# *AO-Finger*: Hands-free Fine-grained Finger Gesture Recognition via Acoustic-Optic Sensor Fusing

Chenhan Xu
Snap Research
New York, NY, USA
University at Buffalo, SUNY
Buffalo, NY, USA
chenhanx@buffalo.edu

Bing Zhou*
Snap Research
New York, NY, USA
bzhou@snapchat.com

Gurunandan Krishnan
Snap Research
New York, NY, USA
guru@snap.com

Shree Nayar
Snap Research
New York, NY, USA
snayar@snap.com

## ABSTRACT

Finger gesture recognition is gaining great research interest for wearable device interactions such as smartwatches and AR/VR headsets. In this paper, we propose a hands-free fine-grained finger gesture recognition system AO-Finger based on acoustic-optic sensor fusing. Specifically, we design a wristband with a modified stethoscope microphone and two high-speed optic motion sensors to capture signals generated from finger movements. We propose a set of natural, inconspicuous and effortless micro finger gestures that can be reliably detected from the complementary signals from both sensors. We design a multi-modal CNN-Transformer model for fast gesture recognition (flick/pinch/tap), and a finger swipe contact detection model to enable fine-grained swipe gesture tracking. We built a prototype which achieves an overall accuracy of 94.83% in detecting fast gestures and enables fine-grained continuous swipe gestures tracking. AO-Finger is practical for use as a wearable device and ready to be integrated into existing wrist-worn devices such as smartwatches.

## CCS CONCEPTS

• **Human-centered computing → Ubiquitous and mobile devices**; *Virtual reality*; **Gestural input**; **Interaction devices**.

**ACM Reference Format:**
Chenhan Xu, Bing Zhou, Gurunandan Krishnan, and Shree Nayar. 2023. *AO-Finger*: Hands-free Fine-grained Finger Gesture Recognition via Acoustic-Optic Sensor Fusing. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems (CHI '23), April 23–28, 2023, Hamburg, Germany.* ACM, New York, NY, USA, 14 pages. https://doi.org/10.1145/3544548.3581264

## 1 INTRODUCTION

Recent years have witnessed the explosion of Extended Reality (i.e., AR, VR, and MR) applications that can dramatically improve productivity and user experiences in many scenarios (e.g., remote collaboration [33], design [35], entertainment [27], etc.). The XR market size is projected to grow by 300% and reach $98 billion by 2025 [18]. While the interactable space is extended freely to the reachable range, the XR interaction is still limited and requires a large effort from users. Users need to lift their hands to reach either a small 2D on-device touch surface [10] or a limited Line-of-Sight space covered by the Field-of-View of the head-mount tracking cameras [31]. VR headsets [22] usually come with hand-held controllers for interaction which are not hands-free and require the users to press buttons without seeing them.

To provide natural and low-effort XR interactions, studies proposed wrist-worn solutions, which can follow hands continuously without hand grasping, to capture gesture interactions. With the advances in sensing technologies, many gesture-related modalities were proposed, which can be categorized into two types. The type I solutions are to infer finger gestures by directly sensing part of the hand from wrist (e.g., palm [38] and back of hands [36, 39]). The modalities (e.g., cameras and IR sensors) used in this type of solution recognize gestures by directly sensing a fraction of hand within the field of view (FoV). However, the most expressive part of the hands −finger tips− are usually not visible to such wrist-worn sensors. To achieve larger FoV, such sensors need to be lifted off the wrist which are uncomfortable to wear [36, 39] or multiple cameras are required which incurs heavy computation overhead and large latency [13]. The type II solutions infer finger gestures by capturing gesture-induced physiological signal on wrist, including skin deformation [29], ultrasound image [21], muscle electrical activities [16], vibrations [37], etc. The modalities in type II solutions capture indirect signals from finger gestures which make the recognition task more challenging. Although such sensors do not have line-of-sight requirement, they are usually prone to disturbances due to the low Signal to Noise ratio. Type II solutions are usually limited to discrete gestures with no fine-grained gesture tracking.

In this paper, we take an exploratory step and unveil the opportunity of consolidating the two types of modalities for a unified hands-free fine-grained gesture sensing system. The wrist anatomy

[*] Corresponding author

*CHI '23, April 23–28, 2023, Hamburg, Germany*

ACM ISBN 978-1-4503-9421-5/23/04…$15.00
https://doi.org/10.1145/3544548.3581264

reveals that finger gestures are controlled by the coordination of hand muscles and tendons going through wrists [21], thus finger gestures create both visual signals that can be observed by optic sensor and internal signals that can be sensed from the wrist. Given this observation, we propose a novel fine-grained gesture recognition system *AO-Finger*, which is built upon high-speed low-power optic motion sensors (i.e., type I modality) and modified stethoscope microphone (i.e., type II modality). Our optic sensor has a low profile and only observes a small thumb area (as shown in Figure 2) such that it can be built into a wristband in a compact design. The modified stethoscope microphone is sensitive to sound signals conducted from finger tips through tendons. By taking advantages of the complementary strength of both sensing modalities, *AO-Finger* enables hands-free, fine-grained finger gestures detection with a compact wrist-worn device. This facilitates multiple important advantages for gesture recognition in XR applications: 1) *Hands-free*: *AO-Finger* is wrist-worn system that releases both hands for other tasks and interactions in XR applications. 2) *Inconspicuousness and effortless*: unlike existing solutions which require to the users to lift up arms in the air to enable camera tracking or using the touch pad on HMD, we design a set of inconspicuous, effortless micro gestures that are easy to perform with minimum finger movements. 3) *Fine-granularity*: one unique advantage of *AO-Finger* is that it enables continuous thumb swipe tracking, which provides more fine-grained interactions compared to a large set of existing work which only recognize discrete gestures [14, 21, 28, 29, 37].

To achieve hands-free, fine-grained and effortless finger gesture recognition with these advantages, we must address several critical challenges: 1) design a set of gestures that are easy and natural to perform, and fits to *AO-Finger* sensing modalities; 2) build a sensing hardware prototype with strategic sensor mounting to ensure stable signals (e.g., optimal microphone location and contact, optic field of views); 3) analyze the complementary sensing data and develop models and systems that take advantage of their signals organically by sophisticated sensor fusion algorithms design.

Specifically, we make the following contributions:

- We comprehensively investigate existing sensing modalities and identify the combination of acoustic-optic sensor fusing for hands-free, fine-grained finger gesture recognition. To take full advantage of our sensors, we design a set of inconspicuous micro finger gestures that are natural to perform with minimum efforts.
- We design a low-cost, noise-resilient modified stethoscope microphone to capture the gesture-induced acoustic signal within a wide spectrum. We customize two low-energy optic sensors with robustness to ambient light and fine-tuned sensing range for privacy-preserving fine-grained gesture tracking.
- We propose a multi-modal CNN-Transformer model for fast gesture (flick/pinch/tap) detection which outperforms the baseline model significantly with an 15% accuracy increase. We also train a separate finger swipe contact detection model which enables fine-grained continuous swipe gesture tracking.
- We develop a series data augmentation techniques based on physics simulations which overcome the overfitting problem and improves overall accuracy significantly by almost 20%. To further enhance the performance, we propose an aggregation model for robust gesture detection and a system level readiness detection mechanism to suppress false alarms in noisy scenarios.
- We build a prototype with 3D printing and evaluate the user experience in multiple applications including interaction with AR glasses. Results show *AO-Finger* is robust in detecting fast gestures with an overall accuracy of 94.83% and has high usability of fine-grained swipe gesture tracking.

To the best of our knowledge, *AO-Finger* is the first wrist worn system for hands-free, fine-grained inconspicuous micro finger gesture detection based on acoustic-optic sensor fusion. Despite its advanced capabilities, *AO-Finger* achieves a compact low-profile hardware design, which is ready to be integrated into existing wrist worn devices such as smartwatches or used as a standalone input device.

## 2 OVERVIEW

We briefly introduce the finger gesture set we propose, rational behind our sensing modality selection and the high-level system overview.

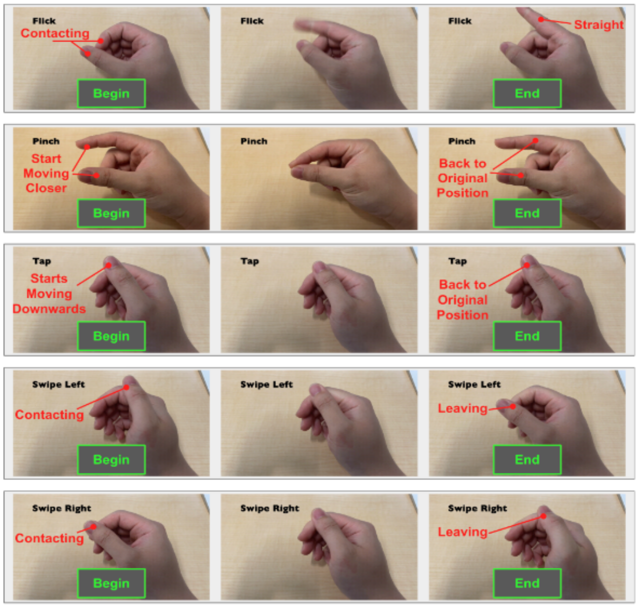


**Figure 1: Gesture set and the start/end definition for data labelling.**

### 2.1 Gesture Set

*2.1.1 Gesture Design Consideration.* During the gesture set design, we have two considerations. First, it is essential to design a set of finger gestures that is easy and natural to remember for most users. As the touch-screen interaction has been widely adopted and accepted, the gesture set that supports a smooth transition from the existing touch-screen interaction method to *AO-Finger*

**Table 1: Comparison of typical wrist worn sensors for finger gesture recognition.**

| Sensor | Wearability | Functionality | Limitations |
|---|---|---|---|
| IR sensor (point to wrist skin) [20] | High | Medium | Low sensor density |
| Camera/IR sensor (point to fingers) [36, 38, 39] | High | High | Power hungry, high latency, require line-of-sight |
| Pressure sensor [7, 28] | Medium | Medium/High | Require firm contact |
| Ultrasound [14] | Low | Medium | Require firm contact, coupling gel |
| EMG sensor [26] | Medium | High | Require good electric contact |
| Magnetic [5] | Low | High | Additional sensors on fingers |
| Microphone [12, 43] | High | Medium | Background noise interference |

is targeted. Therefore, we avoid involving more than two fingers in our gesture design and limit the number of gestures. Second, inconspicuous and fine-grained micro gestures [4] are preferred so users can use *AO-Finger* in most environments with minimum effort. Therefore, we include gestures that users are not required to finish the whole movement to use, i.e., the user's progress of gesture should be mapped to a scale (e.g., slider).

