated around the target location. In the pursuit examples, Raw Gaze shows larger deviations from the target path, while IntentGaze follows the task-defined target trajectory more closely in both the temporal plots and the XY plane.

As shown in fig. 6, IntentGaze reduced Accuracy and Stability for both Fixation and Pursuit. These results indicate that the corrected output is both less variable and more closely aligned with the task-defined target. The qualitative patterns in fig. 8 are consistent with this interpretation: the corrected trajectory is not merely smoother than Raw Gaze, but is also visibly better centered on the task-defined target location or path.

This framing also clarifies the role of motion. Head and body motion should not be treated as signals to be directly subtracted from gaze. The same local gaze fluctuation may correspond to fixation jitter, gait-coupled disturbance, motion-induced deviation, or

[8]https://developers.meta.com/horizon/documentation/unity/move-eye-tracking [Accessed: 2025-09-22]

Table 4. Zero-shot cross-device transfer under reduced sensing. Each cell reports Accuracy / Stability. *Lower values* indicate *better performance*.

Method	Fixation					Pursuit				
	Lying Down	Vehicle Move.	Sitting	Standing	Walking	Lying Down	Vehicle Move.	Sitting	Standing	Walking
IntentGaze (Zero-Shot)	0.733 / 0.198	**1.482 / 0.722**	0.681 / 0.200	**0.641** / 0.221	**1.154 / 0.530**	**2.110 / 1.076**	**2.676 / 1.399**	**2.318 / 1.190**	**2.190 / 1.174**	**2.713 / 1.412**
Raw Gaze	0.803 / 0.327	1.759 / 0.963	0.756 / 0.315	0.716 / 0.306	1.479 / 0.797	2.532 / 1.129	2.985 / 1.579	2.860 / 1.331	2.742 / 1.306	2.977 / 1.568
One Euro	0.786 / 0.292	1.829 / 0.972	0.736 / 0.278	0.697 / 0.272	1.446 / 0.758	2.705 / 1.121	3.085 / 1.587	3.024 / 1.329	2.915 / 1.300	3.108 / 1.576
LP-SD*n*	**0.691 / 0.179**	2.354 / 1.042	**0.665 / 0.177**	0.681 / **0.184**	1.556 / 0.713	2.606 / 1.122	3.031 / 1.580	2.932 / 1.326	2.817 / 1.299	3.038 / 1.569
UKF-SD*n*	0.735 / 0.225	2.066 / 0.980	0.691 / 0.212	0.664 / 0.210	1.464 / 0.697	2.541 / 1.128	2.990 / 1.578	2.868 / 1.329	2.751 / 1.305	2.984 / 1.568

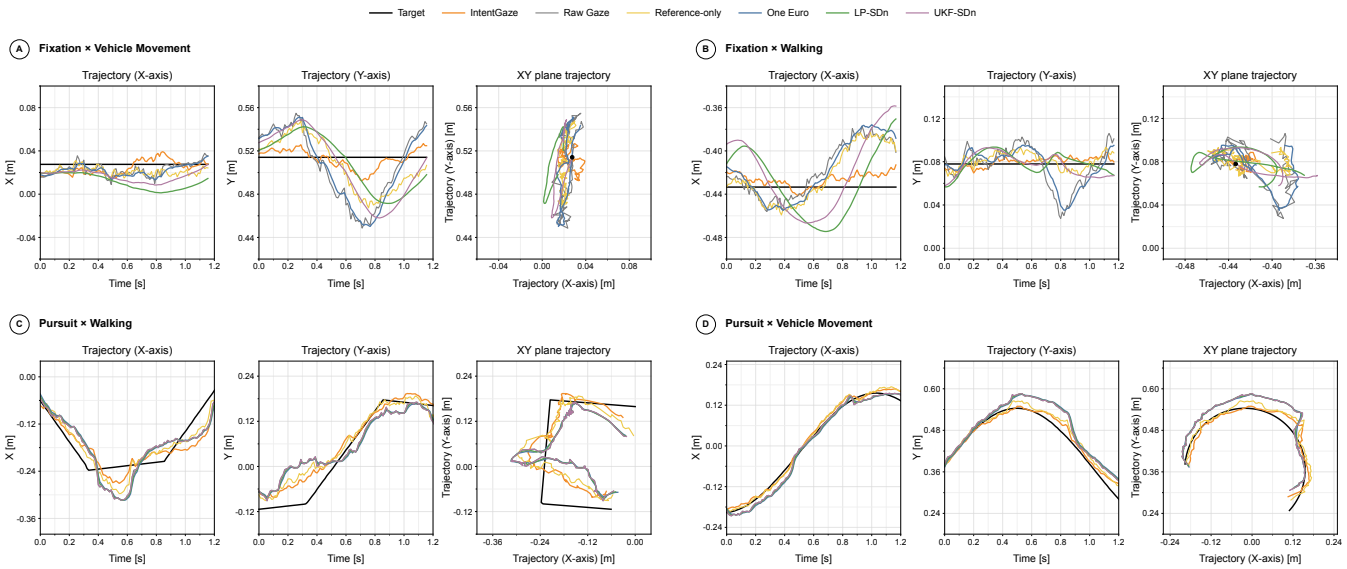

Fig. 8. Representative trajectory examples from the *test sets*, comparing gaze-processing methods under fixation and pursuit tasks in walking and vehicle-movement conditions. Each case shows the output trajectory along the X-axis, the output trajectory along the Y-axis, and the corresponding two-dimensional trajectory on the XY plane. The examples qualitatively compare the task-defined target trajectory, Raw Gaze, IntentGaze, LP-SD*n*, One Euro, Reference-only, and UKF-SD*n*.

task-relevant pursuit movement. Motion cues are therefore useful because they provide context for interpreting gaze dynamics. This interpretation is supported by the ablation results in table 3: removing the motion branch caused the clearest degradation in motion-rich scenarios, especially Vehicle Movement and Walking. For example, in Vehicle Movement, removing the motion branch increased Fixation Accuracy from 0.486° to 0.633°, and increased Stability from 0.198° to 0.281°. Figure 8 further illustrates why this context matters: under these motion-rich conditions, Raw Gaze exhibits larger path distortions and target-relative deviations, whereas IntentGaze produces a trajectory that remains closer to the task-defined target behavior.

