mmET: mmWave Radar-Based Eye Tracking on Smart Glasses

Ruichun Ma*
Meta Platforms, Inc.
ruichun.ma@yale.edu

Yasuo Morimoto†
Meta Platforms, Inc.
yso@meta.com

John S. Ho
Meta Platforms, Inc.
johnsho@meta.com

Sam Shiu
Meta Platforms, Inc.
boon@meta.com

Jiang Zhu†
Meta Platforms, Inc.
jiangzhu@meta.com

Abstract

With the growing popularity of VR and AR devices, eye tracking has become a critical user interface and input modality for on-device AI agents. However, a compact, power-efficient, and robust eye tracking solution for AR/smart glasses remains an unsolved challenge. In this paper, we present mmET, the first mmWave radar-based eye tracking system on glasses. Our system, implemented as a pair of prototype glasses, utilizes sub-1cm mmWave radars placed near the eyes. The radars transmit FMCW signals and capture the reflections from the eyes and surrounding skin as the system input. To refine gaze estimation accuracy and data efficiency, we propose several novel methods: (1) concatenating multiple chirps and beamforming with learnable weights to improve resolution, (2) a novel neural network architecture to enhance robustness against remounting, (3) pretraining with contrastive loss to enable fast adaptation for new users. Experiments with 16 participants show that mmET achieves an average angular gaze direction error of 1.49° within sessions and 4.47° across remounting sessions, and reduces the training data needed for new users by 80% using the pretrained model.

CCS Concepts

• **Computer systems organization** → **Embedded and cyber-physical systems**; • **Human-centered computing** → **Human computer interaction (HCI)**; • **Computing methodologies** → **Machine learning**.

Keywords

Wireless Sensing, mmWave, Eye Tracking, Smart Glasses, Machine Learning

ACM Reference Format:
Ruichun Ma, Yasuo Morimoto, John S. Ho, Sam Shiu, and Jiang Zhu. 2025. mmET: mmWave Radar-Based Eye Tracking on Smart Glasses. In *The 23rd ACM Conference on Embedded Networked Sensor Systems (SenSys '25), May 6–9, 2025, Irvine, CA, USA.* ACM, New York, NY, USA, 13 pages. https://doi.org/10.1145/3715014.3722050

*Ruichun Ma did this work as an intern at Meta Platforms, Inc.
†Corresponding authors are Yasuo Morimoto and Jiang Zhu.

SenSys '25, May 6–9, 2025, Irvine, CA, USA

ACM ISBN 979-8-4007-1479-5/2025/05
https://doi.org/10.1145/3715014.3722050

Figure 1: Illustration of mmWave radar-based eye tracking on smart glasses. mmET predicts user's gaze direction based on the reflected mmWave radar signals from eyeballs, eyelids and surrounding skin.

1 Introduction

Eye tracking is an enabling technology for various applications, including psychology studies [34, 47], health monitoring [43, 50], hands-free user interfaces [28, 41] and inferring user intent [46]. Recently, eye tracking has become increasingly important for metaverse or XR (extended reality) applications, ranging from foveated rendering to user interfaces. For example, recent VR devices allow users to navigate the virtual worlds with their eyes, e.g., selecting UI elements with their gaze [1, 3]. For AR or smart glasses, eye tracking can provide gaze-based input that enables hand-free interactions and context-aware, personalized AI agents [4]. However, integrating eye tracking in the compact form factor of smart glasses remains challenging. The small form factor of glasses imposes very limited power and space budgets. This motivates an eye tracking system that balances high accuracy with minimal sensor size and power usage.

Current eye tracking solutions typically perform computer vision-based eye glint detection using infrared (NIR) LEDs and cameras. Commercial eye tracking glasses [5, 8] achieve state-of-the-art accuracy (~ 1° average error) but suffer from a high hardware footprint, i.e., size, cost, and power consumption, complicating their integration into glasses. They are also prone to occlusion by eyelashes and sensor saturation by strong ambient light, e.g., outdoor sunlight. Recent research work has explored alternative approaches. Li et al. present [32] an accurate battery-free eye tracker using NIR LEDs and photodiodes, but its sensor array exceeds 6 cm per eye and is constrained to indoor scenarios due to sunlight interference. GazeTrak [30] presents a promising accurate eye tracking system on glasses using acoustic waves, but the microphone array spans the entire glasses frame and needs extensive calibration/training for

Figure 2: Example usage scenario of gaze input on smart glasses. Gaze input provides extra context of user's words to enable a more intelligent AI assistant.

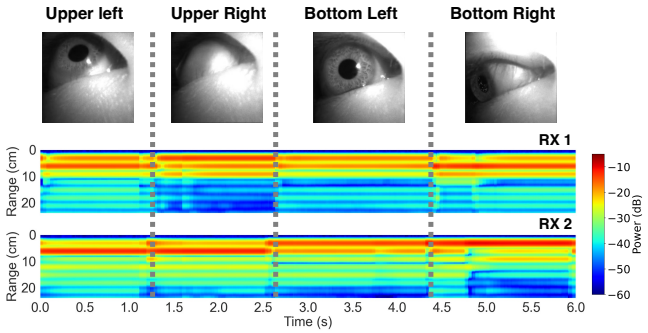


Figure 3: Visualization of eye movements and corresponding radar signals. We show the radar spectrogram from two RX antennas, as an example of clear correspondence between eye movements and radar signal changes over time.

each user. With existing solutions fall short in at least one aspect, we aim to address the gap with a system that offers accuracy, robustness, compactness, and low power consumption together.

In this paper, we present mmET, the first mmWave radar-based eye tracking system on glasses, as illustrated in Figure 1. Together with a front-facing scene camera, users' gaze direction can provide important context for AI agents to understand the focus in the current scene (Figure 2). Compared with other approaches, mmWave radars have a small array size, low power consumption, and immunity to eyelash occlusion and outdoor sunlight, while providing high sensitivity to eye movements. Moreover, one unique benefit of antennas is that they can be transparent [15, 42] and reuse the space occupied by optical glass lenses, meanwhile ensuring proximity to eyes. We compare our system with selected eye tracking solutions for glasses in Table 1, demonstrating that mmET is the first one that provides a combination of all necessary properties for AR/smart glasses – accuracy, immunity to sunlight, small size, and low power.

mmET estimates the user's gaze direction by analyzing reflected FMCW (frequency modulated continuous wave) signals from the eyes. The displacement sensitivity of radars (~100 μm from phase values) and multiple sensing channels (8 channels per eye) enable detection of subtle movements of the eye, eyelids, and the surrounding skin to infer the gaze direction. Figure 3 visualizes the radar spectrogram from two antennas when the user's eyeball orientates towards different directions, showing a clear correspondence between eyeball orientation and radar signals. Spectrogram colors represent dB-scale signal power, while the range axis represents the reflected signal path lengths, derived via FFT processing of down-converted FMCW signals. Unlike conventional radar localization, which detects object peaks, our system analyzes fine-grained amplitude values across range bins, antennas, and beamforming vectors to estimate eye status.

Accurate and robust angular gaze direction prediction based on radar signals presents several challenges. First, eye and surrounding skin movements are often subtle, especially when the gaze direction shifts by only a few degrees. Second, radar signals are sensitive to glasses movement relative to the user's head, especially after remounting. A machine learning (ML) model may overfit to trained sessions and fail to generalize to unseen sessions after remounting. Lastly, extensive training or calibration data collection is often infeasible, making it challenging to adapt to new users with limited training data.

We address these challenges with tailored signal processing, a novel neural network architecture and a cross-modal pretraining strategy, as detailed in section 3: (1) We carefully craft the input radar signals to boost range and angular resolutions by concatenating multiple radar chirps and beamforming with learnable weights respectively. (2) To gain robustness against remounting, we use a ResNet backbone and multiple prediction heads to regress multiple eye pose targets, including gaze direction, relative location and orientation of eyeballs. This multitask learning approach helps the neural network (NN) model to distinguish eye movements from remounting-induced changes. (3) To reduce calibration effort and improve generalization, we pretrain a radar model using cross-modal contrastive learning, based on multiple users' eye images and radar signal data, and fine-tune the pretrained model for new users. Note that multitask learning [11] and contrastive learning [13] are general ML approaches, applicable to various domains and problems, but they do not solve our challenges directly. Instead, we adapt them with tailored model architectures and loss functions for our mmWave sensing problem.

We build an eye tracking prototype with two mmWave radars [2] mounted at the left and right corners of the glasses frame (Figure 10). The mmWave antenna array is integrated with an mmWave chip, resulting in a compact 8 mm sensor size. Two radar MCU boards collect radar signals and stream them to the computer for processing.