*2.1.2 Gestures.* With these considerations, we design a set of five gestures in *AO-Finger*, i.e., *Flick*, *Pinch*, *Tap*, *Swipe_Left*, and *Swipe_Right*, as shown in Figure 1. Since each gesture is a sequence of finger movements, we clearly define the start and end state of fingers for data labelling. These definitions are critical to ensure gestures can be labelled consistently among all sessions labelled by different users. We divide the gesture set into two categories: *fast gestures* (i.e., flick, pinch and tap) and *fine-grained gestures* (i.e., swipe left and swipe right) which require continuous tracking. Fast gestures involves faster finger movements and create high energy in audio signals, and usually they have shorter duration. Fine-grained gestures like swipes involve continuous movement and usually last longer duration. Continuously tracking the precise swiping finger movements enables fine-grained control in human computer interactions.

## 2.2 Why Acoustic-Optic Fusion?

**Sensing Modalities.** Various sensing modalities have been explored for finger gesture recognition from the wrist. We evaluate the possible modalities from three aspects: wearability, functionality and limitations. Wearability has direct impact on the comfortableness of prolonged wearing and the sensors should have small form factor that can be easily integrated into space constrained wristband. Functionality shows the capability of the sensors in detecting finger gestures from the unique signal features. We investigate each sensing modality and propose sensor fusing design to overcome the limitations.

Table 1 shows the typical sensors that have been explored for wrist-worn finger gesture recognition systems. Wrist worn cameras pointing to the fingers [36, 38, 39] reconstruct 3D hand pose from camera frames of a hand part such as palm and hand back. While these solutions are straightforward, they usually have high latency and consume a lot of power on processing large volume of video data in realtime, making them not suitable as wearable device. Additionally, regular cameras suffer privacy issues. Others like pressure [7, 28], ultrasonic [14] and EMG [26] sensors have lower wearability since they require firm contact against the skin. Microphones have shown the capability of distinguishing finger gestures from the sound signals [12, 43]. They have high wearability due to the low profile form factor with limited functionality due to the poor Signal to Noise Ratio (SNR) and background noise interference.

**Why Sensor Fusion?** When finger gestures are performed, the friction between contact surfaces (e.g., thumb vs. index finger) and internal bone/tendon movements generate sound signals, which conducted through the body and can be picked up by microphones on the wrist. Depending on the type of gestures, the captured sound signal has varied energy and frequency distributions, which creates features for gesture recognition. However, some gestures generate very similar sound signals. For example, tap and pinch gestures are close to each other since they only differ slightly on the contact position: thumb contacts index finger center area *vs.* tip. Swiping thumb against index finger left and right create similar sound signals with weak energy which makes it very challenging to distinguish from purely audio signals. Additionally, sound signals are known for poor SNR due to noises from outside environment as well as internal body conduction. In order to complement these limitations, we seek an additional sensing modality in our design.

We choose optic motion sensors as complementary sensing modality in *AO-Finger* for sensing the proposed gesture set. By strategically placement on the wrist, optic sensor is capable on tracking thumb movements (e.g., raising up/down and swiping left/right), which helps resolving the ambiguities between tap *vs.* pinch and swipe directions. Flick is so unique in acoustic signal with high energy, high frequency and sharp duration that it can be reliably detected from sound signals. Through acoustic-optic sensor fusion, *AO-Finger* captures sufficient sensor signals to recognize our defined gesture set reliably with little ambiguity. Additionally, leveraging the object tracking capability of optic sensors, *AO-Finger* enables fine-grained continuous thumb swipe gesture tracking for the next level of interactions compared to existing discrete gesture classification solutions.

## 2.3 System Overview

Figure 2 shows the overall system design of *AO-Finger*. It takes two signals as input for finger gesture recognition: dual optic sensors that track the continuous movement of the skin area connecting thumb and wrist (highlighted in Figure 2) and one modified stethoscope microphone in contact to the inner wrist. We process and enhance the signals on the sensing hardware to provide reliable

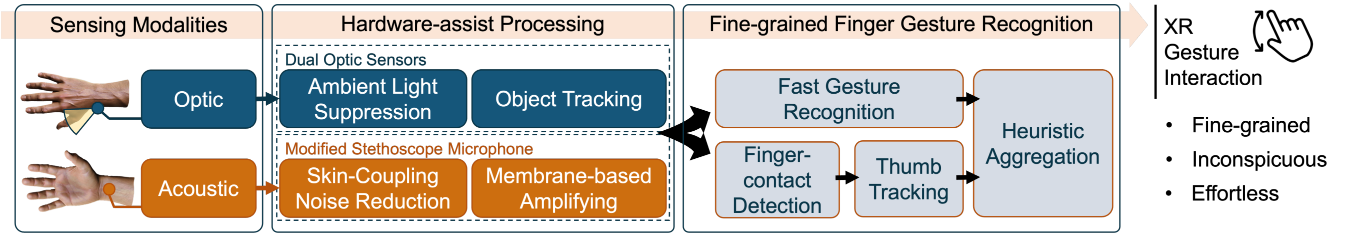


**Figure 2: System overview of *AO-Finger* which contains three major components: sensing modalities, hardware-assist processing and fine-grained finger gesture recognition algorithms.**

signals with high SNR. Specifically, instead of retrieving raw low resolution images (30x30 pixels) from optic sensors to the Micro Controller Unit (MCU) for processing, we leverage the on-chip image processing to enable object tracking and outputs the information of the tracked object only (e.g., position, size and brightness). This enables low power consumption of the system and achieves a high frame rate of 386 fps which is essential for capturing signals from fast finger gestures such as flick, pinch and tap. We strategically placed two optic sensors on the wristband to expand the FoV. We also modify an off-the-shelf MEMS microphone to suppress skin-coupling noise, amplify skin-conducted sound signals and isolate air-conducted noises. Last, we design a set of algorithms for fine-grained finger gesture recognition which consists of a multi-modal CNN-Transformer network for fast gesture recognition, a finger swipe contact detection model (contact between thumb and index finger to trigger swipe gestures), and a heuristic aggregation module to output final gestures. In total, *AO-Finger* enables three fast finger gestures detection and two continuous fine-grained swipe gestures tracking, which are inconspicuous and effortless to perform. *AO-Finger* maintains a lower profile such that it is ready to be integrated into existing wrist-worn devices or use as a standalone input wearable device for AR/VR interactions.

# 3 APPROACH

## 3.1 Hardware and Sensor Signals

To capture the designed gesture set, we design our wristband prototype as shown in Figure 3. The sensors carried by the wristband include two strategically positioned optical sensors (modified PAJ7620U2) for larger FoV and a modified MEMS microphone for skin-coupling noise reduction and signal amplifying. A Bluetooth Low Energy (BLE) controller board (Adafruit Feather nRF52840 Express) is attached to capture the data reported by sensors and communicate with XR devices requiring gesture inputs. The sensors are soldered on soft Printed-Circuits for the flexible placement on wristband. To secure sensor mount and ensure consistent relative sensor positions in repeatable wearing, we designed and 3D-printed a wristband skeleton. The wristband skeleton consists of three pieces on wrist locations, i.e., the volar-wrist piece (VP), the dorsal-wrist piece (DP), and the radial-wrist piece (RP) as shown in Figure 3. The three parts are chained through elastic bands. Wrist anatomy reveals that the volar wrist area is thinner and closer to the tendons that controls finger gestures than the other two locations. Therefore, we place the microphone on the volar-wrist part of the wristband for a better capture of the gesture-induced acoustic signal.