7.2 Accuracy, Stability, and Responsiveness as Joint Requirements

The evaluation highlights that gaze-based XR control requires three coupled signal properties: spatial accuracy, temporal stability, and responsiveness. These properties are related, but not interchangeable. A stable signal may still be biased away from the target; an accurate signal may still fluctuate too much for dwell activation; and an overly stabilized signal may respond too slowly during rapid gaze transitions.

The results show why these properties need to be considered jointly. In Fixation, LP-SD*n* achieves competitive or lower Stability values in several stable scenarios, but IntentGaze maintains the lowest Accuracy error across all Fixation scenarios. This suggests that the desired output is not simply the least variable signal, but one that remains centered on the task-defined target while suppressing task-irrelevant fluctuation. This is also visible in fig. 8: in fixation, a method may look smoother but remain target-offset, whereas IntentGaze is more target-centered in the XY plane.

In Pursuit, where the task-defined target direction changes continuously, IntentGaze is the only evaluated method that improves both Accuracy and Stability, reducing the overall values to 1.300° and 0.880°, respectively. This indicates that the model can improve temporal consistency without introducing excessive lag behind the moving target. The pursuit examples in fig. 8 provide a qualitative illustration of this property: compared with Raw Gaze, IntentGaze

better follows the shape of the target trajectory over time and produces a two-dimensional path that stays closer to the intended motion pattern, rather than merely smoothing the signal at the cost of path fidelity.

Saccadic transitions further require temporal responsiveness. During rapid gaze shifts, a correction policy suitable for fixation or pursuit can become harmful if it continues to pull the output toward a previous state. The saccade-aware controller addresses this problem by regulating when residual correction should be applied. As reported in table 3, removing this controller produces nearly unchanged Fixation and Pursuit Accuracy, but increases Saccade Lag from 8.35 ms to 31.36 ms. This indicates that the controller primarily preserves responsiveness rather than improving spatial accuracy.

Together, these findings suggest that effective XR gaze correction should jointly optimize target alignment, signal stability, and transition responsiveness. IntentGaze addresses this requirement by separating task-aligned residual prediction from saccade-aware correction control.

7.3 Implications for Gaze-Based XR

The replay-based dwell evaluation suggests that task-aligned gaze correction may benefit gaze-based interaction when the task requires both target-centered accuracy and stable target occupancy. As shown in fig. 7, IntentGaze achieved the highest Selection Success Rate under both target widths, with a larger gain for the smaller 1.5° target. The success rate increased from 59.2% for Raw Gaze to 73.8% for IntentGaze. This pattern is consistent with the mechanics of dwell selection: the gaze cursor must enter the target region and remain inside it long enough to complete activation. When the target is small, either spatial bias or local instability can interrupt dwell accumulation and prevent selection.

Beyond dwell selection, the accuracy–stability–responsiveness trade-off is relevant to other gaze-based XR tasks, such as gaze typing, gaze-based steering, continuous target following, and gaze-contingent rendering. These applications differ in their interaction goals, but they share a common dependence on reliable gaze control. Typing emphasizes small-target discrimination, steering emphasizes continuous alignment, selection emphasizes stable target occupancy, and rendering emphasizes spatial reliability under low latency. In these cases, a gaze-processing method that only minimizes local jitter may not be sufficient, because excessive smoothing can harm temporal alignment and responsiveness. This task-aligned formulation, therefore, provides a useful perspective for designing runtime gaze correction as an interaction-facing control layer.

8 Limitations and Future Work

Several limitations point to opportunities for future work. Although our dataset covers a range of posture- and motion-varying XR conditions, the walking and vehicle-movement scenarios were collected in controlled laboratory settings using a treadmill and a 6-DoF motion simulator. This design enabled repeatable and synchronized data collection, but it cannot fully capture the complexity of natural mobility, including uneven terrain, turning, visual clutter, traffic dynamics, and irregular vehicle motion. In addition, while the dataset includes 35 participants and supports cross-subject evaluation, broader generalizability may require a larger and more diverse participant pool across age groups, visual conditions, calibration quality, motion sensitivity, and XR experience. In particular, the present study did not specifically examine older users, participants with ocular conditions, or users experiencing substantially different calibration quality, all of which may affect gaze-tracking reliability and correction performance. Future work should therefore evaluate task-aligned gaze correction with more participants and in more naturalistic walking and in-vehicle environments.

Another limitation is that our interaction-level evaluation was based on offline replay of recorded gaze data. This design allowed all methods to be compared under identical input sequences, but it does not capture how users adapt to corrected gaze feedback during real-time interaction or how end-to-end system latency and feedback dynamics affect task performance. Accordingly, the replay-based results provide evidence of potential downstream interaction benefits, but do not constitute validation of IntentGaze in live closed-loop XR interaction. Moreover, the downstream interaction evaluation focused on dwell selection, while pursuit-oriented interaction tasks, such as continuous target following or gaze steering, were not evaluated at the same level. Future work should conduct live closed-loop XR studies in explicit target-driven tasks, including dwell selection, target-guided steering, and continuous target following, to evaluate task performance, perceived controllability, user adaptation, and end-to-end interaction latency.

9 Conclusion

This work presented IntentGaze, a causal task-aligned gaze estimation framework for stable and responsive gaze interaction in XR. We formulate online gaze correction as causal task-aligned gaze estimation, where recent gaze and motion histories are used to predict a current-frame residual correction for producing an interaction-ready gaze control signal. To support this formulation, we collected a multi-context XR gaze dataset covering fixation, pursuit, and saccadic transitions across stable postures, walking, and simulated vehicle movement. Building on this dataset, IntentGaze uses gaze dynamics for residual correction, motion-derived FiLM modulation for context-aware adaptation, and saccade-aware control to preserve responsiveness during rapid gaze shifts. Across the main evaluation, ablation analysis, reduced-sensor cross-device evaluation, and replay-based interaction evaluation, the results show that task-aligned gaze correction improves spatial accuracy and temporal stability while maintaining responsiveness. These findings suggest that gaze processing can serve as an intermediate runtime layer that transforms noisy eye-tracking measurements into interaction-ready control signals. By making gaze correction behavior- and context-aware, IntentGaze provides a practical step toward more reliable gaze-based selection, steering, typing, and gaze-contingent XR applications under motion-varying conditions.