We evaluate the performance of mmET with data from 16 users. Each participant wore the prototype and followed a marker displayed on a screen in front of them to collect data; users remounted the glasses multiple times, with each mounting considered as starting a new *session*. We collect radar frames paired with eye images (for ground truth) at 100 frames per second (FPS), 20 minutes per user. The metric for eye tracking accuracy is the angular error of the estimated gaze direction. For personalized models trained on individual user's dataset, mmET achieve an average angular error of 1.49° across 10 users when tested with data split from training sessions (in-session). When tested with unseen sessions after remounting glasses (cross-session), the error increases to 4.47° on average. For

	Sensor Type	Accuracy	Adversarial Factors	Sensor Size	Power Consumption
Pupil Labs [5]	LEDs and Cameras	1.3°	Sunlight, FOV, hair occlusion	~1 cm	>100 mW
Cider [39]	LEDs and Camera	0.6°	Sunlight, FOV, remounting	~1 cm	40 μW
Li et al [32]	LEDs and Photo diodes	< 2°	Sunlight, remounting	>6cm	~40 μW
GazeTrak [30]	Speakers and Mics	3.6°	remounting	>5cm	16.4 mW
mmET	mmWave radars	1.49°	remounting	8mm	6 mW

Table 1: Comparison of eye tracking systems on glasses. [2]

our pretrained model trained on data from 10 users, we achieve comparable accuracy on 6 new users after fine-tuning the model with only 5-minute calibration per user. Compared to GazeTrak [30] (Table 2), we improve in-session accuracy by 2.1° and cross-session accuracy by 1.2°, while requiring only half the training data and ~13.3% of the sensor array size.

To summarize, we make the following contributions:

- We introduce mmET, the first mmWave-based eye tracking system for AR/smart glasses, which achieves a combination of accuracy, robustness, compactness, and low power consumption for the first time.
- We propose novel mmWave sensing techniques: (1) concatenating chirps and learnable beamforming to enhance resolution, (2) estimating eye locations and orientations to improve robustness against remounting, (3) pretraining with radar signals and eye images to learn generalizable features and reduce calibration data needed.
- We evaluate mmET on 16 users, achieving an average angular gaze direction error of 1.49° within sessions and 4.47° across remounting sessions, and an 80% reduction in required training data of new users using a pretrained model.
- We present a demo video [1] to showcase system performance.

2 Background and Related Work

2.1 Eye tracking

Eye tracking is the process of estimating the direction or point of gaze, i.e., which direction or which position the user is looking at. Various eye-tracking solutions exist, each with its own advantages and limitations.

Webcam-based eye tracking. Researchers have extensively explored web camera-based eye tracking [26, 55] with various potential applications including human-computer interaction, use behavior or psychology studies [44, 45, 53]. Webcams have the advantages of low-cost and widespread availability, which facilitates the broader use of eye tracking. However, webcam-based eye tracking methods have relatively low accuracy and are susceptible to factors like lighting conditions, head and camera orientations. In this paper, we focus on eye tracking solutions on wearable devices, specifically, eyeglasses form factor.

Eye camera-based eye tracking. Wearable-based eye tracking devices gain proximity to users' eyes and avoid the impact of head/camera orientation as the hardware is mounted on users' face. The common approach is using NIR (near-infrared) LEDs and eye cameras, which take NIR images of user's eyes and estimate gaze direction/position according to the relative position of pupils. Many commercial products, such as Pupil Labs Neon [5] and Tobii Glasses 3 [8], follow this approach. Such eye cameras are very expensive (thousands of dollars) and power-hungry (> 100mW for eye cameras alone) and require external batteries to power the device. Due to the nature of near-infrared light, they are also prone to strong sunlight or ambient light, and hair/eyelash occlusion. To mitigate outdoor sunlight interference, they typically increase the brightness of LEDs, worsening power consumption and necessitating frequent recalibration when ambient light condition changes.

iShadow [38] proposes a novel design to reduce power consumption by reducing the redundancy in eye images with a low-power image sensor, but it cannot operate under outdoor sunlight. Cider [39] explores the power-robustness trade-offs by adapting to different illumination situations and achieves high accuracy and frame rate. However, Cider remains susceptible to sunlight interference, and changes in the relative position between glasses and eyes during remounting can increase error. Another limitation, noted in the paper, is the camera's limited field of view (FoV), which can cause users' eyes to fall out of the camera's view—a common issue for camera-based solutions.

VR/AR devices. VR devices often incorporate eye tracking features, as they have larger form factors than glasses and relatively large (or external) batteries, which allow the use of multiple NIR cameras to capture images of users' eyes [1, 3]. Users' eyes are also enclosed in a dark environment, which excludes the sunlight interference. For AR and smart glasses, it is significantly more challenging to integrate eye tracking solutions. To support eye tracking with NIR LEDs and cameras, AR glasses often show a bulky form factor and require an external battery pack to maintain a reasonable battery life. Moreover, outdoor use introduces sunlight interference, which can saturate NIR camera sensors. For AI assistant use cases, smart glasses [4, 7] so far do not widely use eye tracking and reply on users pointing cameras towards the target direction, which can fail in cluttered scenes (Figure 2). The power consumption of eye-tracking hardware alone is difficult to determine, but our measurements show that eye cameras consume a minimum of 100 mW, reaching up to 200 mW depending on FPS and LED brightness.

Non camera-based eye tracking. To reduce power consumption, and avoid interference from outdoor sunlight, recent work explores non-camera based eye tracking system. Li et al. present [31, 32] a low-power low-cost eye tracker using NIR LEDs and photodiodes. However, the size of sensor is over 6 cm for each eye and remains constrained to indoor scenarios where ambient sunlight interference is limited. EyeGesener [49] presents an eye gesture listener for smart

[1] We visualize mmET performance after remounting with demos. Video link: https://drive.google.com/file/d/19odygKsT1zpPJFLscONSFzOZCtTWGErd

[2] This is not an exhaustive list. We exclude the power consumption of computation to focus on comparing sensing approaches.

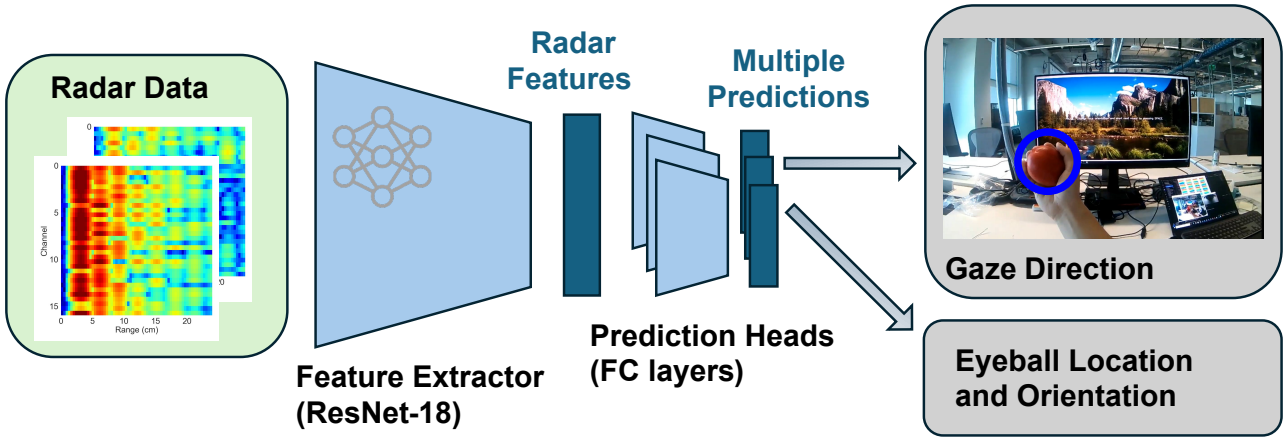


Figure 4: Overview of neural network model design. Using crafted radar data as input, we extract radar features with ResNet as backbone and make multiple predictions, including gaze direction and eyeball status.

glasses but is limited to classification of coarse-grained gaze directions. GazeTrak [30] presents the first acoustic-based eye tracking system on glasses and shows promising results, but it needs extensive calibration/training for each user and after remounting. Moreover, the microphone array spans the entire glasses frame, posing challenges for hardware integration on AR/smart glasses. mmET falls into the category of non-camera-based eye tracking systems, and we mainly compare against GazeTrak, the most relevant non-camera-based system.

2.2 mmWave Radar-based Sensing

mmWave radars [23] are sensor devices that transmit and receive reflected mmWave signals, which are typically encoded as chirps using FMCW (frequency modulated continuous wave). They can measure the position, direction, and movement of detected targets accurately with the small wavelength of mmWave. Due to their high accuracy, low power consumption, and penetration for certain visual blockage, they have been used for various industrial [24], automotive [12, 16, 18, 37, 51], and IoT applications [14, 25, 29, 40, 48]. Recent work also explores deploying radars to sense face or eye status. They show the capabilities of radar signals for facial expression recognition [54], heart rate monitoring [21], seismocardiography capture [17], blink detection [10, 20], eye gesture/movement [35, 56]. However, existing works often use a standalone radar rather than wearable glasses, and are limited to coarse-grained classification, such as blinks (eye closed or open), and eye gestures (left or right). Accurate eye tracking with mmWave radars, a regression task for angular gaze direction down to a sub-degree accuracy, remains challenging and unexplored.

Our Approach. To the best of our knowledge, mmET is the first mmWave radar-based eye tracking system. mmET advances mmWave sensing accuracy using tailored signal processing, a novel ML model design, and a cross-modal pre-training strategy (section 3).