**Dual Optic Sensors.** One of the key challenges to vision based sensing modalities is the disturbance from ambient lights. To get rid of the influence of ambient lights, *AO-Finger*'s optic sensor works in infrared spectrum and carries its own IR LED. Specifically, for each sensing frame, the IR LED first illuminates the sensing area and then dims (5% duty cycle). The illumination area is fine-tuned and limited to 5-15 cm to keep a low energy consumption. The sensor array then takes an IR image (30x30 pixels) when the IR LED is on and off, respectively. Due to the ultra-low resolution and short depth range, our sensors mitigate privacy issues. Subtraction is then performed on the two adjacent shots to remove any possible IR noises from environment and generate a single frame. To detect the object in the view, we apply a threshold on the frame and select the pixels brighter than the threshold as valid pixels followed by a clustering algorithm to find out the largest cluster of bright pixels. We use the center of the cluster to represent the object location and convert the center pixel location to X/Y coordinate values. Note that these image processing procedures execute on highly optimized integrated circuits in the sensor for extreme efficiency and high frame rate. In our settings, we achieve a frame rate of 386 fps for capturing fine-grained finger movements. The optic sensor captures the shroud of thumb finger in the view. We used two optic sensors with slightly different tilting angles to expand the field of view. On average, each optic sensor only consumes 2.82 mA current when operating, including IR LED illumination, imaging, running image processing, and communication. In contrast, a typical camera (OV5647 [24]) designed for IoT applications draws 200 mA (10 times of our optical sensors), excluding any image processing and illumination.

**Modified Stethoscope Microphone.** One important difference between the regular microphone and our skin-contact microphone is the skin-coupling noises due to the capacitance between skin and microphone shell. To prevent radiated disturbances, the COTS MEMS microphone is usually encapsulated into a metal shell. However, given the small form factor of MEMS microphone, the contact area between the shell and skin will be limited and tend to generate capacitance. Consequently, the skin-coupling capacitance can disturb the MEMS microphone (as MEMS Mic leverages the inner capacitance to measure the sound pressure) and the filtering capacitors on FPC. The skin-coupling noise frequency band overlaps that of finger gesture acoustic signals, thereby making it hard to be removed by post signal filtering. To mitigate the noise, we specifically design a z-shape slot on the volar-wrist part for microphone installation (shown in Figure 3). It can be seen that the capacitors of microphone can be hide in z-slot and is well isolated from skin surface. In addition, we apply a thin metallic membrane (aluminium foil) to expand the microphone contact area with skin to mitigate the disturbing capacitance. The metal membrane also brings two important benefits: 1) the metal membrane isolate the air vibration from the environment and only takes the vibration from contacted wrist skin which increase the SNR significantly. Therefore, *AO-Finger* does not require extra noise-cancellation microphone and algorithms[1]. 2) As the opening of microphone shell is covered by the metallic membrane, the microphone works similar as a stethoscope. The acoustic signals from finger gestures is further amplified by the formed metal drum.

**Sensor Data Synchronization.** Intuitively, the controller board can retrieve the sensors' readings by looping to keep all the sensors synchronized. However, the gestures that *AO-Finger* recognizes is usually fast (less than half a second) and the acoustic and optical sensing modalities are of different and high sensing rates. In *AO-Finger*, the controller board access the two optical sensors via I2C in a synchronized way as their sample rates are the same (386fps) and relative low compared with the microphone. As the microphone sends acoustic signals to the controller board via Pulse-density modulation (PDM) at 41667 Hz, we enable asynchronous transmission for acoustic signals utilizing direct memory access with double-buffering. The interruption triggered by buffer full event is minimized to reduce the influence on the loop duration. In addition, we make the controller board send the same amount data in each loop to keep the communication overhead consistent.

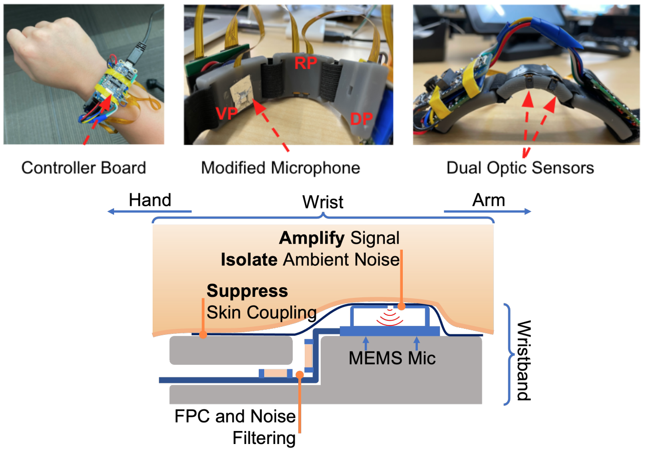


**Figure 3: *AO-Finger*'s prototype consists of a modified stethoscope microphone and dual optic sensors.**

**Signal Pre-processing.** The contact friction between modified microphone and skin may create artifact spikes. To mitigate such artifacts, we first apply a median filter to remove such outliers. After filtering on the time domain, we compute the spectrogram of the audio signal to convert it into both frequency and time domain to capture richer information. We use a time window of 1024 samples (24.6 milliseconds) with an stride of 80 samples for computing the Power Spectrogram Density map. Then we take the log on the signal so that is can be easier to be normalized. We use a min-max scalar to normalize the spectrogram. Note that the min and max are estimated from the whole dataset across all the users. It is critical to make sure not to apply sample level normalization otherwise we lose amplitude information which is critical for flick gesture detection. After the audio signal pre-processing, we get a normalized 2D spectrogram as acoustic input for the model.

Same as the audio signal, we also apply median filter on the optic data to filter out possible spike artifacts. To accommodate varying wearing positions, we subtract the raw X/Y values for each gesture sample window by the mean value. Then we normalize the values by dividing the values by a fixed range, which is the maximum possible moving distance range estimated from all the dataset. This normalization retains the amplitude of the gesture and is agnostic to the wearing position. We take signals from both optic sensors [X1, Y1, X2, Y2] as input to our model.

## 3.2 Gesture Detection Model

It is relatively intuitive to train a single multi-class classifier to classify the 5 gestures and negative class with both acoustic and optic signals as input. Such straightforward design faces multiple limitations: 1) a simple classification model only produces a class label for each performed gesture, which lacks the fine-grained gesture tracking capability; 2) training a multi-modal neural network on certain gestures could lead to overfitting to a single modality, thus the model ignores other sensing modalities. In our experiments, we find that swipe gestures are mainly controlled by optic signals rather than acoustic signal. This causes a major problem if not addressed properly: swiping thumb without touching index finger also triggers gestures, which creates false positives.

Based on the observations, we decompose our problem into two models and then we aggregate the results based on heuristics. Specifically, we train a multi-modal classification model for fast finger gestures only to detect flick, pinch and tap; and another finger swipe contact detection model specifically for detecting the thumb/index finger swipe contact status. If the swipe contact is confirmed, the system enters continuous swipe gesture tracking mode which keeps tracking the thumb movements for fine-grained control.

### 3.2.1 *Fast Gesture Detection.*

We first present our multi-modal sensor fusion model for fast gesture detection.

*Input representations.* We choose the input signals based on the gesture set analysis. Flick gestures usually have shortest duration (as shown in Figure 8) and highest pitch (i.e., frequency and energy), these features can be captured by acoustic signals. Since flick mainly involves index finger movement, it is hardly visible to the optic sensor on the wrist. Pinch gestures create lower pitch sound signals compared to flick due to the "soft" contact of thumb and index finger tips, which also applies to tap gesture. Pinch and tap share similar acoustic features since the major difference is the contact position: tapping requires thumb taps index finger middle part while pinch

[1] We designed dual microphones in hardware with one reserved for background noise cancellation. We decided not to use the second one due to the excellent noise isolation capability brought by metal membrane covering.

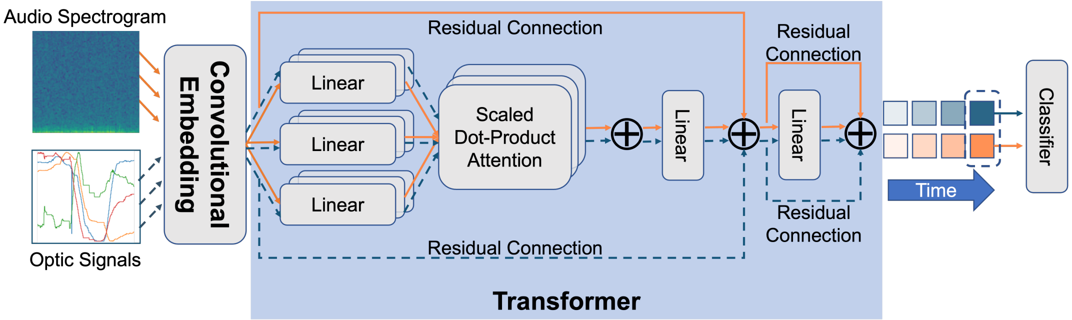


**Figure 4: Multi-modal CNN-Transformer model for fast gesture detection and finger contact detection model.**

requires the thumb to "tap" index finger tip. It is challenging to distinguish pinch from tap due to the similar acoustic features. Thus, we introduce the optic signals. Tap gestures involve thumb moving which can be captured by optic sensors, while pinch mainly involve index finger movements, which is less sensitive to optic sensors. We take signals from both optic sensors and the modified microphone as input, and propose a multi-modal neural network for signal classification.

*Model Architecture.* As our model takes two modalities as input, we design a dual-branch network, one for acoustic signals and one for optic signals. Acoustic data is represented as spectrogram after pre-processing, which inspires us to use 2D CNN as initial embedding extraction layer. We also 1D CNN on optic data as embedding extraction layer. While CNNs have been successful in many computer vision tasks such as image classification [8] and object detection [3], Transformers, which are initially used in Nature Language Processing tasks [32] have shown superior performance in many of these tasks. We use transformer layers an encoder for high level feature extraction and propose a CNN-Transformer fusion model to classify signal sequences to gestures (as shown in Figure 4). After three transformer encoder layers, we only take the features at last time stamp, which are fed into fully connected layers to enrich the features. We concatenate both features from audio and optic, and classify the features with two fully connected layers. The output are the one-hot encoding for the three gestures and none class. We use cross-entropy loss for training the model.