References

Rachel Albert, Anjul Patney, David Luebke, and Joohwan Kim. 2017. Latency Requirements for Foveated Rendering in Virtual Reality. *ACM Transactions on Applied Perception* 14, 4, Article 25 (2017), 25:1–25:13 pages. https://doi.org/10.1145/3127589

Elena Arabadzhiyska, Okan Tarhan Tursun, Karol Myszkowski, Hans-Peter Seidel, and Piotr Didyk. 2017. Saccade landing position prediction for gaze-contingent rendering. *ACM Trans. Graph.* 36, 4, Article 50 (July 2017), 12 pages. https://doi.org/10.1145/3072959.3073642

Shaojie Bai, J Zico Kolter, and Vladlen Koltun. 2018. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv preprint arXiv:1803.01271* (2018).

GR Barnes and JF Lawson. 1989. Head-free pursuit in the human of a visual target moving in a pseudo-random manner. *The Journal of physiology* 410, 1 (1989), 137–155.

G. R. Barnes. 1979. Vestibulo-ocular function during co-ordinated head and eye movements to acquire visual targets. *The Journal of Physiology* 287 (Feb. 1979), 127–147. https://doi.org/10.1113/jphysiol.1979.sp012650

Graham R. Barnes. 2008. Cognitive processes involved in smooth pursuit eye movements. *Brain and Cognition* 68, 3 (2008), 309–326. https://doi.org/10.1016/j.bandc.2008.08.020

Joanna Bergstrom-Lehtovirta, Antti Oulasvirta, and Stephen Brewster. 2011. The effects of walking speed on target acquisition on a touchscreen interface. In *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services* (Stockholm, Sweden) *(MobileHCI '11)*. Association for Computing Machinery, New York, NY, USA, 143–146. https://doi.org/10.1145/2037373.2037396

Richard W. Bohannon. 1997. Comfortable and maximum walking speed of adults aged 20—79 years: reference values and determinants. *Age and Ageing* 26, 1 (01 1997), 15–19. https://doi.org/10.1093/ageing/26.1.15

Géry Casiez, Nicolas Roussel, and Daniel Vogel. 2012. 1 € filter: a simple speed-based low-pass filter for noisy input in interactive systems. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Austin, Texas, USA) *(CHI '12)*. Association for Computing Machinery, New York, NY, USA, 2527–2530. https://doi.org/10.1145/2207676.2208639

Ishan Chatterjee, Robert Xiao, and Chris Harrison. 2015. Gaze+Gesture: Expressive, Precise and Targeted Free-Space Interactions. In *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction* (Seattle, Washington, USA) *(ICMI '15)*. Association for Computing Machinery, New York, NY, USA, 131–138. https://doi.org/10.1145/2818346.2820752

Yihua Cheng, Haofei Wang, Yiwei Bao, and Feng Lu. 2024. Appearance-Based Gaze Estimation With Deep Learning: A Review and Benchmark. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 46, 12 (2024), 7509–7528. https://doi.org/10.1109/TPAMI.2024.3393571

Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. 2014. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. arXiv:1412.3555 [cs.NE] https://arxiv.org/abs/1412.3555

Mark Colley, Pascal Jansen, Enrico Rukzio, and Jan Gugenheimer. 2022. SwiVR-CarSeat: Exploring Vehicle Motion Effects on Interaction Quality in Virtual Reality Automated Driving Using a Motorized Swivel Seat. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 5, 4, Article 150 (Dec. 2022), 26 pages. https://doi.org/10.1145/3494968

Haike Dietrich and Max Wuehr. 2019. Selective suppression of the vestibulo-ocular reflex during human locomotion. *Journal of neurology* 266, Suppl 1 (2019), 101–107.

Anna Maria Feit, Shane Williams, Arturo Toledo, Ann Paradiso, Harish Kulkarni, Shaun Kane, and Meredith Ringel Morris. 2017. Toward Everyday Gaze Input: Accuracy and Precision of Eye Tracking and Implications for Design. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (Denver, Colorado, USA) *(CHI '17)*. Association for Computing Machinery, New York, NY, USA, 1118–1130. https://doi.org/10.1145/3025453.3025599

Ajoy Savio Fernandes, T. Scott Murdison, Immo Schuetz, Oleg Komogortsev, and Michael J Proulx. 2024. The Effect of Degraded Eye Tracking Accuracy on Interactions in VR. In *Proceedings of the 2024 Symposium on Eye Tracking Research and Applications* (Glasgow, United Kingdom) *(ETRA '24)*. Association for Computing Machinery, New York, NY, USA, Article 63, 7 pages. https://doi.org/10.1145/3649902.3656369

Ajoy S. Fernandes, Immo Schütz, T. Scott Murdison, and Michael J. Proulx. 2025. Gaze Inputs for Targeting: The Eyes Have It, Not With a Cursor. *International Journal of Human-Computer Interaction* 41, 19 (2025), 12251–12269. https://doi.org/10.1080/10447318.2025.2453966

Tobias Fischer, Hyung Jin Chang, and Yiannis Demiris. 2018. RT-GENE: Real-Time Eye Gaze Estimation in Natural Environments. In *Computer Vision – ECCV 2018*, Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss (Eds.). Springer International Publishing, Cham, 339–357.

Florin-Timotei Ghiurău, Mehmet Aydın Baytaş, and Casper Wickman. 2020. ARCAR: On-Road Driving in Mixed Reality by Volvo Cars. In *Adjunct Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology* (Virtual Event, USA) *(UIST '20 Adjunct)*. Association for Computing Machinery, New York, NY, USA, 62–64. https://doi.org/10.1145/3379350.3416186

Roy S Hessels, Jelte S Benjamins, Thérèse H W Cornelissen, and Ignace T C Hooge. 2018. A Validation of Automatically-Generated Areas-of-Interest in Videos of a Face for Eye-Tracking Research. *Frontiers in Psychology* 9 (August 2018), 1367. https://doi.org/10.3389/fpsyg.2018.01367

Kenneth Holmqvist, Marcus Nyström, Richard Andersson, Richard Dewhurst, Halszka Jarodzka, and Joost Van de Weijer. 2011. *Eye tracking: A comprehensive guide to methods and measures*. oup Oxford.