Comparison. We compare mmET with related eye tracking systems in Table 1. We select state-of-the-art systems with distinct sensing approaches for ease of comparison. Light-based solutions commonly suffer from sunlight interference, making them unsuitable for outdoor AR/smart glasses. Low-power eye-tracking methods often struggle with remounting and glasses movement. This is likely because low-power solutions often have reduced input data dimensions, such as low-resolution cameras, which can lead to reduced robustness. We mitigate this issue with multitask learning in subsection 3.3. Our system provides low gaze direction error, a compact sensor size, low power consumption, and is not impacted by sunlight – a combination of characteristics suited for AR/smart glasses.

Power consumption. For a fair comparison, we list sensor power consumption and exclude computation power for all solutions in Table 1 to the best of our knowledge. For mmET, we measure power consumption based on the electric current of the radar sensor board, which includes the RF front end, ADC, and raw data transmission. For Pupil Labs glasses, we estimate LED and camera power based on measurements of similar solutions and use 100 mW as an estimated minimal power consumption, while excluding front-view camera and video streaming power consumption. We note that ML computation power depends on the application processor and remains similar across different sensors. For instance, AR glasses using the Qualcomm XR2Gen2 [6] run face tracking and eye tracking ML models at approximately 20-40 mW and mmWave radars offer potential savings due to lower input data dimensions compared to cameras.

Safety considerations. Ensuring biological safety is critical for wireless devices. 60 GHz mmWave signals are non-ionizing radiation, excluding the risk of causing cancer. Therefore, potential health concerns relate to heating effects, i.e., raising the temperature of human body. The metric to set the limit is power density (PD) [9, 22]. IEEE C95.1-2005 [9] states the maximum permissible exposure at 60 GHz is 10 W/m^2 (i.e., 1 mW/cm^2), aligned with FCC and ICNIRP guidelines [22]. Related studies [27, 52] show that 60 GHz mmWave radiation under PD restrictions on animals' eyes cause no ocular damage. Given the proximity to eyes for our system, we have ample SNR for the radar signals and intentionally tune down the radar transmission power. We perform power density measurement with mmWave radars at 60GHz, averaging over a 4 cm^2 area at 2 mm distance. The measured PD is 0.79 W/m^2 for continuous wave

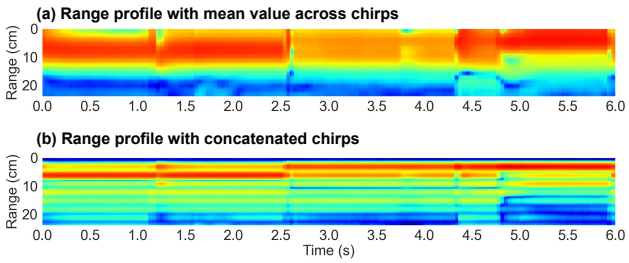


Figure 5: Processing radar range profile. Concatenating the radar chirps provides a range profile with higher resolution.

operation. Note that our system uses bursts of FMCW frames, with a duty cycle of 1-3% (depending on FPS), so the average power density is over 100× lower than the safety limit.

3 System Design

3.1 Overview

With FMCW radar signals from our hardware prototype as input, we train a neural network model to predict the corresponding gaze direction, employing tailored signal processing and neural network architecture (Figure 4). We first process the radar signals to enhance the resolution. Specifically, we concatenate signals from multiple radar chirps to increase the frequency/range resolution and employ beamforming with learnable weights to extract spatial information. Next, we combine signals from two radars as one radar data frame and use both the current radar data frame and the difference with the last frame as input. With the input, we utilize ResNet-18 as the backbone to extract radar features and multiple fully-connected layers to make predictions for multiple targets: gaze direction, eyeball location and orientation. Additionally, to minimize the calibration/training needed for users, we pretrain a ML model with multiple users' data using contrastive learning, which enables adaptation to new users with significantly reduced calibration.

3.2 Crafting FMCW Radar Signals

Following the standard processing of down-converted FMCW signals, we perform FFT on the signals to get the frequency domain response, i.e., the range profile of mmWave radar, where each range bin corresponds to the reflected signals with a specific path distance. The radar range bin resolution follows

$$d_{res} = \frac{c}{2B} \tag{1}$$

where c is the speed of light and B is the FMCW bandwidth.

Concatenating chirps to improve resolution. A key challenge in our use case is the short distance between radars and eyes, resulting in only a few range bins within the meaningful range. Our experiments show that the useful range extends up to 24 cm, as signals may reflect multiple times between the face and the glasses. To prevent unrelated data from misleading our ML model, we only use this range of radar range profile.

This short range leads to low input data dimension and low resolution that hurt performance. To solve this, we concatenate signals of multiple adjacent FMCW chirps to increase the length of signals

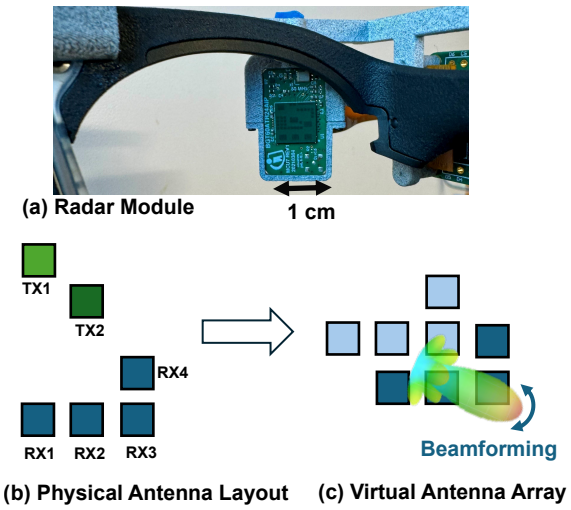


Figure 6: Beamforming with the virtual antenna array. We perform learnable beamforming with a 2×4 virtual antenna array from our radar module.

in one frame. The increase of time domain signal length, in turn, increases the resolution of frequency domain range profile:

$$d_{res} = \frac{c}{2BN_c} \tag{2}$$

where N_c is the number of chirps we concatenated. mmWave radars typically transmit multiple chirps in one frame to capture Doppler information. However, for our eye tracking use case, eyes and surrounding skin move slowly or even keep stationary. Within a single frame, chirp-to-chirp differences are minimal. This means we cannot extract Doppler information, but can have longer signal duration by concatenating signals. We choose to use 4 chirps per frame. Because of the phase discontinuity between chirps, concatenating more chirps leads to zero values in the range profile and brings no extra gain. Figure 5 compares the radar range profiles using mean value across chirps versus concatenated chirps. With concatenated chirps, we can get a higher resolution, showing clearer peaks and more distinguishable differences between different eye status.

Is the resolution enough? A key question is whether radars provide enough resolution to capture eye details such as eyeball orientation and skin movement. Concatenating chirps improves range bin resolution as the basis for further steps. For our configurations shown in section 4, the resolution appears to be around one centimeter, which may seem insufficient for detecting subtle eye changes. However, the range bin resolution does not directly determine the minimal detectable eye motion. Conventional range resolution describes the ability to distinguish two separate objects, whereas our goal is to analyze the status of a single close-proximity object – the eye. Instead of identifying distinct peaks in the range profile, we leverage variations in the summed multipath reflections from different parts of the eye and surrounding skin. Although individual eye structures may not produce clearly separate peaks, changes in the overall range profile—specifically in amplitude— correspond to gaze direction shifts (Figure 3). We use all range bins including the first few and

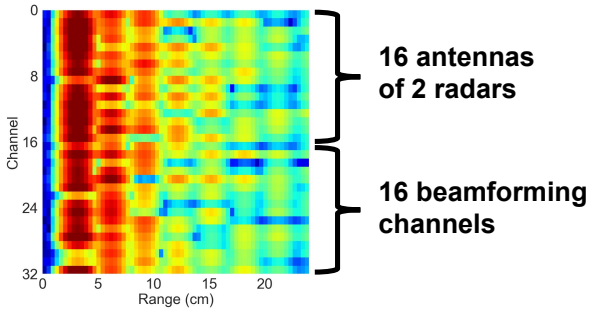


Figure 7: Radar data frame format. We use 32 channels in total, including 16 channels from (virtual) antennas of 2 radars and 16 beamforming channels.

the fine-grained amplitude variations provide sufficient information for accurate gaze direction estimation.

Learnable beamforming. Each of our radar has 2 TX (transmitting) antennas and 4 RX (receiving) antennas, which forms an 8 element virtual array as shown in Figure 6. The array allows us to perform beamforming, i.e., steer the radar receiving beam towards different directions to collect spatial information. Intuitively, each beam gives us information about different regions of the eye. However, we cannot manually select the beam directions, i.e., a fixed set of beamforming weights/configurations. First, the virtual array is irregular, making it difficult to manually specify beams that fully utilize the array size. Second, the eyes are so close to array so that they are not in the far field. Conventional beamforming formulation may break due to near field coupling. Thus, we treat the beamforming weights (phase shift values for each antenna's signals) as learnable parameters, which can be optimized during ML model training using gradient descent. This is similar to prior work on learnable beamforming for mmWave communication [36, 57].