### 3.2.2 Continuous Gesture Detection.

A naive solution for continuous gesture (swipe left and swipe right) detection is to train similar classifier as fast gesture detection model. However, such model can only outputs binary results such as left or right, they are not capable of continuous fine-grained thumb tracking. This is a big sacrifice in user experience (e.g., continuously swiping tracking can be used to adjust volume level smoothly).

As our optic sensor produces raw object movement coordinates in pixels, we can directly leverage such measurements for fine-grained control. The challenge is that without knowing if the thumb is contacting the index finger, the signal will be noisy as optic sensor has no idea whether the thumb is contacting the index finger. Thus, we need to do the contact detection so that we can filter out the movements when the thumb is moving in the air. To achieve this goal, we train another neural network for detecting the contact.

*Finger Contact Detection.* It is important to decide which sensing modality we should use for contact detection. In this case, we choose to use audio signal only without optic signals. When the thumb is rubbing against index finger, the sound can be captured by the microphone which gives us opportunity to identify the contact. There is also information from optic sensor as well such as high Y value since the thumb is usually farther away from the sensor when it's touching the index finger. However, such absolute signal numbers are highly unreliable when the sensor wearing position changes. Naively training a model based on both audio and optic signals can easily overfit the model to optic signals, which turns out to be highly sensitive to wearing positions when testing in real-time.

Contact detection based audio signal is not an easy task as well. In our experiments, we also found that thumb movements in the air without contacting the index finger also generates non-trivial sound signals. Thus, a sophisticated machine leaning model is needed rather than heuristic method. We reuse the similar CNN-Transformer model without the optic branch and modify the output as a binary classifier. We also use cross-entropy loss for training the model.

## 3.3 Prediction Aggregation

The aggregation is conducted via an finite-state machine (FSM). The state transition from the current gesture $i$ to a new gesture $j$ will be triggered when $t_{ij}$-consecutive windows are predicted to be of the gesture $j$. Therefore, we have the gesture transition matrix as $t_{ij}, i, j \in G$, where $G$ is the states set of gestures including the state where no gesture is presented. Considering the gestures are usually mapped to atomic input operation, we disabled the transition between two gestures, i.e., only transitions between the non-gesture state and gesture state are allowed. In this way, the aggregation is controlled by ten parameters.

*Aggregation Optimization.* we use the number of correct gestures as our target function to optimize the ten aggregation parameters. The correctly recognized gesture includes the gestures with whom the recognition is perfectly matched or partially matched. As the solution space is large, we use a random search to approximate the optimal solution. The optimization is performed on dedicated data and the results are reported on unseen data.

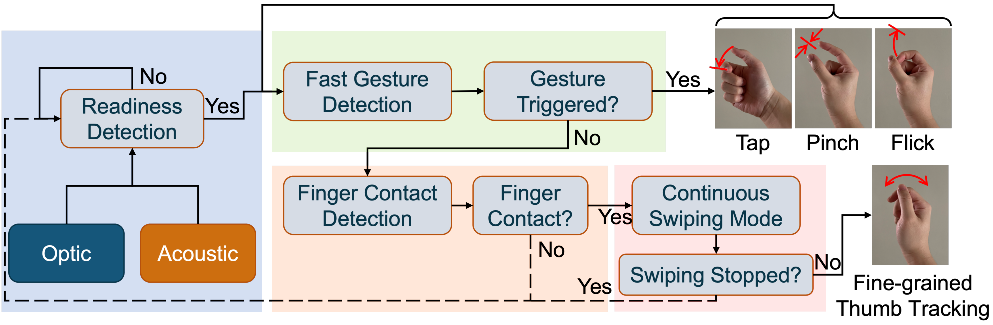


**Figure 5: *AO-Finger* finger gesture recognition logic built upon detection models.**

## 3.4 Putting It Together

To give user an integral experience on gesture recognition, we put all components together and designed the logic shown in Figure 5. The system by default is in *Not Ready* state, where the incoming audio and optic signals are dropped directly without processing. *AO-Finger* uses the signal energy to detect the readiness for gesture recognition. Specifically, if the standard deviation of the optic signal energy in the last 3 seconds is less than 6, *AO-Finger* enters the *ready* state and setup an countdown timer, which brings the system back to *Not Ready* when expires. In addition, once the signal energy become greater than 50, the system will be reset to *Not Ready* immediately. This design *drastically reduces false alarms brought by daily activities* and save battery as well as computation resources for other tasks on the devices. When *AO-Finger* is in Ready state, the acoustic and optic signals are fed into both fast gesture detection and contact detection models, where the fast gesture detection has the higher priority, i.e., once flick, pinch, or tap is detected, the contact detection result is ignored. If no fast gesture is detected and contact is detected, *AO-Finger* enters swiping mode, where the thumb locations is continuously tracked and reported until the the swiping is not detect for 1 second (i.e., exit swiping mode). To further improve the responsiveness for multiple finger gestures in a row, we also configure *AO-Finger* to 1) reset and restart readiness countdown timer each time a fast finger gesture is detected; 2) reset the timer when enter swiping mode and restart the timer when exit swiping mode.

# 4 EXPERIMENT SETUP

## 4.1 Data Collection and Labelling

We collect data from participants using the hardware setup shown in Figure 6(a). We develop a data collection tool which display random gesture video to the users so that they can perform the designated gesture in the time counting down period. In the meantime, we also record a video using webcam of the user's hand gestures which is used as reference for manual data labelling. The data collection tool captures the audio and optic data via USB connection to a laptop. Both the MCU time of sensor data and video frames are saved for data synchronization. We capture two stream of audio data at 41667 Hz and two optic sensors at 368 fps, including the X/Y coordinates.

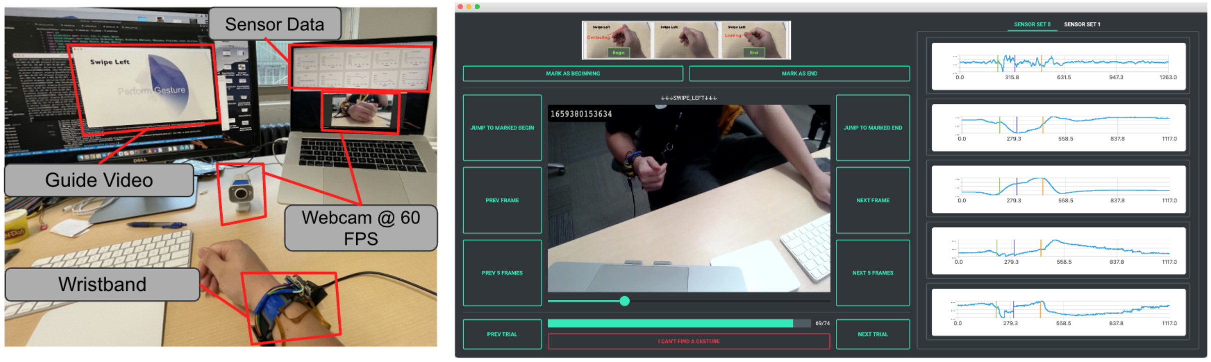
**Dataset Collection.** We invite 20 volunteers for data collection, which includes 17 males and 3 females in the age range of 23-33. Each participant spends around 30 minutes on data collection to receive an incentive gift card. We divide the data collection in multiple sessions and take off/put on the wristband between sessions to make sure the data captures. During these sessions, volunteers change their postures freely (e.g., arms on the desk, under the desk, on the chair arm, bent and straight). Each volunteer contribute 5 sessions of which 4 session are 5-minute sessions for positive samples collection. The last session is designed to capture hard negative samples. In this session, the participant wear the wristband and move the hand/fingers freely by doing daily activities such as typing on a keyboard, using smartphones/mouse, drinking water, etc. We instruct the participant to perform hard negative gestures such as swiping thumbs left and right without touching index fingers, attempt pinch/tap gestures without actually finger contacts. These negative data are critical for us to train a model that is robust to false positives.

**Data Labelling.** We develop a data labelling tool for precise start/end positions labelling on the collected data, as shown in Figure 6(b). The tool displays the video frames on the left and one set of sensor (one microphone and one optic sensor) signal on the right side. To speed up the labelling, the tool automatically generates start/end positions based on heuristics, e.g., audio signal energy, peaks for flick/pinch/tap gestures and optic signal high/low plateaus for swipe gestures. The labelling person only needs to verify the automatically generated labels and make minor adjustments by referencing to the videos.

## 4.2 Training Data Preparation

Given the labeled gesture data collected from our subjects (see Section 4.1), in this part, we describe how the training samples are generated in detail.

One *critical problem* for data preparation is the margin between positive and negative samples. For each labeled gesture period, multiple samples can be generated by moving the sampling window, which is similar to the real-time gesture recognition. An intuition is that samples should be considered as positive when they overlap the gesture periods and the overlapping duration is longer than a threshold. However, a hard threshold brings a problem that the samples close to the threshold are similar. We found these similar samples can severely confuse classifiers.



a). Data collection setup

b). Data labelling tool

**Figure 6: The pipeline for training data collection and labelling. We record the sensor data and RGB videos simultaneously and use the videos as reference for gesture labelling in our data labelling tool.**

**Raw Clips Generation.** We first generate raw gesture clips and then extract positive and negative samples online during training. We label the start and end positions of each performed gesture on the synchronized audio-optic data precisely following the labelling guideline. For each labelled gesture period, we crop a signal segment of 3 seconds with the gesture in the middle of the whole segment.