Kenneth Holmqvist, Marcus Nyström, and Fiona Mulvey. 2012. Eye tracker data quality: what it is and how to measure it. In *Proceedings of the Symposium on Eye Tracking Research and Applications (ETRA '12)*. ACM, 45–52. https://doi.org/10.1145/2168556.2168563

Kenneth Holmqvist, Saga Lee Örbom, and Raimondas Zemblys. 2022. Small head movements increase and colour noise in data from five video-based P–CR eye trackers. *Behavior Research Methods* 54, 2 (2022), 845–863. https://doi.org/10.3758/s13428-021-01648-9

Ignace T. C. Hooge, Diederick C. Niehorster, Roy S. Hessels, Jeroen S. Benjamins, and Marcus Nyström. 2023. How robust are wearable eye trackers to slow and fast head and body movements? *Behavior Research Methods* 55, 8 (2023), 4128–4142. https://doi.org/10.3758/s13428-022-02010-3

Baosheng James Hou, Lucy Abramyan, Prasanthi Gurumurthy, Haley Adams, Ivana Tosic Rodgers, Eric J Gonzalez, Khushman Patel, Andrea Colaço, Ken Pfeuffer, Hans Gellersen, Karan Ahuja, and Mar Gonzalez-Franco. 2025. Online-EYE: Multimodal Implicit Eye Tracking Calibration for XR. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems (CHI '25)*. Association for Computing Machinery, New York, NY, USA, Article 550, 16 pages. https://doi.org/10.1145/3706598.3713461

Xuning Hu, Wenxuan Xu, Yushi Wei, Hao Zhang, Jin Huang, and Hai-Ning Liang. 2025a. Optimizing Moving Target Selection in VR by Integrating Proximity-Based Feedback Types and Modalities. In *2025 IEEE Conference Virtual Reality and 3D User Interfaces (VR)*. 52–62. https://doi.org/10.1109/VR59515.2025.00030

Xuning Hu, Xinan Yan, Yichuan Zhang, Yushi Wei, Yue Li, Wolfgang Stuerzlinger, and Hai-Ning Liang. 2026. *IntentGaze Dataset: Official Data for "IntentGaze: Task-Aligned Gaze Correction for Stable and Responsive Gaze Interaction in XR"*. https://doi.org/10.5281/zenodo.21933665

Xuning Hu, Yichuan Zhang, Yushi Wei, Yue Li, Wolfgang Stuerzlinger, and Hai-Ning Liang. 2025b. Exploring and Modeling Gaze-Based Steering Behavior in Virtual Reality. In *Proceedings of the Extended Abstracts of the CHI Conference on Human Factors in Computing Systems (CHI EA '25)*. Association for Computing Machinery, New York, NY, USA, Article 260, 8 pages. https://doi.org/10.1145/3706599.3720273

Zhiming Hu, Sheng Li, and Meng Gai. 2020. Temporal Continuity of Visual Attention for Future Gaze Prediction in Immersive Virtual Reality. *Virtual Reality & Intelligent Hardware* 2, 2 (2020), 142–152. https://doi.org/10.1016/j.vrih.2020.01.002

International Organization for Standardization. 1997. ISO 2631-1:1997. Mechanical vibration and shock – Evaluation of human exposure to whole-body vibration – Part 1: General requirements. https://www.iso.org/standard/7612.html. Accessed: 2025-08-05.

Bingyi Kang, Saining Xie, Marcus Rohrbach, Zhicheng Yan, Albert Gordo, Jiashi Feng, and Yannis Kalantidis. 2020. Decoupling Representation and Classifier for Long-Tailed Recognition. In *International Conference on Learning Representations*. https://openreview.net/forum?id=r1gRTCVFvB

Amith Khandakar, David G. Michelson, Mansura Naznine, Abdus Salam, Md. Nahiduzzaman, Khaled M. Khan, Ponnuthurai Nagaratnam Suganthan, Mohamed Arselene Ayari, Hamid Menouar, and Julfikar Haider. 2025. Harnessing Smartphone Sensors for Enhanced Road Safety: A Comprehensive Dataset and Review. *Scientific Data* 12, 1 (March 2025), 418. https://doi.org/10.1038/s41597-024-04193-0

Do Hyong Koh, Sandeep A. Munikrishne Gowda, and Oleg V. Komogortsev. 2009. Input evaluation of an eye-gaze-guided interface: kalman filter vs. velocity threshold eye movement identification. In *Proceedings of the 1st ACM SIGCHI Symposium on Engineering Interactive Computing Systems* (Pittsburgh, PA, USA) *(EICS '09)*. Association for Computing Machinery, New York, NY, USA, 197–202. https://doi.org/10.1145/1570433.1570470

Rakshit Kothari, Zhizhuo Yang, Christopher Kanan, Reynold Bailey, Jeff B. Pelz, and Gabriel J. Diaz. 2020. Gaze-in-wild: A dataset for studying eye and head coordination in everyday activities. *Scientific Reports* 10, 1 (2020), 2539. https://doi.org/10.1038/s41598-020-59251-5

Manu Kumar, Jeff Klingner, Rohan Puranik, Terry Winograd, and Andreas Paepcke. 2008. Improving the Accuracy of Gaze Input for Interaction. In *Proceedings of the 2008 Symposium on Eye Tracking Research & Applications (ETRA '08)*. ACM, 65–68. https://doi.org/10.1145/1344471.1344488

Michael F. Land and Benjamin W. Tatler. 2009. The Human Eye Movement Repertoire. In *Looking and Acting: Vision and Eye Movements in Natural Behaviour*. Oxford University Press, Chapter 2, 13–25. https://doi.org/10.1093/acprof:oso/9780198570943.003.0002

R. John Leigh and David S. Zee. 2015. Eye-Head Movements. In *The Neurology of Eye Movements*. Oxford University Press. https://doi.org/10.1093/med/9780199969289.003.0008

Tinghui Li, Eduardo Velloso, Anusha Withana, and Zhanna Sarsenbayeva. 2025. Estimating the Effects of Encumbrance and Walking on Mixed Reality Interaction. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems (CHI '25)*. Association for Computing Machinery, New York, NY, USA, Article 1153, 24 pages. https://doi.org/10.1145/3706598.3713492

Yang Li, Juan Liu, Jin Huang, Yang Zhang, Xiaolan Peng, Yulong Bian, and Feng Tian. 2024. Evaluating the effects of user motion and viewing mode on target selection in augmented reality. *International Journal of Human-Computer Studies* 191 (2024), 103327. https://doi.org/10.1016/j.ijhcs.2024.103327

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. 2017. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*. 2980–2988.