Assembling the radar frame. We combine all the signals as a radar data frame, including 16 channels from all antennas of 2 radars and 16 channels from beamforming, as shown in Figure 7. Each beamforming channel corresponds to the result by applying one set of beamforming weights/configurations.

$$\mathbf{H}_{bf} = \mathbf{W}_{bf}\mathbf{H}_{ant} \tag{3}$$

where W_{bf} is a 16×16 matrix of beamforming weights, H_{ant} is a 16×32 matrix of raw signals from 16 antennas, and H_{bf} is 16-channel results after beamforming. The final radar data frame is a matrix that combines $\mathbf{H}_{bf}$ and $\mathbf{H}_{ant}$.

To extra time-domain differential information, we also calculate the different between two neighboring radar data frames. We use both the current radar data frame and inter-frame difference data as the final input for our ML model.

3.3 Multitask Learning for Robustness

With the processed radar data input, we use a neural network to estimate/predict the corresponding gaze direction. We use ResNet-18 [19] as the backbone to extract radar features and multiple fully connected layers to output the gaze direction, eyeball location and orientation.

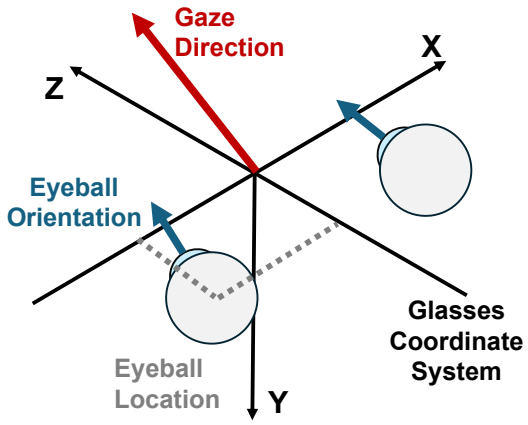


Figure 8: Gaze direction and related eyeball information. We use multitask learning to improve system robustness by estimating gaze direction, eyeball locations and orientations together.

Remounting challenge. A major challenge for mmWave based eye tracking is remounting the glasses, i.e., taking the glasses off and then putting them back on. This causes small shifts in the relative position between the radars and the eyes. Such movements relative to eyes can be over 8 mm, according to our experiments. Due to the sensitivity of mmWave radars and the complex multipath scattering of mmWave signals, the radar signals for the same gaze direction can be very different after remounting. We define the continuous wearing of glasses without remounting as *one session*. The same remounting challenge exists in prior work on acoustic-based eye tracking [30], which leads to the need of calibration for each session after remounting. We also see similar issues of glasses movement in Cider [39] and photo diodes-based glasses [31]. Next, we first describe the glasses coordinate system and show how we mitigate this issue with multitask learning.

Coordinate system. We define the glasses coordinate system with its origin at the front camera on the glasses as shown in Figure 8. The gaze direction is the direction that users look towards relative to the glasses coordinate system on their head. We can express the gaze direction as an angular vector. It can be converted to the 2D location of the users' gaze on the video frame of front camera for video based applications, such as our demos. This aligns with the needs of context-aware AI agents [4] and is the typical set up for eye tracking glasses [5, 8]. Within the glasses coordinate system, the eyeball location is a 3-element vector describing the offset between the coordinate center and the eyeball. The eyeball orientation is 3-element unit vector describing the eyeball's direction. Rather than enforcing an explicit mapping from eyeball location and orientation to gaze direction, we expect the ML model to learn the relationship implicitly.

Estimating eyeball information. We apply multitask learning [11, 33] to mitigate the influence of remounting. Multitask learning is an approach to improves model generalization by using extra domain information contained in the training data of related tasks. In our case, our primary task is estimating gaze direction, while related

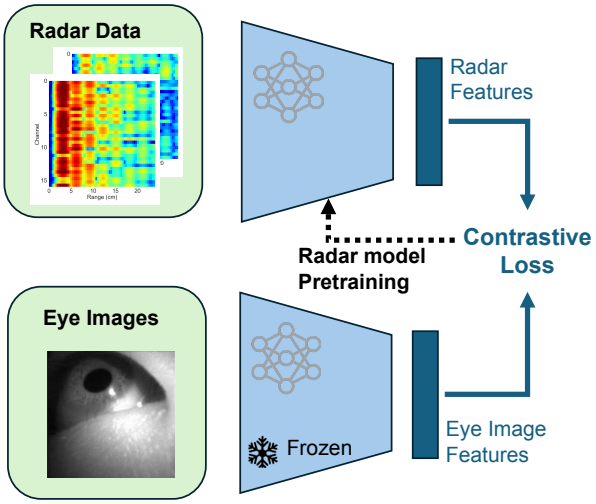


Figure 9: Pretraining the radar model using contrastive learning. We maximize the agreement between the radar features and eye image features using contrastive loss.

auxiliary tasks are estimating eyeball location and orientation. When improving the auxiliary tasks, the primary task benefits from the related information extracted from the data.

Multitask loss function. We optimize all tasks together during training with gradient descent of Adam optimizer. The multitask loss function $\mathcal{L}_{MT}$ follows the following equation:

$$\mathcal{L}_{MT} = \frac{1}{B}\sum_{i}^{B}\left((g_i - \bar{g}_i)^2 + \lambda_{mt}(e_i - \bar{e}_i)^2 + \lambda_{mt}(o_i - \bar{o}_i)^2\right) \quad (4)$$

where B is the number of samples per training batch, g_i, e_i, o_i are the ground truth vectors of gaze direction, eyeball location, orientation, $\bar{g}_i, \bar{e}_i, \bar{o}_i$ are the respective output from the model, λ_{mt} is a hyperparameter. This multitask learning loss guides the ML model to learn the related context about the eye tracking task, allowing it to learn robust radar features that can distinguish gaze direction and eyeball location changes. Thus, across different remounting sessions, we can tolerate the eyeball location change while not affecting gaze direction prediction. We show that mmET accurately estimates eyeball location and orientation in Figure 16, and multitask learning reduces the gaze direction error substantially in Figure 15.

3.4 Pretraining with Contrastive Learning

Following approach described above, we train models based on each user's dataset to have personalized user-dependent models that work well for each user as shown in Figure 13. Due to the difference of each user's eyes, such models cannot work for unseen new users. Eye tracking systems typically use a calibration process to collect data for unseen users. Our next goal is to have a user-adaptive model that can adapt to new users by fine-tuning with a small amount of training/calibration data.

Pretraining for fast adaptation. To achieve fast adaptation, we pretrain a model with multiple users' datasets mixed together (10 users in this paper). By training on diverse user data, the model can learn a set of model weights that extract generalizable features and ignore unrelated user-specific variations. This allows the pretrained model to serve as an effective initialization, requiring only a small amount of fine-tuning data from new users. With limited calibration data, we achieve performance comparable to user-dependent models. As shown in Figure 19, we reduce the data needed by 5×. Ideally, with more diverse users, e.g., over 100 users, we could eliminate the need for calibration for new users as new users would fall within the training distribution.

Cross-modal contrastive learning. To boost the effectiveness of pretraining, we use cross-modal contrastive learning, guiding the radar model with an eye image based model during pretraining stage. Our insight is that pretrained eye image models generalize more easily due to the higher pixel density in images. Although radar data frames have fewer channels and lower resolution, similar information can be buried within, not perceptible to human eyes. We use contrastive learning [13], inspired by recent work on automotive detection [18], to help the radar model learn generalizable feature vectors similar to those of the eye image model.

As shown in Figure 9, we prepare a batch of corresponding radar data and eye images pairs, feeding them into radar and eye image models respectively, and maximize the agreement between radar features and eye image features by minimizing contrastive loss. We train the eye image model separately in advance and freeze its weights during radar model pretraining stage to make sure the eye image features doesn't change and knowledge is distilled from the vision model into the radar model. Contrastive learning aligns the distributions of projected radar features and eye image features, instead of forcing exact values of the features. If we use mean square loss to directly force features to match, the pretrained model performs badly since the radar data and eye images are inherently different.