**Training Samples Generation from Raw Clips.** Take audio signal as example, to generate positive gesture samples, we randomly crop on raw clips with a *sample window* of fixed length $l_w$, which contains $N = l_w \cdot Fs$ audio data points, where $Fs$ is the audio sampling rate. Let's assume the *labeled gesture window* has a start index $t^{label}_{start}$ and end index $t^{label}_{end}$ in raw clip, as highlighted in Figure 7a. For the cropping, we make sure the complete labeled gesture window is fully covered by sample window when the gesture window length is small than $l_w$ (as shown in Figure 7a); and the sample window is fully occupied by gesture window when the gesture window length is larger than $l_w$(as shown in Figure 7b). We determine the starting index of the sample window $t^{crop}_{start}$ as:

$$t^{crop}_{start} = \begin{cases} randint(max(0, t^{label}_{end} - l_w), t^{label}_{start}) & \text{if } l_w > t^{label}_{end} - t^{label}_{start} \\ randint(t^{label}_{start}, t^{label}_{end} - l_w) & \text{if } l_w \leq t^{label}_{end} - t^{label}_{start} \end{cases} \quad (1)$$

where $randint()$ is a function to generate a random index number within the given range. To generate negative samples, we have two resources: 1) The none gesture segments in the raw clips with positive gestures *(Soft Negative Samples)*; and 2) Raw clips of negative samples from the pure negative sessions *(Hard Negative Samples)*. For soft negative samples, we randomly crop a window of length $l_w$ before $t^{label}_{start}$ or after $t^{label}_{end}$; for hard negative samples, we just randomly crop a window of length $l_w$ from the raw clip of negative sessions.

Similarly, we apply the same cropping on optic signals so that both modalities are synchronized. When a raw clip with positive gesture is loaded during training, we use a parameter $P_{negative}$ to control the probability of cropping a soft negative sample vs. positive sample. This helps us to control the percentage of negative samples for a balanced training set. Hard negative samples are in addition to these cropped negative samples. Note that the cropping happens in realtime during training which serves as one of the data augmentation techniques to make the model inference robust to signal time shifts.

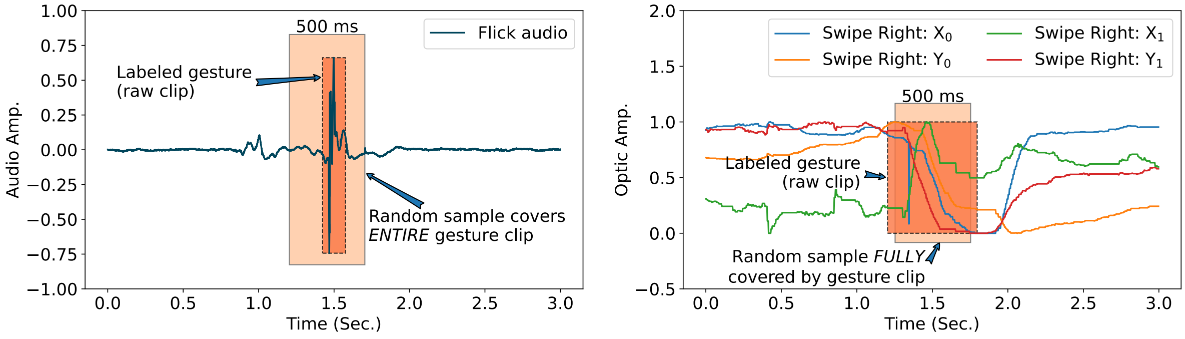
## 4.3 Machine Learning Pipeline

*4.3.1 Data Augmentation.* Since gestures performed by different users are usually different, and gestures could also differ for the same person under different trials as well. We found many interesting insights through extensive human data labelling and analysis (tens of hours of labelling). Flick gesture is most consistent across all samples while pinch gesture can differ a lot. Some people tend to keep the thumb and index finger in contact for a longer period before separating them. Gestures are performed at various speed, resulting different lengths (see statistics in Figure 8). The gesture strength also varies resulting higher or lower signal energy. These factors make the model trained with limited data performs poorly on unseen data since the model can be severely overfitted.

We analyze the sensor signals from a physics point of view (e.g., moving speed, contact impact energy, friction noises, environment background noises, etc.), and propose a set of data augmentation techniques to enhance both acoustic and optic signals to avoid overfitting. Specifically, we design four types of augmentations:

- *Speed Augmentation.* We randomly choose a scaling factor $S_{speed}$ from a normal distribution $S_{speed} \sim \mathcal{N}(1.0, \sigma^2)$. We may generate $S_{speed}$ multiple times to make sure $S_{speed}$ is within a given range $[S_{min}, S_{max}]$. Given the speed scaling factor, we draw $N' = S_{speed} \cdot l_w \cdot F_s$ audio data points during the training sample generation instead. Then, we utilize Akima interpolation [1] that fits audio signal well to resample the $N'$ audio data points to $N$ and its corresponding optic signals. [2]
- *Random Time Shift Augmentation.* This step is done as we randomly cropping training samples, described earlier training data preparation.

[2] We also tried a Python audio processing library Librosa [19] for audio speed augmentation in frequency domain, which is much slower than interpolation while the results are similar.



(a) (b)

**Figure 7: Training sample generation using flick audio signal and swipe right gesture optic signals as examples.**

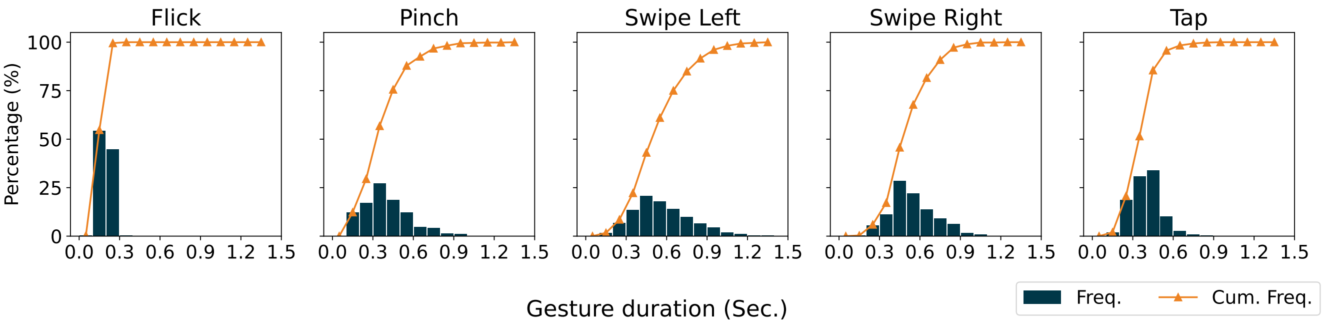


**Figure 8: Duration distribution of *AO-Finger*'s gestures.**

- *Amplitude Augmentation.* We scale the amplitude of the signals by multiplying the signals by a scaling factor $S_{amplitude}$ which is randomly drawn from a normal distribution $S_{amplitude} \sim \mathcal{N}(\mu, \sigma^2)$.
- *Jitter Noise Augmentation.* To simulate the sensor noises, we add random white noises to the signals.

We apply the above augmentations on both audio and optic signals in a synchronized manner. There augmentation steps are applied in real-time during model training, followed by signal pre-processing described in Section 3.1.

*4.3.2 Model Training and Inference.* We train our models on Google Cloud Platform with Tesla V100 GPUs. To scale up parallel model training for parameter fine-tuning, we configure a virtual machine with 32 CPU cores, 64GB RAM and 4 Tesla V100 GPUs. We train our models with Adam optimizer with a starting learning rate of 1e-4 with a decay of 0.9 every 200 epochs for up to 2000 epochs. During inference, we stream realtime sensor data from wristband to a Macbook pro via USB with Intel 6-Core i7 CPU and 16GB RAM for model inference and results visualization. The detection happens every 50ms with a moving window of 500ms with a stride length of 50ms. Then we run the aggregation to output the final prediction result. Moving the whole inference pipeline to the wristband is our future work.

# 5 EVALUATION

## 5.1 Dataset

We collected 10 hours of data from 20 subjects from various age (22-32), gender (17 males and 3 females) and background (africa american, asian, hispanic, and white), each subject contributed half an hour on data collection. After labeling and data cleaning, we around 3000 positive samples (about 600 samples for each gesture). We also generate hard negative samples with sliding window of a stride length of 1000 ms, which generates 3500 hard negative samples. We reserve 5 individual sessions from participants whose data are not used in the model training as *user-independent* test data. We apply sliding window cropping on the raw test data and generate about 500 test samples for each gesture, resulting a test set of 3000 samples including negative class.

## 5.2 Fast Gesture Detection

The fast gesture detection model classifies the gestures into four categories: flick, pinch, tap and none. We set the soft negative sample cropping probability $P_{negative} = 0.3$ to introduce additional negative samples on-the-fly during training.