Xueshi Lu, Difeng Yu, Hai-Ning Liang, and Jorge Goncalves. 2021. iText: Hands-free Text Entry on an Imaginary Keyboard for Augmented Reality Systems. In *The 34th Annual ACM Symposium on User Interface Software and Technology* (Virtual Event, USA) *(UIST '21)*. Association for Computing Machinery, New York, NY, USA, 815–825. https://doi.org/10.1145/3472749.3474788

Hamish G. MacDougall and Steven T. Moore. 2005. Marching to the beat of the same drummer: the spontaneous tempo of human locomotion. *Journal of Applied Physiology* 99, 3 (2005), 1164–1173. https://doi.org/10.1152/japplphysiol.00138.2005 PMID: 15890757.

Pavel Manakhov, Ludwig Sidenmark, Ken Pfeuffer, and Hans Gellersen. 2024. Filtering on the Go: Effect of Filters on Gaze Pointing Accuracy During Physical Locomotion in Extended Reality. *IEEE Transactions on Visualization and Computer Graphics* 30, 11 (2024), 7234–7244. https://doi.org/10.1109/TVCG.2024.3456153

Aunnoy K Mutasim, Mohammad Raihanul Bashar, Christof Lutteroth, Anil Ufuk Batmaz, and Wolfgang Stuerzlinger. 2025. There Is More to Dwell Than Meets the Eye: Toward Better Gaze-Based Text Entry Systems With Multi-Threshold Dwell. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems (CHI '25)*. Association for Computing Machinery, New York, NY, USA, Article 1088, 18 pages. https://doi.org/10.1145/3706598.3713781

Diederick C. Niehorster, Thiago Santini, Roy S. Hessels, Ignace T. C. Hooge, Enkelejda Kasneci, and Marcus Nyström. 2020a. The impact of slippage on the data quality of head-worn eye trackers. *Behavior Research Methods* 52, 3 (2020), 1140–1160. https://doi.org/10.3758/s13428-019-01307-0

Diederick C. Niehorster, Raimondas Zemblys, Tanya Beelders, and Kenneth Holmqvist. 2020b. Characterizing gaze position signals and synthesizing noise during fixations in eye-tracking data. *Behavior Research Methods* 52, 6 (2020), 2515–2534. https://doi.org/10.3758/s13428-020-01400-9 Published: 2020/12/01.

Anjul Patney, Marco Salvi, Joohwan Kim, Anton Kaplanyan, Chris Wyman, Nir Benty, David Luebke, and Aaron Lefohn. 2016. Towards Foveated Rendering for Gaze-Tracked Virtual Reality. *ACM Transactions on Graphics* 35, 6, Article 179 (2016), 179:1–179:12 pages. https://doi.org/10.1145/2980179.2980246

Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, and Aaron Courville. 2018. FiLM: Visual Reasoning with a General Conditioning Layer. *Proceedings of the AAAI Conference on Artificial Intelligence* 32, 1 (Apr. 2018). https://doi.org/10.1609/aaai.v32i1.11671

Ken Pfeuffer, Benedikt Mayer, Diako Mardanbegi, and Hans Gellersen. 2017. Gaze + pinch interaction in virtual reality. In *Proceedings of the 5th Symposium on Spatial User Interaction* (Brighton, United Kingdom) *(SUI '17)*. Association for Computing Machinery, New York, NY, USA, 99–108. https://doi.org/10.1145/3131277.3132180

Ken Pfeuffer, Mario Vidal, Jamie Turner, Andreas Bulling, and Hans Gellersen. 2013. Pursuit calibration: Making gaze calibration less tedious and more flexible. In *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology (UIST)*. 261–270. https://doi.org/10.1145/2501988.2501998

Markus Sasalovici, Albin Zeqiri, Robin Connor Schramm, Oscar Javier Ariza Nunez, Pascal Jansen, Jann Philipp Freiwald, Mark Colley, Christian Winkler, and Enrico Rukzio. 2025. Bumpy Ride? Understanding the Effects of External Forces on Spatial Interactions in Moving Vehicles. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems (CHI '25)*. Association for Computing Machinery, New York, NY, USA, Article 280, 28 pages. https://doi.org/10.1145/3706598.3714077

Ludwig Sidenmark, Franziska Prummer, Joshua Newn, and Hans Gellersen. 2023. Comparing Gaze, Head and Controller Selection of Dynamically Revealed Targets in Head-Mounted Displays. *IEEE Transactions on Visualization and Computer Graphics* 29, 11 (2023), 4740–4750. https://doi.org/10.1109/TVCG.2023.3320235

Oleg Špakov. 2012. Comparison of Eye Movement Filters Used in HCI. In *Proceedings of the 2012 Symposium on Eye Tracking Research and Applications (ETRA '12)*. Association for Computing Machinery, New York, NY, USA, 281–284. https://doi.org/10.1145/2168556.2168616

Josef Spjut, Ben Boudaoud, Jonghyun Kim, Trey Greer, Rachel Albert, Michael Stengel, Kaan Aksit, and David Luebke. 2020. Toward Standardized Classification of Foveated Displays. *IEEE Transactions on Visualization and Computer Graphics* 26, 5 (2020), 2126–2134. https://doi.org/10.1109/TVCG.2020.2973053

Mikhail Startsev, Ioannis Agtzidis, and Michael Dorr. 2019. 1D CNN with BLSTM for automated classification of fixations, saccades, and smooth pursuits. *Behavior Research Methods* 51, 2 (2019), 556–572.