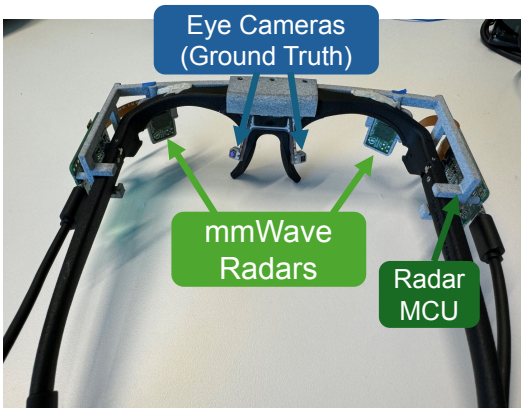
Contrastive loss. For each training iteration, a batch of data pairs flow through the networks and output the encoded feature vectors, h_r and h_v, for radar and vision (eye image) respectively. Then, we map the feature vectors to a hyperspace with non-linear project heads (2-layer MLP), $p_r(\cdot)$ and $p_v(\cdot)$. Thus, we have the projected vectors $z_r := p_r(h_r)$ and $z_v := p_v(h_v)$ and calculate the contrastive loss as follow:

$$\mathcal{L}_{CL} = -\frac{1}{B}\sum_{i}^{B}\log\frac{\exp\left(\text{sim}\left(z_{r,i}, z_{v,i}\right)\right)}{\sum_{j=0}^{B}\exp\left(\text{sim}\left(z_{r,i}, z_{v,j}\right)\right)} \quad (5)$$

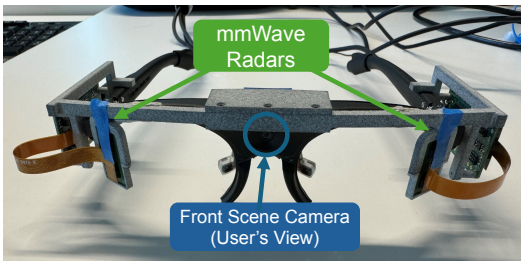
where B is the batch size, $\text{sim}(x, y) := x^\top y/\tau$ is the similarity function and τ is temperature – a hyperparameter that controls the strength of penalties on hard negative samples. By minimizing the loss, the contrastive loss encourages similarity between corresponding radar-vision vector pairs and discourages similarity between mismatched pairs.

Contrastive learning is a self-supervised learning approach which means it does not use ground truth labels. For our system, we have the ground truth gaze direction which can be used to guide pretraining process together. Thus, we combine contrastive loss with the multitask learning loss in Equation 4 to have the final loss function for radar pretraining:

$$\mathcal{L} = \mathcal{L}_{MT} + \lambda_{CL}\mathcal{L}_{CL} \quad (6)$$



(a) Back View



(b) Front View

Figure 10: Eye tracking glasses prototype.

where λ_{CL} is a hyperparameter to control the ratio between two kinds of loss, which we set as 0.5 empirically.

4 Implementation

Glasses with mmWave radars. Figure 10 shows our eye tracking glasses prototype for system evaluation and demos. We use eye cameras to collect ground truth for mmET training and evaluation, which provides gaze direction, eyeball locations, and eyeball orientations via a black-box computer vision model with approximately 1° accuracy. We mount 2 Infineon BGT60ATR24C radars [2] on the left and right corners of the glasses. We use the antenna-in-package version of radar chips, i.e., antennas integrated with mmWave RF chip, and rotate the radar arrays to orientate towards the center of users' eyes. The radars, small and positioned at the corners, cause negligible obstruction to the user's view. This is further confirmed through user study participants, who reported no noticeable blockage of their field of view. Two infineon radar MCU boards are mounted on the legs of glasses to collected radar signals and send them to computer with cables. We configure the radars to operate from 58 GHz to 63 GHz, with 1 MHz sampling rate, 4 chirps per frame, and 64 samples per chirp. Radar FPS is set to 100 for training data collection and 30 for ML model inference. For future work, we aim to improve system integration and use transparent antennas on lenses.

Other prototype designs. For early system development, we test mounting mmWave radars on an unmodified prescription lens and achieve comparable performance. This shows that our approach is applicable to other hardware setups. We report the results of the final prototype as we collected data from most users with it.

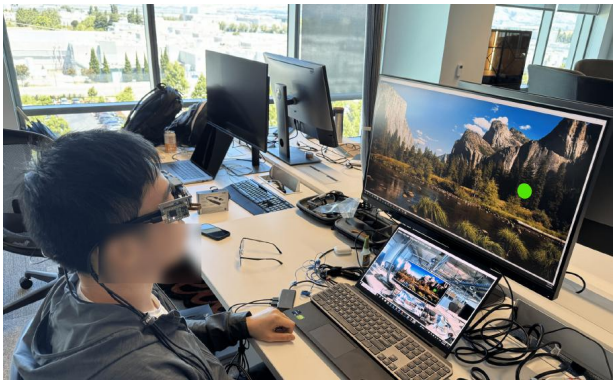


Figure 11: User study setup.

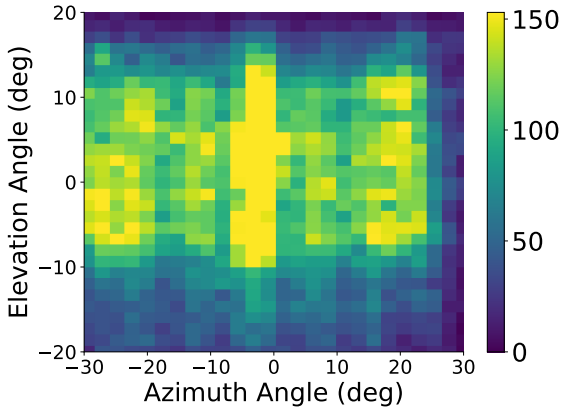
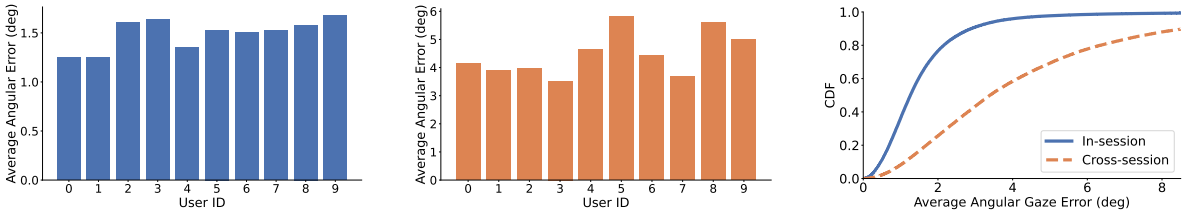


Figure 12: Spatial distribution heatmap of gaze direction in our dataset.

Model implementation and training. We implement our model using PyTorch and train it with the Adam optimizer (learning rate: 0.0001) on an Nvidia V100 GPU. The input data is a tensor of shape (batch_size, 2, 16, 64), where 2 represents the radar data frame and inter-frame difference, 16 is the number of virtual antennas, and 64 is the number of data samples per antenna. A custom beamforming layer following Equation 3 is applied first, with trainable weights updated via gradient descent. We then use a modified ResNet-18 [19] as the feature extractor, adjusting the first layer for radar input and the last layer to produce a 512-dimensional feature vector. This vector is fed into three single-layer fully connected layers to predict eyeball location, orientation, and gaze direction respectively. We use a batch size of 128 for pretraining and 32 for both user-dependent training and fine-tuning. Training and fine-tuning rely solely on radar data and gaze ground truth, while contrastive learning based pretraining also uses eye images.



(a) In-session accuracy **(b) Cross-session accuracy (remounting)** **(c) CDF of angular gaze error**

Figure 13: Gaze direction accuracy for personalized models. Our personalized (user-dependent) models for 10 users show an average of 1.49° error for in-session and 4.47° for cross-session testing;

5 Evaluation

In this section, we first describe our user study details, including data collection procedure. Then, we evaluate the performance of user-dependent personalized models under the impact of different setups and the performance when adapting our pretrained model to unseen new users.

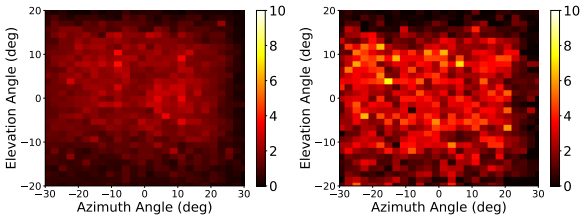
5.1 Experiment Setup

Our data collection and research project have been reviewed and approved by institutional review board, i.e., IRB approval.

User study. We recruited 16 participants, 4 female and 12 male, with age ranging from 24 to 50. During the study, the user sits in a chair and wears the glasses prototype, while looking at the display for instructions, as shown in Figure 11. The distance between the user and the display is around 50 cm and sitting position is adjusted slightly per user to make sure the display fills most of their field of view. Then, we instruct the users to look at the green guiding marker on the screen and follow the marker as it moves across the display area. Each user performs 24 sessions of data collection, 50 seconds for each session, i.e., 20 minutes data collected per user. Within each session, the gaze guiding marker shows in 49 different locations, 1 second each. After each session, we ask the user to take off the glasses and put it back on, i.e., remounting the glasses. With these remounting sessions, we can collect data with different glasses wearing positions and evaluate the impact of remounting on system performance.

Dataset Distribution. Our dataset has a continuous and mostly uniform distribution of gaze direction for training and testing, as shown in Figure 12. The collected data cover a horizontal angle range from -30 to 30, and a vertical angle range from -20 to 20. To reflect practical usage, we do not restrict the head position and orientation during experiments. While guiding markers are placed at fixed locations, natural head movements introduce minor variations in gaze direction. For all sessions together, the data distribution heatmap shows a concentration of data samples at the center, with fewer samples at the edges.

Evaluation metrics. We use the average gaze angular error, i.e., the angle between the radar-based and camera-based gaze direction output, as the evaluation metric for gaze direction accuracy, which is the same metric used by prior research work [30] and commercial eye tracking glasses [5]. By default, we show the performance of user-dependent personalized models trained and tested individually for 10 different users. We then train a general pretrained model with



(a) In-session **(b) Cross-session**

Figure 14: Error distribution heatmaps over the field of view.