*5.2.1 Sample Level Detection.* Figure 9 shows the multi-modal network classification results as normalized by each row on test samples. Apart from none gesture class, flick gesture has the highest

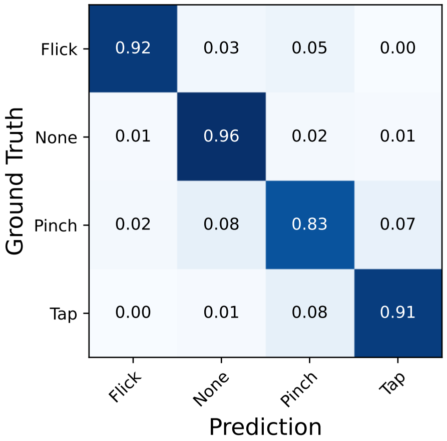


**Figure 9: Confusion matrix of fast gesture detection on testing data set, each class has 500 test samples.**

accuracy. This is mainly due to unique high frequency/energy acoustic features created by high speed index finger movement. A small portion (~5%) of flicks are classified as pinch, which we think are caused by "slow" flicks when the user puts more energy when contacting thumb and index finger rather than separating them. There is no flick gesture classified as tap because flick involves nearly no optic signal changes. Tap gesture is the second most reliable due to the strong optic signals, which distinguishes tap from pinch. We see a small portion (~7%) of confusion between pinch and tap gestures. This is mainly because pinch gestures sometimes also create optic signal changes if the users move the thumb too much. Such errors could be minimized by instructing the users on performing more standard gestures. Pinch detection has the 83% precision, relatively lower than flick and tap. The main reason is that many users pinch gently, not generating sufficient acoustic signals nor optic features, thus 8% of them are missed (i.e., classified as none gesture). Overall, we have very few false positives with a high precision of 96% on none gestures. Note that such results are sample level detection result, we further aggregate the results in adjacent sliding windows for more stable and robust recognition.

*5.2.2 Ablation Study.* We study the impact of multiple factors on the model performance such as signal window length, percentage of hard negatives, data augmentations and neural network architecture.

*Signal window length.* We show the influence of training sample window length (i.e., $l_w$) on fast gesture recognition performance in Figure 10(a). The F-score is reported as 0.7708, 0.9005, 0.707, and 0.67 when the $l_w$ is set to 0.3, 0.5, 0.8, and 1.0, respectively. We observe that $l_w$ has very significant influence on the performance and the best F-score in achieved when the window length is 0.5 seconds. On the one hand, when the $l_w$ is shorter than 0.5 seconds, the captured signals could be too similar to be distinguished as gestures have shared finger motions. For example, the tap and pinch share the same motion that is the thumb and index finger are moving toward each other. However, the two gestures are different before and after the shared finger motions, which needs a window longer than 0.3 seconds. On the other hand, given a longer signal window can significantly reduce the SNR because majority segment are noises (soft negatives are included).

*Percentage of soft negatives.* We also evaluate the impact of the $P_{negative}$ defined in Section 4.2, which is a hyperparameter that controls the probability of cropping soft negative samples from raw data when we generate training samples on-the-fly during training. We run experiments with four probabilities: 10%, 20%, 30% and 40%. Figure 10(b) shows the result. The model achieves best performance with $P_{negative}$ = 30%. When $P_{negative}$ gets smaller, the model does not get sufficient negative samples in training, hence the resulting model is more prone to false positives. As $P_{negative}$ gets larger, less positive samples are generated while training. The performance starts to drop as no sufficient positive samples are provided for training. Balancing the training dataset with parameter $P_{negative}$ tuning is critical for best performance.

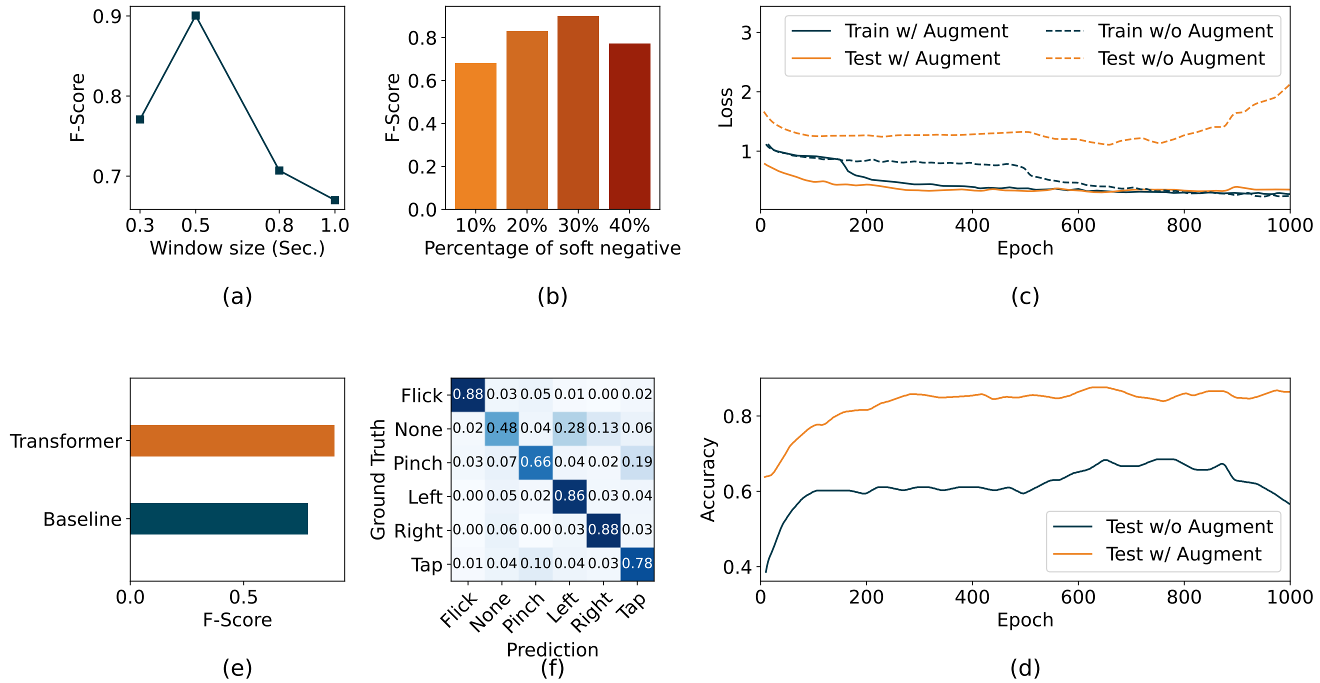
*Data augmentations.* To validate our proposed data augmentations improves the gesture recognition performance and helps stabilize the training processing, we report the train/test loss with and without augmentation in Figure 10(c). The figure shows that the model trained without data augmentation overfits to the training data after 600 epochs, while the augmented training have lower and more stable test loss. The test accuracy reported in Figure 10(d) also confirms that the model trained with data augmentation (accuracy=87.6%) outperforms the non-augmented model (accuracy=68.5%)

*Neural network architecture. AO-Finger* utilizes self-attention transformer that has the strong capability to capture the temporal pattern and relation residing in signals. We are interested in the effectiveness of this architecture on our new sensing modalities and how much performance improvement the transformer architecture could bring to the gesture recognition. Therefore, we use a basic two stream CNN model with out self-attention transformer encoders as the baseline model. As shown in Figure 10(e), the baseline CNN model reports F-score as 0.7827 and *AO-Finger* can achieve higher F-score as 0.9005, which is a 15% improvement. This is because the self-attention can guide the model to focus on gesture signals rather than noise in the sample window.

*Single model for all gestures.* Best F1 score we achieved with the same model on detecting 5 gestures is 0.7535, which is way below the fast gesture detection model. The main reason is that the model overfits to optic sensors in recognizing tap and swipe gestures since the optic features are strong. While it tends to ignore the acoustic features in swiping, which introduce false positives (i.e., none gestures are classified as swipe gestures. This happens a lot when users swipe thumbs left and right without touching index finger.). It also justifies our design with a separate finger contact detection model.

## 5.3 Fine-grained Swipe Gesture Detection

Finger contact detection is the key component whose performance can largely influence the triggering of thumb tracking model. During the test, *AO-Finger*'s contact detection model reports a high precision as 0.9197 and a recall as 0.7975 with a relatively low false positive rate (0.0809). We observe some contact samples (20%) are



(a) (b) (c)

(e) (f) (d)

**Figure 10: We evaluate the following factors for ablation study: a). Signal window length; b). Percentage of soft negatives; c&d). Data augmentation; e). Neural network architecture; f). Single model for all gestures.**