Niklas Stein, Diederick C. Niehorster, Tamara Watson, Frank Steinicke, Katharina Rifai, Siegfried Wahl, and Markus Lappe. 2021. A Comparison of Eye Tracking Latencies Among Several Commercial Head-Mounted Displays. *i-Perception* 12, 1 (2021), 1–16. https://doi.org/10.1177/2041669520983338

Mario Vidal, Andreas Bulling, and Hans Gellersen. 2013. Pursuits: Spontaneous interaction with displays based on smooth pursuit eye movement and moving targets. In *Proceedings of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. 439–448. https://doi.org/10.1145/2493432.2493477

E.A. Wan and R. Van Der Merwe. 2000. The unscented Kalman filter for nonlinear estimation. In *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No.00EX373)*. 153–158. https://doi.org/10.1109/ASSPCC.2000.882463

Yushi Wei, Rongkai Shi, Sen Zhang, Anil Ufuk Batmaz, Pan Hui, and Hai-Ning Liang. 2025. Reevaluating the Gaze Cursor in Virtual Reality: A Comparative Analysis of Cursor Visibility, Confirmation Mechanisms, and Task Paradigms. *IEEE Transactions on Visualization and Computer Graphics* (2025), 1–13. https://doi.org/10.1109/TVCG.2025.3622042

Difeng Yu, Xueshi Lu, Rongkai Shi, Hai-Ning Liang, Tilman Dingler, Eduardo Velloso, and Jorge Goncalves. 2021. Gaze-Supported 3D Object Manipulation in Virtual Reality. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) *(CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 734, 13 pages. https://doi.org/10.1145/3411764.3445343

Xucong Zhang, Seonwook Park, Thabo Beeler, Derek Bradley, Siyu Tang, and Otmar Hilliges. 2020. ETH-XGaze: A Large Scale Dataset for Gaze Estimation Under Extreme Head Pose and Gaze Variation. In *Computer Vision – ECCV 2020*, Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm (Eds.). Springer International Publishing, Cham, 365–381.

A Diagnostic metrics and analysis settings

This appendix provides the formal definitions and implementation settings of the diagnostic quantities used in section 3.7.

A.1 Signal definitions and stratification

Let

$$\mathbf{r}_t = \begin{bmatrix} \theta_{r,t} \\ \phi_{r,t} \end{bmatrix} = \mathbf{g}_t^* - \mathbf{g}_t^{\text{raw}} \tag{7}$$

denote the gaze residual in yaw and pitch, where $\mathbf{g}_t^*$ is the ground-truth vector and $\mathbf{g}_t^{\text{raw}}$ is the raw gaze. The residual was analyzed against 12 motion-related features:

$$\mathcal{K} = \left\{ \begin{matrix} \omega_{\text{head},x}, \omega_{\text{head},y}, \omega_{\text{head},z}, v_{\text{head},x}, v_{\text{head},y}, v_{\text{head},z}, \\ \omega_{\text{chest},x}, \omega_{\text{chest},y}, \omega_{\text{chest},z}, v_{\text{chest},x}, v_{\text{chest},y}, v_{\text{chest},z} \end{matrix} \right\}. \tag{8}$$

All diagnostic quantities were computed within strata defined by `Context|EventType`. Saccade segments were excluded from the analysis, so the reported results reflect only non-saccadic strata. Extremely short segments were discarded to avoid unstable spectral estimates.

A.2 Analysis settings

Table 5 summarizes the settings used in the residual–motion coupling analysis.

Table 5. Diagnostic analysis settings used in section 3.7.

Sampling rate f_s	90.0 Hz
Maximum lag window	±0.3 s
Hot-pair threshold	peak $\lvert\rho\rvert \geq 0.10$
Minimum useful segment length	16 frames
Welch segment length	$n_{\text{perseg}} = 2048$
Welch overlap	50%
Coherence threshold	$\gamma^2(f) > 0.3$
Low-frequency cutoff for band aggregation	$f_{\min} = 0.2$ Hz

A.3 Time-domain quantities

For a residual axis $u_t \in \{\theta_{r,t}, \phi_{r,t}\}$ and a motion signal $x_{k,t}$, the normalized cross-correlation was defined as

$$\rho_{u,k}(\tau) = \frac{\sum_t (u_t - \bar{u})(x_{k,t-\tau} - \bar{x}_k)}{\sqrt{\sum_t (u_t - \bar{u})^2}\sqrt{\sum_t (x_{k,t} - \bar{x}_k)^2}}, \qquad |\tau| \leq \tau_{\max}, \quad (9)$$

where $\tau_{\max} = 0.3$ s, and $\bar{u}$ and $\bar{x}_k$ are the sample means within the analyzed segment.

The *peak cross-correlation magnitude* was reported as

$$\rho^{\max}_{u,k} = \max_{|\tau| \leq \tau_{\max}} \left|\rho_{u,k}(\tau)\right|, \quad (10)$$

and the corresponding *peak lag* was

$$\tau^*_{u,k} = \arg \max_{|\tau| \leq \tau_{\max}} \left|\rho_{u,k}(\tau)\right|. \quad (11)$$

A residual–motion pair was retained as a *hot pair* if

$$\rho^{\max}_{u,k} \geq \rho_{\text{th}}, \qquad \rho_{\text{th}} = 0.10. \quad (12)$$

A.4 Spectral estimation and magnitude-squared coherence

For each *hot pair*, the auto-spectral densities $S_{uu}(f)$ and $S_{kk}(f)$ and the cross-spectral density $S_{u,k}(f)$ were estimated using a Hamming-windowed Welch method. For segments shorter than n_{perseg} but longer than the minimum useful length, a zero-padded periodogram fallback was used on the same frequency grid.

The magnitude-squared coherence was then defined as

$$\gamma^2_{u,k}(f) = \frac{|S_{u,k}(f)|^2}{S_{uu}(f)\, S_{kk}(f)}, \qquad 0 \leq \gamma^2_{u,k}(f) \leq 1. \quad (13)$$

The *peak coherence* for a pair was

$$\gamma^{2,\max}_{u,k} = \max_f \gamma^2_{u,k}(f). \quad (14)$$

A.5 Above-threshold coherence bands

To localize the frequencies at which residual–motion coupling was supported, a binary coherence mask was defined as

$$M_{u,k}(f) = \mathbb{I}\left[\gamma^2_{u,k}(f) > \eta_{\text{coh}}\right], \qquad \eta_{\text{coh}} = 0.3, \quad (15)$$

where $\mathbb{I}[\cdot]$ is the indicator function.