10 users' data together and show the performance when adapting to the rest 6 new users individually in subsection 5.3. For 24 sessions of each user, we randomly select 21 sessions and divide the data into training, validation and testing uniformly, which gives *in-session testing* performance (no unseen remounting sessions for testing). The rest 3 sessions are not used for training and only used to test *cross-session* performance, which shows the performance after the user remounts the glasses. Different from prior work [30], we don't perform any short calibration training for cross-session performance testing, which means models are not calibrated/fine-tuned each time user remounts the glasses.

5.2 Eye Tracking Performance

Gaze direction accuracy. To evaluate mmET eye tracking accuracy, we train and test personalized user-dependent models for users individually. We show the average gaze direction angular errors of each user in Figure 13. Across all users, mmET achieves an average error of 1.49° for in-session performance and 4.47° for cross-session performance, a median error of 1.53° for in-session performance and 4.29° for cross-session performance. The cross-session accuracy is lower than in-session due to the relative position shift between glasses and face. To understand the error distribution of gaze distribution angular, we combine all results from 10 users and show the CDF plot in Figure 13c. Across all samples from 10 users, mmET achieves a median angular error of 1.26° for in-session and 3.39° for cross-session, a 90-percentile angular error of 2.89° for in-session and 8.65° for cross-session.

Error heatmaps. We show the gaze direction error distribution over the 2D field-of-view of users as spatial heatmaps in Figure 14. mmET provides uniformly distributed errors across all regions of the

	GazeTrak	mmET
Per-user data duration	40 mins	20 mins
In-session error	3.6°	1.5°
Cross-session error	5.9°	4.7°
New user error	6.7°	5.7°
Per-session calibration	Yes	No

Table 2: Comparison table with GazeTrak.

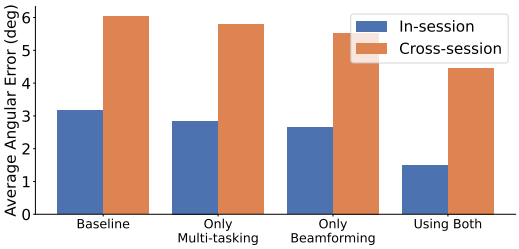


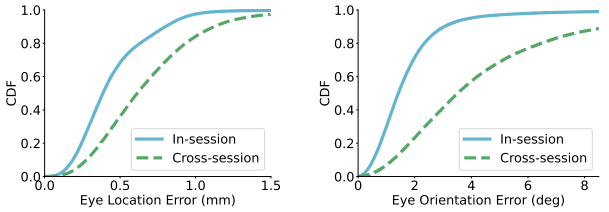
Figure 15: Ablation study. We use both multitask learning and beamforming for the best accuracy, reducing the average error by 1.6 degrees.

field-of-view, which ensures that no abnormal region may experience high error and fail unexpectedly.

Comparison to GazeTrak. GazeTrak [30] is the first acoustic based eye tracking glasses, which is the closed non-camera-based system to mmET. GazeTrak present promising results and capabilities, while mmET achieves lower error with only half of the amount of training data, as shown in Table 2. Moreover, we achieved 1° lower average angular error for new users' cross-session (remounting) testing *without needing per-session short calibration*, which significantly improves user experience.

Effectiveness of beamforming and multitask learning. We verify the effectiveness of beamforming and multitask learning in our system design by performing an ablation study (Figure 15). First, we remove both beamforming and multitask learning from our system and train a bare metal baseline model for each user, which shows 3.17° and 6.03° for average in-session and cross-session error respectively. Then, we train a model with only multitask learning and another model with only beamforming. Both models show around 0.5° error reduction compared with the baseline model. Lastly, we compare the baseline model with mmET model that uses beamforming and multitask learning together. mmET shows around 1.5° average error reduction compared with the baseline model. Thus, we show beamforming and multitask learning work together to reduce gaze direction estimation error effectively.

Accuracy of eyeball location and orientation. As we use multitask learning to estimate eyeball information together with gaze direction, we evaluate the accuracy of estimated eyeball location and orientation. For eyeball location, according to our measurements, the location offset distance is up to 8 mm when the user remounts the glasses. As shown in Figure 16, mmET accurately estimates the eyeball locations with a median error of only 0.3 mm and 0.6 mm for in-session and cross-session tests respectively. For



(a) Eyeball Location **(b) Eyeball Orientation**

Figure 16: Accuracy of (a) eyeball location and (b) orientation estimation using multitask learning.

eyeball orientation, the accuracy is similar to gaze direction accuracy. mmET provides a median error of 1.42° and 3.52° for in-session and cross-session tests respectively. These results show the feasibility and accuracy of using mmWave radars to estimate the eyeball information and also validate the proper functioning of multitask learning.

Impact of antenna number. The number of (virtual) antennas on the mmWave radar fundamentally decides the spatial sensing resolution, and consequently the eye tracking accuracy. A larger number of antennas leads to more input data channels and better accuracy, but also means the size of radar is larger and power consumption is higher. To understand the quantitative relation, we reduce the number of antennas (8 per radar for mmET) virtually by ignoring corresponding radar signals from the data. For example, to test the performance of radars with 2 TX antennas and 2 RX antennas, i.e., 4 virtual antennas, we ignore the rest 4 antenna channels of each radar during training and testing as if they do not exist. We vary the number of antennas per radar from 8 (2TX 4RX) to 6 (2TX 3RX), 4 (2TX 2RX), 3(1TX 3RX), 2(1TX 2RX), 1 (1TX 1RX) and train models for 10 users and evaluate the in-session and cross-session performance. We show the average angular gaze error for 10 users in Figure 17 and the error bars show the standard deviations among 10 users for each setup. Both the average error and standard deviation decrease as the number of antennas increases. We see a significant improvement from 3 to 4 antennas because the antenna array becomes a 2D array from a 1D linear array. Moreover, the gain when increasing antennas from 4 to 8 is small, which implies that mmWave radars with 4 antennas can provide similar performance with a smaller hardware size. This provides valuable lessons for future system development.

Impact of myopia. Many of user study participants have various extent of myopia, ranging from 0 to -7 diopters. We divide the users into two groups, one with low myopia, from 0 and -3.0, and another group with high myopia above -3. There are no evident difference when comparing the eye tracking performance between two groups. As mmET rely on the mix of reflections from eyeball, eyelids, surrounding skin, the different on myopia diopters does not affect the system performance.

Impact of head movements. To verify the eye tracking performance in scenarios that are different from data collection setup, we instruct 6 user study participants to remount the glasses, move their heads freely and look at random daily objects around them. The eye tracking accuracy is stable and similar to fore-mentioned

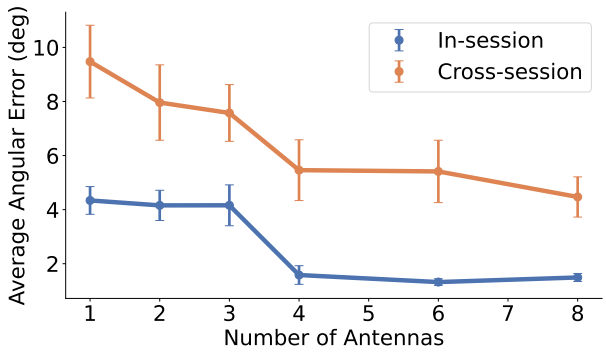


Figure 17: Impact of antenna number. With more antennas on each radar, we achieve a lower angular gaze direction error.

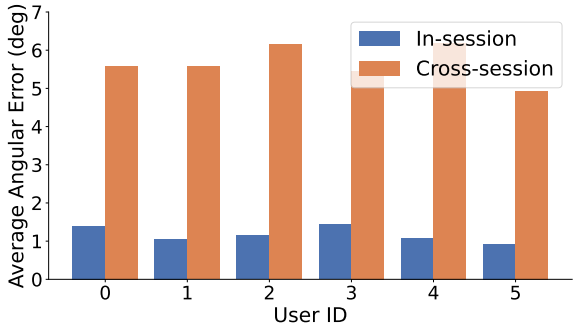


Figure 18: Our pretrained model can perform well for new users with only 5 mins calibration.

evaluation results. We visualize the gaze direction estimation results from mmET in the demo video, which shows the cross-session eye tracking accuracy.

5.3 Adaptation for New Users

Accuracy with a pretrained model. In this section, we train a general pretrained model with 10 users' data together and show the performance when adapting to the rest 6 new users individually. We expect to have a one-time calibration process to collect data for such fine-tuning, and the pretrained model help minimize the duration of calibration. For each new user, we randomly select 6 sessions (5 mins) as training data to fine tune the pretrained model for the specific new user. Figure 18 shows the in-session and cross-session accuracy for 6 new users. mmET achieves an average angular error of 1.12° for in-session tests and 5.57° for cross-session tests.