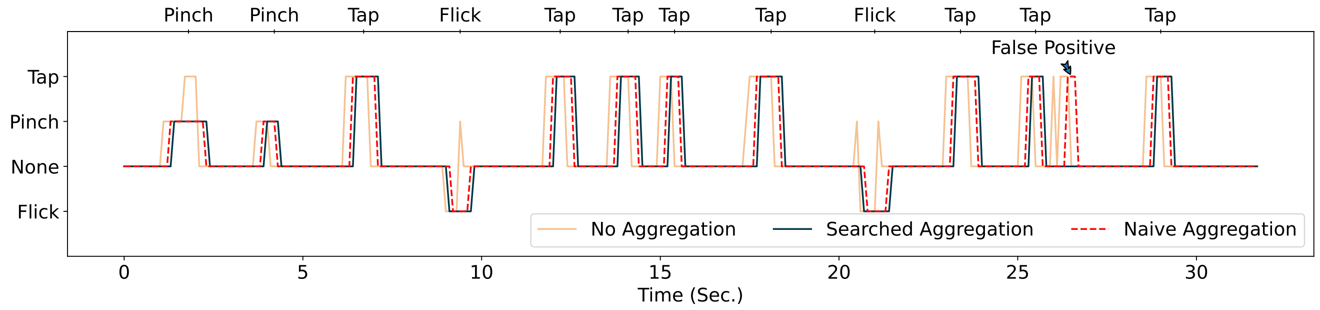


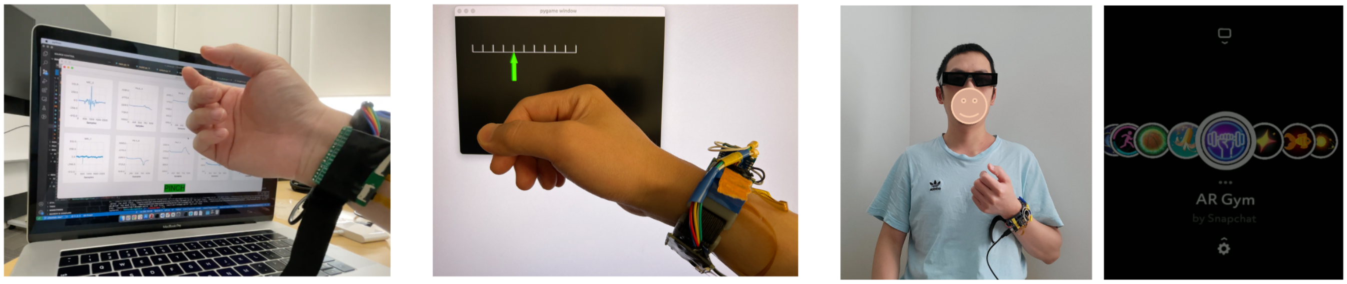
**Figure 11: Aggregation results on a session data.**

misclassified. This is because thumb moving without touch friction also generates sound signals due to the tendons' movement within the wrist. We found the misclassification does not influence the final user experience as the aggregation is applied on the detection results. Moreover, once entering the continuous swiping mode, the tracking will stop only when there is no swiping for 1 second. We further study the usability of fine-grained swipe gestures in later application study section.

## 5.4 Heuristic Aggregation Evaluation

Figure 11 shows the aggregated results on continuous data. We observe that the spikes due to misclassification can be suppressed by aggregation. The naive aggregation cannot accommodate balance between false positive and false negative, resulting in the false positive shown in the figure. Compared with the naive implementation, the aggregation based on our proposed searching method is more stable and can deal with misclassification that is more complicated.

*AO-Finger* also demonstrates strong robustness to false positives thanks to the readiness detection module in Figure 5 and model training with both soft/hard negative samples. False positives are seldom triggered under noisy and challenging scenarios. Please check our Video Figure to get a better sense of the performance.



a). Realtime demo on laptop b). Fine-grained gesture tracking c). Interaction with AR glasses and UI

**Figure 12: *AO-Finger* applications: a). Realtime sensor data visualization and gesture recognition on laptop; b). Fine-grained gesture tracking; c). Interaction with AR glasses, the user can browser applications by swiping left/right, tap to select and flick to exit.**

## 5.5 Application Study

We experience *AO-Finger* in multiple applications as shown in Figure 12: a) a realtime demo on laptop for gesture detection and sensor data visualization; b) precise cursor control via fine-grained gesture tracking; and c) interaction with AR glasses.

*Realtime demo on laptop.* We invite 4 users to try our demo and two of them have no prior knowledge of our system. Users are asked to perform fast gestures randomly and we count the correctness. We collected 200 samples in total for each gesture, and find there are 12 flick gestures misclassified as pinch; 6 pinch gestures are classified as flick, 8 classified as tab; and tab gesture has a high precision that only 5 are misclassified as pinch. The average accuracy reaches 94.83% in this test. Such result is slightly higher than than our offline sample level accuracy thanks to the aggregation and the model has demonstrated good generalizability to new users.

*Fine-grained gesture tracking.* In this experiment, we want to show the accuracy and usability of fine-grained swipe gesture tracking. The users are asked to control a virtual arrow position against a ruler, as shown in Figure 12 b). Results show that arrow is following swipe gestures smoothly and responsively due to high frame rate. Users can easily move the arrow to specified position with minimum trials. Please check our Video Figure for a live demo.

*Interaction with AR glasses.* We compare *AO-Finger* with different interaction modalities listed in Table 2 on a commercial AR glasses. The users are asked to interact with the AR glasses using the built-in touchpad, speech recognition and hand tracking, as well as our proposed *AO-Finger*. According to the major feedback, touchpad is reliable to use due to the physical touch and haptic feedback. It also enables fine-grained control on selecting apps by swiping gestures. However, users have to raise the arms while using the touchpad which incurs arm fatigue after a short period of continuous use. Speech recognition is not preferred due to the limited functionalities and it is not suitable to use in public environments. Hand tracking on the tested AR glasses has noticeable lag and adds significant computation overhead to the device which leads to overheating issues more frequently. *AO-Finger* enables basic interaction with fine-grained control. It is highlighted that the inconspicuous, effortless micro gestures are highly appreciated compared to other modalities. The main negative feedback is the low sensitivity issue to trigger swipe gestures (e.g., some swipe gestures are triggered with a short delay since contact needs to be detected for triggering swipe mode), which we will address by fine-tuning the system configuration parameters as future work.

## 6 RELATED WORK

We introduce the related work from two topics:

**Finger Gesture Recognition.** Finger gesture recognition has a long history. Recently, as XR becomes popular, infrastructure-free solutions, due to its portable user experience, attracts ever-increasing attention from the research community and industries. Naturally, XR devices carries camera for finger gesture recognition. However, the hand interaction is limited to the FoV of the camera and results in user's fatigue. To solve the FoV problem, solutions staying with user's hand is required. Gesture tracking using ring-shape devices [9, 17, 42] on fingers are proposed. However, ring-shaped devices cannot achieve the small dimension of a real ring (the space on the finger is very limited), thereby having a hard time creating a seamless and effortless experience in daily use. Hands-free solutions on the wrist nowadays attract more attention due to their better user experience. Beamband [15] utilizes an ultrasonic array on the wrist to detect the partial hand shape and infer the hand gesture. Due to the low resolution of the ultrasonic beam, fine-grained finger gesture tracking is not supported. Several studies [13, 36, 38, 39] proposed a updated version, i.e., to reconstruct 3D hand pose from camera frames of a hand part such as palm and hand back. The proposed systems utilize data-driven models to infer the finger gestures. While optic sensing can capture high fidelity motion, these solutions require a large area of hand to be visible to make inference feasible and consumes much more computational resources. Instead of challenging finger gestures inference using part of the hand, more studies choose to utilize the wrist activities as large part of tendons that controls gestures go through wrist and causes deformation on wrist. Pressure sensors [7], Barometric sensors [28], ultrasonic sensor [14], IR sensors [20], EMG sensors [26], and IMU [34, 40] are proposed to capture the wrist activities. There are also several studies leverage modality fusion, such as EMG and IMU [6].

Different from the existing works, *AO-Finger* utilizes optic sensors to capture the fine-grained thumb information and a modified stethoscope microphone for the global information of the gestures

Table 2: Interaction modalities for AR glasses.

| Modality | Advantages | Limitations |
|---|---|---|
| Touchpad | Physical touch, reliable | Arm fatigue, involve macro body movements |
| Speech recognition | Hands-free | Not suitable in public |
| Hand tracking | Hands-free | Latency; line-of-sight and lighting condition requirement |
| *AO-Finger* | Inconspicuous, fine-grained, hands-free | Require extra hardware |

on wrist. The fusion of the two sensing modalities provides an *integral fine-grained finger gesture recognition experience.*

**AR/VR Interaction.** The ever-increasing demand of XR application brings new requirements (e.g., convenient to access and easy to use) to the interaction technologies. Beside the traditional interaction such as touchpad [10], hand-held controller [22], camera [31], and voice [11], many input methods with better user experience have been proposed and discussed. Zhang et al. [41] proposed Tapskin, a IMU-based system that can turn skin into a touch surface. Plopski et al. [25] discussed the gaze/eye-tracking interaction in a comprehensive survey Tauscher et al. [30] discussed the feasibility of using EEG on XR headset.

*AO-Finger* shares the same vision with these studies and brings better XR experience by providing effortless inconspicuous and fine-grained finger gesture interaction.

## 7 DISCUSSION

**Limitations.** *AO-Finger* is only a research prototype in current stage, it has several major limitations: *1) Offloading inference.* Current system offloads sensor data to a laptop via USB for model inference, which is only sufficient to validate the gesture detection performance. To make it as a well engineered product, we need to move the computation to the wristband. *2) Conditioned gesture detection.* Current system requires the user's hand to be relatively static for a short period to trigger the detection. While this design significantly reduces the false alarms, it adds additional friction on the user to use the system. *3) Validation on large population.* Due to the pandemic, we only have access to 20+ users for data collection and testing. Although *AO-Finger* has shown promising results on this group of users, we need to validate the system on a larger population to make it as a mature solution.

**Future Work.** We have the following directions as future work: *1) On-device model inference. AO-Finger* uses BLE SoC for both wireless communication and data processing, which has very limited computation resources. In order to achieve on-device model inference, we will introduce a separate MCU specifically for data processing and model inference. *2) Sensing hardware optimization.* We also plan to explore and enhance our sensing hardware. We will explore optic sensor with larger FoV so that finger movements can be more reliably captured. A array of modified microphones could enable more features in hand gesture tracking. *3) Large scale data collection.* Although our physics based data augmentation has shown tremendous help given we have limited data. Large scale data collection (e.g., hundreds or more) is still needed for a mature solution. We will seek ways to do that at such scale. *4) System usability study.* Our system provides a fine-grained and effortless finger gesture tracking experience to users. To further quantize the usability of AO-Finger, we plan to do subjective usability studies, such as System Usability Scale (SUS) [2] and NASA Task Load Index (TLX) [23].