An *above-threshold coherence band* was defined as a maximal contiguous interval over which the mask equaled one:

$$B_{u,k,m} = [f_{m,\ell}, f_{m,h}] \subseteq \{f : M_{u,k}(f) = 1\}, \quad (16)$$

for the m-th contiguous supported interval of pair (u, k).

The *peak frequency* of a band was

$$f^*_{u,k,m} = \arg \max_{f \in B_{u,k,m}} \gamma^2_{u,k}(f), \quad (17)$$

and its *band peak coherence* was

$$\gamma^{2,\max}_{u,k,m} = \max_{f \in B_{u,k,m}} \gamma^2_{u,k}(f). \quad (18)$$

A.6 Aggregation of coherence-supported bands

To obtain compact frequency support summaries for the dominant fixation conditions discussed in the main text, coherence-supported bands were aggregated across

$$C_{\text{fix}} = \left\{ \begin{matrix} \texttt{Simulated Vehicle Movement|Fixation}, \\ \texttt{Walking|Fixation} \end{matrix} \right\}. \quad (19)$$

For a given residual axis u and motion channel k, the aggregated support mask was

$$M^{\text{agg}}_{u,k}(f) = \max_{c \in C_{\text{fix}}} M^{(c)}_{u,k}(f), \quad (20)$$

where $M^{(c)}_{u,k}(f)$ is the coherence mask for pair (u, k) within stratum c.

Very-low-frequency drift was removed by applying

$$M^{\text{ret}}_{u,k}(f) = M^{\text{agg}}_{u,k}(f)\, \mathbb{I}[f \geq f_{\min}], \qquad f_{\min} = 0.2 \text{ Hz}. \quad (21)$$

The retained coherence-supported frequency set was then

$$\Omega^{\text{ret}}_{u,k} = \{f : M^{\text{ret}}_{u,k}(f) = 1\}. \quad (22)$$

These retained supports correspond to the compact ranges summarized in section 3.7, such as the narrow-band yaw-related support and the broader, more multi-channel pitch-related support.

B Causal State Machine Logic

The state machine Φ, as described in section 4.3, ensures temporal consistency by managing transitions between three conceptual states: *Saccadic* ($S = 1$), *Stable* ($S = 0, C \geq d_{\text{ref}}$), and *Recovery* ($S = 0, C < d_{\text{ref}}$). Let $S_t \in \{0, 1\}$ denote the binary state and C_t denote a frame counter. The transition logic is formally defined in Algorithm 1.

Algorithm 1 Causal State Machine for Saccade Fusion Control

Require: Raw saccade probability p_t^{sac}, previous state $\mathbf{z}_{t-1} = (S_{t-1}, C_{t-1})$
Require: $\tau_{\text{enter}}, \tau_{\text{exit}}, d_{\text{hold}}, d_{\text{ref}}$
Initialize: Before processing the first frame, initialize $\mathbf{z}_{-1} = (S_{-1}, C_{-1}) \leftarrow (0, d_{\text{ref}})$. ▷ Start in Stable Phase
if $S_{t-1} = 0$ **then** ▷ Currently in Stable or Recovery Phase
  if $p_t^{\text{sac}} > \tau_{\text{enter}} \wedge C_{t-1} \geq d_{\text{ref}}$ **then**
    $S_t \leftarrow 1, C_t \leftarrow 0$ ▷ Enter Saccadic Phase
  else
    $S_t \leftarrow 0, C_t \leftarrow C_{t-1} + 1$ ▷ Stay/Recovery increment
  end if
else ▷ Currently in Saccadic Phase
  if $p_t^{\text{sac}} < \tau_{\text{exit}}$ **then**
    $C_t \leftarrow C_{t-1} + 1$
    if $C_t \geq d_{\text{hold}}$ **then**
      $S_t \leftarrow 0, C_t \leftarrow 0$ ▷ Enter Recovery Phase
    else
      $S_t \leftarrow 1$
    end if
  else
    $S_t \leftarrow 1, C_t \leftarrow 0$ ▷ Reset exit counter
  end if
end if
Output Gating Value $\tilde{p}_t$**:**
if $S_t = 1$ **then**
  $\tilde{p}_t \leftarrow 1.0$ ▷ Saccadic Phase: Binary High
else
  if $C_t < d_{\text{ref}}$ **then**
    $\tilde{p}_t \leftarrow p_t^{\text{sac}}$ ▷ Recovery Phase: Soft Pass-through
  else
    $\tilde{p}_t \leftarrow 0.0$ ▷ Stable Phase: Binary Low
  end if
end if
return $(\tilde{p}_t, \mathbf{z}_t = (S_t, C_t))$

This state machine transition logic partitions eye-movement behavior into three functional phases:

Saccadic Phase: Triggered when the probability exceeds τ_{enter}, the machine enforces a binary high output ($\tilde{p}_t = 1.0$), which minimizes the gating weight w_t to fully attenuate the residual and prevent motion lag during rapid gaze shifts.

Recovery Phase: Immediately following a saccade, the system enters a refractory period for d_{ref} frames. During this interval, the machine is prohibited from re-entering the saccadic state and passes the raw probability p_t^{sac} directly to the output. This soft transition allows the system to naturally dampen post-saccadic oscillations using the underlying data distribution.

Stable Phase: Once the refractory counter exceeds d_{ref}, the system reaches the default state for fixation and pursuit. In this phase, a binary low output ($\tilde{p}_t = 0.0$) is enforced, ensuring the gating weight w_t remains high and the Motion-FiLM predictor provides maximum correction accuracy without interference from low-level probability noise.

C Hyperparameter Configurations

This appendix provides the hyperparameter configurations selected during the validation process. Following the protocol in section 5.1.3, all parameters were determined via grid search on the validation set of each fold.

Table 6. Hyperparameter configurations for IntentGaze across 7-fold cross-validation. d_{hold} and d_{ref} are in frames.