Reducing training data needed. The main advantage of using the pretrained model is that we can fine tune the model and achieve similar performance with much less training data for the new users, so that new users can undergo a much shorter data collection procedure before using the system. To quantify the amount of training data reduction, we compare the performance of fine-tuning a pretrained model versus training a model from scratch with various amount of training data. As shown in Figure 19, the non-pretraind model achieves lower error with more training data given, but the

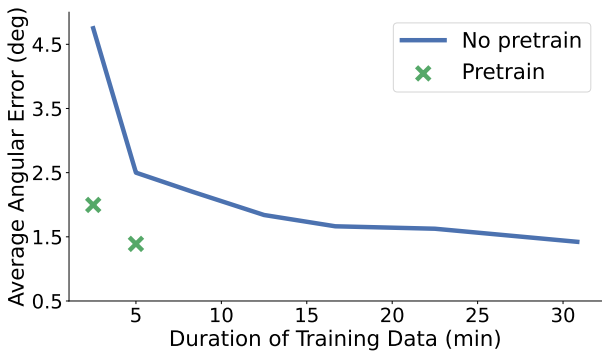


Figure 19: Accuracy versus the training data duration. The pretrained Model can reduce the training data needed by 80%.

	No CL	With CL	Improvement
In-session Error	1.96°	1.17°	0.8°
Cross-session Error	6.13°	5.45°	0.5°

Table 3: Effectiveness of contrastive learning. Contrastive learning reduces the average error when adapting to new users.

pretrained model can achieve a low error with 5 mins or less training data. For achieving an accuracy of ~ 1.4°, we can reduce the duration of needed training data significantly, from 25 mins to 5 mins, with the pretrained model. We use one user for evaluation here as we need over 30 mins training data to capture the trend, which is only available for one test user. We show the benefit of contrastive learning based pretraining for multiple users in Table 3 in terms of accuracy improvement under the same amount of calibration data. Given the limited number of pretrained users, we cannot eliminate the calibration/training for new users completely. We expect that with more users, the pretrained model can be more universal and needs almost no calibration. Note that for state-of-art eye tracking cameras or glasses in general, they still need calibration for each use to achieve the best performance.

Effectiveness of contrastive learning. mmET uses contrastive learning (CL) during the pretraining process to improve generalizability. To verify the effectiveness, we compare the average accuracy across 6 new users after fine-tuning two different pretrained models, one without contrastive learning and one with contrastive learning. As shown in Table 3, using contrastive learning reduces the average error by 0.8° and 0.5° for in-session and cross-session respectively.

5.4 Power Consumption

We use the official tool from Infineon to measure mmWave radar sensor board power consumption and show the results in Table 4. With 30 FPS, each radar consumes only 3 mW power, lower than the microphone and speakers in GazeTrak [30] and much lower than eye cameras (>100 mW). The measurement is based on the electric current reading from the mmWave front end. We focus on the power consumption of sensors only, i.e., mmWave radar board here. If mmWave eye tracking is integrated as part of a smart glasses or AR glasses, low-power ASIC or SoC should handle the baseband

FPS	30	50	100
Power (mW)	3.04	4.88	9.40

Table 4: Power consumption of one mmWave radar under different FPS.

processing and ML model inference for mmWave radars and various other sensors together, which is out of the scope of this work.

6 Limitations and Future Work

Generalizability. We have not yet develop models that generalize to unseen users without calibration and provide accurate gaze direction. This is within expectation given the limited number of users we have for model pretraining. Using eye image based eye tracking models as reference, models that can generalize for unseen new users typically need hundreds of users' data to effectively cover the diversity of population. For mmET, we need to scale up the data collection with 100+ users to improve its generalizability with larger datasets for pretraining. We leave this as future work.

Robustness. Although we tested the impact of head movements, we have not extensively verified the performance during large-scale user activity, such as jumping or running. This is largely decided by the mechanical design, i.e., whether the glasses can fix on the user's face in a stable way. Commercial camera-based eye tracking glasses are also susceptible to such movements and provides adjustable nose pads and head strips to solve this, which are also applicable to our system.

Hardware prototyping. Our mmWave radars are currently mounted on the glasses externally. A more integrated solution is to embed the radars within the glasses frames, which is feasible given the small size of antenna array (8 mm). As we analyzed the impact of antenna number on accuracy (Figure 17), we can potentially even further reduce the number of antennas. Another solution is to place the mmWave antennas on the optical lens with transparent antenna technology, which is our future work.

7 Conclusion

Accurate and robust eye tracking, with a small hardware size and low power consumption, is a critical component to enable gaze-based user interface and context-aware AI on emerging AR/smart glasses. We present mmET, the first mmWave radar-based eye tracking system with a glasses form factor. Using the FMCW signals reflected from eyes, mmET accurately estimates the gaze direction with tailored signal processing and a multitask ML model. We further use cross-modal contrastive learning to pre-train a model that adapts to new users with short one-time calibration. Our work advances mmWave radar-based sensing on mobile devices, offering a robust, compact, and low-power solution for eye tracking.

Acknowledgments

We thank Wenjun Hu for insightful comments and all user study participants for their efforts. We also thank the anonymous reviewers and our shepherd for their valuable comments and suggestions.

References

[1] Apple Vision Pro Specs. https://www.apple.com/apple-vision-pro/specs/. Accessed: 2024-08-30.
[2] Infineon BGT60ATR24C: 60GHz radar sensor for automotive sensing. https://www.infineon.com/cms/en/product/sensor/radar-sensors/radar-sensors-for-automotive/60ghz-radar/bgt60atr24c. Accessed: 2024-09-15.
[3] Meta Quest Pro Eye Tracking. https://www.meta.com/help/quest/articles/getting-started/getting-started-with-quest-pro/eye-tracking/. Accessed: 2024-08-30.
[4] Meta's Ray-Ban Glasses Added AI. https://www.cnet.com/tech/computing/metas-ray-ban-glasses-added-ai-that-can-see-what-youre-seeing/. Accessed: 2024-09-15.
[5] Pupil Lab Neon Eye Tracking Glasses. https://pupil-labs.com/products/neon. Accessed: 2024-08-30.
[6] Qualcomm Snapdragon XR2 Gen 2 Platform. https://www.qualcomm.com/products/mobile/snapdragon/xr-vr-ar/snapdragon-xr2-gen-2-platform. Accessed: 2025-02-01.
[7] Smart Glasses. https://www.meta.com/smart-glasses/. Accessed: 2024-08-30.
[8] Tobii Pro Glasses 3. https://www.tobii.com/products/eye-trackers/wearables/tobii-pro-glasses-3/. Accessed: 2024-09-15.
[9] IEEE Standard for Safety Levels with Respect to Human Exposure to Radio Frequency Electromagnetic Fields, 3 kHz to 300 GHz. *IEEE Std C95.1-2005 (Revision of IEEE Std C95.1-1991)*, pages 1–238, 2006.
[10] Emanuele Cardillo, Luigi Ferro, Gaia Sapienza, and Changzhi Li. Reliable eye-blinking detection with millimeter-wave radar glasses. *IEEE Transactions on Microwave Theory and Techniques*, 2023.
[11] Rich Caruana. Multitask learning. *Machine learning*, 28:41–75, 1997.
[12] Zhaoxin Chang, Fusang Zhang, Jie Xiong, Weiyan Chen, and Daqing Zhang. Msense: Boosting wireless sensing capability under motion interference. In *Proceedings of the 30th Annual International Conference on Mobile Computing and Networking*, pages 108–123, 2024.
[13] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.
[14] Xingyu Chen and Xinyu Zhang. Rf genesis: Zero-shot generalization of mmwave sensing through simulation-based data synthesis and generative diffusion models. In *Proceedings of the 21st ACM Conference on Embedded Networked Sensor Systems*, pages 28–42, 2023.
[15] Arpan Desai, Cong Danh Bui, Jay Patel, Trushit Upadhyaya, Gangil Byun, and Truong Khang Nguyen. Compact wideband four element optically transparent mimo antenna for mm-wave 5g applications. *Ieee Access*, 8:194206–194217, 2020.
[16] Junfeng Guan, Sohrab Madani, Suraj Jog, Saurabh Gupta, and Haitham Hassanieh. Through fog high-resolution imaging using millimeter wave radar. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11464–11473, 2020.
[17] Unsoo Ha, Salah Assana, and Fadel Adib. Contactless seismocardiography via deep learning radars. In *Proceedings of the 26th annual international conference on mobile computing and networking*, MobiCom '20, pages 1–14, 2020.
[18] Yiduo Hao, Sohrab Madani, Junfeng Guan, Mohammed Alloulah, Saurabh Gupta, and Haitham Hassanieh. Bootstrapping autonomous driving radars with self-supervised learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15012–15023, 2024.
[19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
[20] Jingyang Hu, Hongbo Jiang, Daibo Liu, Zhu Xiao, Schahram Dustdar, Jiangchuan Liu, and Geyong Min. Blinkradar: non-intrusive driver eye-blink detection with uwb radar. In *2022 IEEE 42nd International Conference on Distributed Computing Systems (ICDCS)*, pages 1040–1050. IEEE, 2022.
[21] Irene Wei Huang, Paurakh Rajbhandary, Sam Shiu, John S Ho, Jiang Zhu, Ben Wilson, and Geng Ye. Radar-based heart rate sensing on the smart glasses. *IEEE Microwave and Wireless Technology Letters*, 2024.
[22] Infineon. Health effects of mmWave radiation. https://www.infineon.com/dgdl/Infineon-Health%20Effects%20of%20mmWave%20Radiation-PI-v01_01-EN.pdf. Accessed: 2025-02-01.
[23] Cesar Iovescu and Sandeep Rao. The fundamentals of millimeter wave sensors. *Texas Instruments*, pages 1–8, 2017.
[24] Chengkun Jiang, Junchen Guo, Yuan He, Meng Jin, Shuai Li, and Yunhao Liu. mmvib: micrometer-level vibration measurement with mmwave radar. In *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking*, pages 1–13, 2020.
[25] Zhu Juncen, Jiannong Cao, Yanni Yang, Wei Ren, and Huizi Han. mmdrive: Fine-grained fatigue driving detection using mmwave radar. *ACM Transactions on Internet of Things*, 4(4):1–30, 2023.
[26] Kyle Krafka, Aditya Khosla, Petr Kellnhofer, Harini Kannan, Suchendra Bhandarkar, Wojciech Matusik, and Antonio Torralba. Eye tracking for everyone. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2176–2184, 2016.

[27] Henry A Kues, Salvatore A D'Anna, Robert Osiander, William R Green, and John C Monahan. Absence of ocular effects after either single or repeated exposure to 10 mW/cm2 from a 60 GHz CW source. *Bioelectromagnetics: Journal of the Bioelectromagnetics Society, The Society for Physical Regulation in Biology and Medicine, The European Bioelectromagnetics Association*, 20(8):463–473, 1999.

[28] Mikko Kytö, Barrett Ens, Thammathip Piumsomboon, Gun A Lee, and Mark Billinghurst. Pinpointing: Precise head-and eye-based target selection for augmented reality. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, pages 1–14, 2018.

[29] Haowen Lai, Gaoxiang Luo, Yifei Liu, and Mingmin Zhao. Enabling visual recognition at radio frequency. In *Proceedings of the 30th Annual International Conference on Mobile Computing and Networking*, pages 388–403, 2024.

[30] Ke Li, Ruidong Zhang, Boao Chen, Siyuan Chen, Sicheng Yin, Saif Mahmud, Qikang Liang, François Guimbretière, and Cheng Zhang. Gazetrak: Exploring acoustic-based eye tracking on a glass frame. In *Proceedings of the 30th Annual International Conference on Mobile Computing and Networking*, pages 497–512, 2024.

[31] Tianxing Li, Qiang Liu, and Xia Zhou. Ultra-low power gaze tracking for virtual reality. In *Proceedings of the 15th ACM Conference on Embedded Network Sensor Systems*, pages 1–14, 2017.

[32] Tianxing Li and Xia Zhou. Battery-free eye tracker on glasses. In *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking*, pages 67–82, 2018.

[33] Shikun Liu, Andrew Davison, and Edward Johns. Self-supervised generalisation with meta auxiliary learning. *Advances in Neural Information Processing Systems*, 32, 2019.

[34] Simon P Liversedge and John M Findlay. Saccadic eye movements and cognition. *Trends in cognitive sciences*, 4(1):6–14, 2000.

[35] Lina Ma, Yangtao Ye, Changzhan Gu, and Junfa Mao. High-accuracy contactless detection of eyes' activities based on short-range radar sensing. In *2022 IEEE MTT-S International Microwave Biomedical Conference (IMBioC)*, pages 266–268. IEEE, 2022.

[36] Ruichun Ma, Shicheng Zheng, Hao Pan, Lili Qiu, Xingyu Chen, Liangyu Liu, Yihong Liu, Wenjun Hu, and Ju Ren. AutoMS: Automated Service for mmWave Coverage Optimization using Low-cost Metasurfaces. In *Proceedings of the 30th Annual International Conference on Mobile Computing and Networking*, ACM MobiCom '24, New York, NY, USA, 2024. ACM.

[37] Sohrab Madani, Jayden Guan, Waleed Ahmed, Saurabh Gupta, and Haitham Hassanieh. Radatron: Accurate detection using multi-resolution cascaded mimo radar. In *European Conference on Computer Vision*, pages 160–178. Springer, 2022.

[38] Addison Mayberry, Pan Hu, Benjamin Marlin, Christopher Salthouse, and Deepak Ganesan. ishadow: design of a wearable, real-time mobile gaze tracker. In *Proceedings of the 12th annual international conference on Mobile systems, applications, and services*, pages 82–94, 2014.

[39] Addison Mayberry, Yamin Tun, Pan Hu, Duncan Smith-Freedman, Deepak Ganesan, Benjamin M Marlin, and Christopher Salthouse. Cider: Enabling robustness-power tradeoffs on a computational eyeglass. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*, pages 400–412, 2015.

[40] Nishant Mehrotra, Divyanshu Pandey, Akarsh Prabhakara, Yawen Liu, Swarun Kumar, and Ashutosh Sabharwal. Hydra: Exploiting multi-bounce scattering for beyond-field-of-view mmwave radar. In *Proceedings of the 30th Annual International Conference on Mobile Computing and Networking*, MobiCom '24, pages 1545–1559, 2024.

[41] Raphael Menges, Chandan Kumar, and Steffen Staab. Improving user experience of eye tracking-based interaction: Introspecting and adapting interfaces. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 26(6):1–46, 2019.

[42] Yasuo Morimoto, Sam Shiu, Irene Wei Huang, Eric Fest, Geng Ye, and Jiang Zhu. Optically transparent antenna for smart glasses. *IEEE Open Journal of Antennas and Propagation*, 4:159–167, 2023.

[43] Gillian A O'driscoll, Mark F Lenzenweger, and Philip S Holzman. Antisaccades and smooth pursuit eye tracking and schizotypy. *Archives of general psychiatry*, 55(9):837–843, 1998.

[44] Alexandra Papoutsaki. Scalable webcam eye tracking by learning from user interactions. In *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems*, pages 219–222, 2015.

[45] Alexandra Papoutsaki, James Laskey, and Jeff Huang. Searchgazer: Webcam eye tracking for remote studies of web search. In *Proceedings of the 2017 conference on conference human information interaction and retrieval*, pages 17–26, 2017.

[46] Dario D Salvucci. Inferring intent in eye-based interfaces: tracing eye movements with process models. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, pages 254–261, 1999.

[47] Dario Dino Salvucci. *Mapping eye movements to cognitive processes*. Carnegie Mellon University, 1999.

[48] Emerson Sie, Zikun Liu, and Deepak Vasisht. Batmobility: Towards flying without seeing for autonomous drones. In *Proceedings of the 29th Annual International Conference on Mobile Computing and Networking*, pages 1–16, 2023.

[49] Tao Sun, Yankai Zhao, Wentao Xie, Jiao Li, Yongyu Ma, and Jin Zhang. Eyegesener: Eye gesture listener for smart glasses interaction using acoustic sensing. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 8(3):1–28, 2024.

[50] Mélodie Vidal, Jayson Turner, Andreas Bulling, and Hans Gellersen. Wearable eye tracking for mental health monitoring. *Computer Communications*, 35(11):1306–1311, 2012.

[51] Timothy Woodford, Kun Qian, and Xinyu Zhang. Metasight: High-resolution nlos radar with efficient metasurface encoding. In *Proceedings of the 21st ACM Conference on Embedded Networked Sensor Systems*, pages 308–321, 2023.

[52] Ting Wu, Theodore S Rappaport, and Christopher M Collins. Safe for generations to come: Considerations of safety for millimeter waves in wireless communications. *IEEE microwave magazine*, 16(2):65–84, 2015.

[53] Zhefan Ye, Yin Li, Alireza Fathi, Yi Han, Agata Rozga, Gregory D Abowd, and James M Rehg. Detecting eye contact using wearable eye-tracking glasses. In *Proceedings of the 2012 ACM conference on ubiquitous computing*, pages 699–704, 2012.

[54] Xi Zhang, Yu Zhang, Zhenguo Shi, and Tao Gu. mmfer: Millimetre-wave radar based facial expression recognition for multimedia iot applications. In *Proceedings of the 29th Annual International Conference on Mobile Computing and Networking*, pages 1–15, 2023.

[55] Xucong Zhang, Seonwook Park, Thabo Beeler, Derek Bradley, Siyu Tang, and Otmar Hilliges. Eth-xgaze: A large scale dataset for gaze estimation under extreme head pose and gaze variation. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part V 16*, pages 365–381. Springer, 2020.

[56] Zijing Zhang and Edwin C Kan. Radiooculogram (rog) for eye movement sensing with eyes closed. In *2022 IEEE Sensors*, pages 1–4. IEEE, 2022.

[57] Minghe Zhu, Tsung-Hui Chang, and Mingyi Hong. Learning to beamform in heterogeneous massive mimo networks. *IEEE Transactions on Wireless Communications*, 22(7):4901–4915, 2022.