Fold	Thresholds		Durations		Probabilities	
	τ_{enter}	τ_{exit}	d_{hold}	d_{ref}	p_{off}	p_{low}
Fold 1	0.85	0.55	14	18	0.85	0.40
Fold 2	0.85	0.40	9	33	0.90	0.30
Fold 3	0.80	0.40	17	33	0.40	0.35
Fold 4	0.85	0.50	15	33	0.50	0.30
Fold 5	0.85	0.45	16	32	0.55	0.25
Fold 6	0.85	0.50	17	18	0.40	0.25
Fold 7	0.85	0.55	13	27	0.95	0.00

Table 6 presents the parameters for IntentGaze. These were jointly tuned across both tasks to maintain their character as a unified correction method. The observed variance in parameters across folds reflects the model's adaptation to different training partitions.

As a post-hoc sensitivity analysis, we evaluated a fixed median state-machine configuration. Its test Accuracy and Stability differed from the fold-specific settings only at the fourth decimal place. Across 843 validation-defined Pareto configurations, the fold-averaged angular ranges were at most 0.00308°, while Saccade Lag varied by 2.90 ms on average. This indicates a broad plateau of near-equivalent operating points; for deployment, any validation-defined Pareto configuration satisfying the Saccade Lag constraint may be used.

Table 7. Selected hyperparameters for baseline methods. For these methods, the optimal configurations were observed to be uniform across all seven validation folds.

Method	Task	Selected Parameters
One Euro	Fixation	$f_{\text{min}} = 4.50$, $\beta = 0.12$, $f_{\text{d\_cutoff}} = 2.50$
	Pursuit	$f_{\text{min}} = 8.00$, $\beta = 0.13$, $f_{\text{d\_cutoff}} = 4.00$
LP-SDn	Fixation	$f_{\text{cutoff}} = 0.3$, $n = 3$, $\delta_{\text{rst}} = 5.0°$
	Pursuit	$f_{\text{cutoff}} = 25.0$, $n = 1$, $\delta_{\text{rst}} = 5.0°$
UKF-SDn	Fixation	$q_{\text{UKF}} = 2.2$, $r_{\text{UKF}} = 6.5$, $n = 3$, $\delta_{\text{rst}} = 5.0°$
	Pursuit	$q_{\text{UKF}} = 6.5$, $r_{\text{UKF}} = 0.2$, $n = 1$, $\delta_{\text{rst}} = 1.5°$

Table 7 summarizes the configurations for the baseline filters. While these were optimized independently for each fold and task, the grid-search procedure converged to a consistent set of optimal parameters across all seven folds within the resolution of our search space. This suggests a relatively stable performance profile for these baselines on the current dataset.

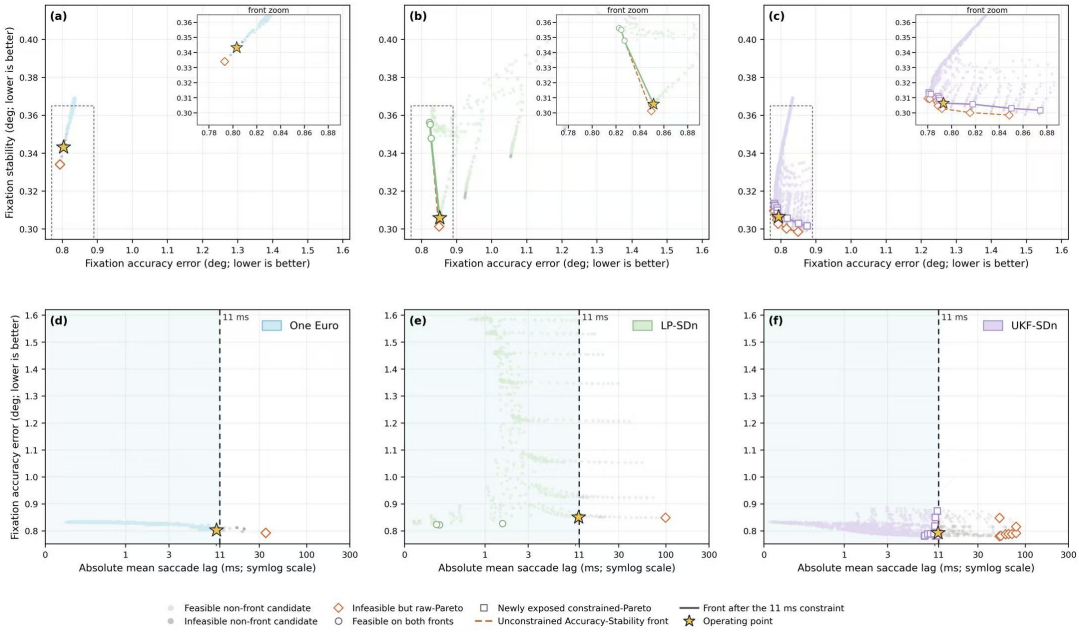

Fig. 9. Effect of the Saccade Lag constraint on the Accuracy–Stability trade-off for One Euro, LP-SD*n*, and UKF-SD*n* on the fold-1 validation set. Panels (a)–(c) show Fixation Accuracy Error versus Fixation Stability, with insets enlarging the Pareto-front regions. Panels (d)–(f) show Fixation Accuracy Error versus absolute mean Saccade Lag on a symmetric logarithmic scale. The shaded region satisfies the 11 ms constraint. Dashed curves denote unconstrained fronts, solid curves denote constrained fronts, diamonds mark excluded Pareto points, squares mark newly exposed points, and stars indicate the evaluated operating points.

D Post-hoc Analysis of the Saccade-Lag Constraint

We conducted a diagnostic analysis on the fold-1 validation set to examine how the Saccade Lag constraint affected the Accuracy–Stability trade-off. For each method, we first computed the unconstrained Pareto front by minimizing Fixation Accuracy Error and Fixation Stability. We then retained only candidates with an absolute mean Saccade Lag of at most 11 ms and recomputed the Pareto front within this feasible set, as shown in fig. 9.

For One Euro, all 12 unconstrained Pareto points violated the constraint, leaving one feasible point that matched the evaluated operating point. For LP-SD*n*, one of four unconstrained points was removed, and the constrained front contained four points, including one newly exposed point. For UKF-SD*n*, all eight unconstrained points were excluded, exposing an 11-point constrained front. The evaluated operating point lay on the constrained front for all three methods.

These results confirm that the lag criterion was an active constraint. Configurations favored by Accuracy and Stability alone could introduce excessive delay, whereas enforcing the 11 ms threshold exposed feasible trade-offs previously dominated by lag-violating configurations.