## 8 CONCLUSION

In this paper, we propose *AO-Finger*, a hands-free and fine-grained gesture sensing system aiming for next-generation XR input experience. *AO-Finger* fuses on-wrist direct optic sensing and indirect acoustic sensing. We design a set of inconspicuous and effortless micro gestures for *AO-Finger* and implement *AO-Finger* with a modified stethoscope microphone and two high-speed optic motion sensors. The fine-grained gesture sensing is achieved via a two-branch architecture: a CNN-Transformer based fast gesture (i.e., flick, pinch, and tap) detector and a continuous fine-grained finger tracking model based on finger contact detection. To further enhance system robustness, we propose to utilize data augmentation for stabilized model training and aggregation to suppress false alarms. The extensive evaluation shows *AO-Finger* can perform accurate and robust finger gesture sensing as well as provide excellent user experience.

## REFERENCES

[1] Hiroshi Akima. 1970. A new method of interpolation and smooth curve fitting based on local procedures. *Journal of the ACM (JACM)* 17, 4 (1970), 589–602.

[2] John Brooke et al. 1996. SUS-A quick and dirty usability scale. *Usability evaluation in industry* 189, 194 (1996), 4–7.

[3] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. 2020. End-to-End Object Detection with Transformers. In *Computer Vision – ECCV 2020*, Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm (Eds.). Springer International Publishing, Cham, 213–229.

[4] Edwin Chan, Teddy Seyed, Wolfgang Stuerzlinger, Xing-Dong Yang, and Frank Maurer. 2016. User Elicitation on Single-Hand Microgestures. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (San Jose, California, USA) *(CHI '16)*. Association for Computing Machinery, New York, NY, USA, 3403–3414. https://doi.org/10.1145/2858036.2858589

[5] Liwei Chan, Rong-Hao Liang, Ming-Chang Tsai, Kai-Yin Cheng, Chao-Huai Su, Mike Y. Chen, Wen-Huang Cheng, and Bing-Yu Chen. 2013. FingerPad: Private and Subtle Interaction Using Fingertips. In *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology* (St. Andrews, Scotland, United Kingdom) *(UIST '13)*. Association for Computing Machinery, New York, NY, USA, 255–260. https://doi.org/10.1145/2501988.2502016

[6] J. Guillermo Colli-Alfaro, Anas Ibrahim, and Ana Luisa Trejos. 2019. Design of User-Independent Hand Gesture Recognition Using Multilayer Perceptron Networks and Sensor Fusion Techniques. In *2019 IEEE 16th International Conference on Rehabilitation Robotics (ICORR)*. IEEE, New York, NY, USA, 1103–1108. https://doi.org/10.1109/ICORR.2019.8779533

[7] Artem Dementyev and Joseph A. Paradiso. 2014. WristFlex: Low-Power Gesture Input with Wrist-Worn Pressure Sensors. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology* (Honolulu, Hawaii, USA) *(UIST '14)*. Association for Computing Machinery, New York, NY, USA, 161–166. https://doi.org/10.1145/2642918.2647396

[8] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929* (2020).

[9] Sarthak Ghosh, Hyeong Cheol Kim, Yang Cao, Arne Wessels, Simon T. Perrault, and Shengdong Zhao. 2016. Ringteraction: Coordinated Thumb-Index Interaction Using a Ring. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems* (San Jose, California, USA) *(CHI EA '16)*. Association for Computing Machinery, New York, NY, USA, 2640–2647. https://doi.org/10.1145/2851581.2892371

[10] Google. 2022. Tech Specs – Glass. https://www.google.com/glass/tech-specs/

[11] Mahfoud Hamidia, Nadia Zenati, Hayet Belghit, Kamila Guetiteni, and Nouara Achour. 2015. Voice interaction using Gaussian mixture models for augmented reality applications. In *2015 4th International Conference on Electrical Engineering (ICEE)*. IEEE, IEEE, New York, NY, USA, 1–4.

[12] Teng Han, Khalad Hasan, Keisuke Nakamura, Randy Gomez, and Pourang Irani. 2017. Soundcraft: Enabling spatial interactions on smartwatches using hand generated acoustics. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*. Association for Computing Machinery, New York, NY, USA, 579–591.

[13] Fang Hu, Peng He, Songlin Xu, Yin Li, and Cheng Zhang. 2020. FingerTrak: Continuous 3D Hand Pose Tracking by Deep Learning Hand Silhouettes Captured by Miniature Thermal Cameras on Wrist. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4, 2 (June 2020), 71:1–71:24. https://doi.org/10.1145/3397306

[14] Youjia Huang, Xingchen Yang, Yuefeng Li, Dalin Zhou, Keshi He, and Honghai Liu. 2017. Ultrasound-based sensing models for finger motion classification. *IEEE journal of biomedical and health informatics* 22, 5 (2017), 1395–1405.

[15] Yasha Iravantchi, Mayank Goel, and Chris Harrison. 2019. BeamBand: Hand gesture sensing with ultrasonic beamforming. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. Association for Computing Machinery, New York, NY, USA, 1–10.

[16] Jonghwa Kim, Stephan Mastnik, and Elisabeth André. 2008. EMG-Based Hand Gesture Recognition for Realtime Biosignal Interfacing. In *Proceedings of the 13th International Conference on Intelligent User Interfaces* (Gran Canaria, Spain) *(IUI '08)*. Association for Computing Machinery, New York, NY, USA, 30–39. https://doi.org/10.1145/1378773.1378778

[17] Hyunchul Lim, Jungmin Chung, Changhoon Oh, SoHyun Park, Joonhwan Lee, and Bongwon Suh. 2018. Touch+Finger: Extending Touch-Based User Interface Capabilities with "Idle" Finger Gestures in the Air. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology* (Berlin, Germany) *(UIST '18)*. Association for Computing Machinery, New York, NY, USA, 335–346. https://doi.org/10.1145/3242587.3242651

[18] Iván Markman. 2021. *The Future of XR in A Post-Pandemic World*. Forbes. Retrieved Sept 06, 2022 from https://www.forbes.com/sites/verizon-media/2021/06/15/the-future-of-xr-in-a-post-pandemic-world/?sh=4628758923f4

[19] Brian McFee, Colin Raffel, Dawen Liang, Daniel P Ellis, Matt McVicar, Eric Battenberg, and Oriol Nieto. 2015. librosa: Audio and music signal analysis in python. In *Proceedings of the 14th python in science conference*, Vol. 8. Scipy, IEEE, Austin, TX, USA, 18–25.

[20] Jess McIntosh, Asier Marzo, and Mike Fraser. 2017. Sensir: Detecting hand gestures with a wearable bracelet using infrared transmission and reflection. In *Proceedings of the 30th annual ACM symposium on user interface software and technology*. Association for Computing Machinery, New York, NY, USA, 593–597.

[21] Jess McIntosh, Asier Marzo, Mike Fraser, and Carol Phillips. 2017. Echoflex: Hand gesture recognition using ultrasound imaging. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. Association for Computing Machinery, New York, NY, USA, 1923–1934.

[22] Meta. 2022. Meta Quest 2 Accessories | Meta Quest. https://store.facebook.com/quest/products/quest-2

[23] NASA. 2022. NASA TLX Task Load Index. https://humansystems.arc.nasa.gov/groups/tlx/

[24] OMNIVISION. 2022. OMNIVISION introduces 1/4-inch, 5-megapixel raw sensor for high-performance mobile applications. https://www.ovt.com/press-releases/omnivision-introduces-1-4-inch-5-megapixel-raw-sensor-for-high-performance-mobile-applications/

[25] Alexander Plopski, Teresa Hirzle, Nahal Norouzi, Long Qian, Gerd Bruder, and Tobias Langlotz. 2022. The Eye in Extended Reality: A Survey on Gaze Interaction and Eye Tracking in Head-Worn Extended Reality. *ACM Comput. Surv.* 55, 3, Article 53 (mar 2022), 39 pages. https://doi.org/10.1145/3491207

[26] T. Scott Saponas, Desney S. Tan, Dan Morris, Ravin Balakrishnan, Jim Turner, and James A. Landay. 2009. Enabling Always-Available Input with Muscle-Computer Interfaces. In *Proceedings of the 22nd Annual ACM Symposium on User Interface Software and Technology* (Victoria, BC, Canada) *(UIST '09)*. Association for Computing Machinery, New York, NY, USA, 167–176. https://doi.org/10.1145/1622176.1622208

[27] William J Shelstad, Dustin C Smith, and Barbara S Chaparro. 2017. Gaming on the rift: How virtual reality affects game user satisfaction. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, Vol. 61. SAGE Publications Sage CA: Los Angeles, CA, 2072–2076.

[28] Peter B Shull, Shuo Jiang, Yuhui Zhu, and Xiangyang Zhu. 2019. Hand gesture recognition and finger angle estimation via wrist-worn modified barometric pressure sensing. *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 27, 4 (2019), 724–732.

[29] Yuta Sugiura, Fumihiko Nakamura, Wataru Kawai, Takashi Kikuchi, and Maki Sugimoto. 2017. Behind the palm: Hand gesture recognition through measuring skin deformation on back of hand by using optical sensors. In *2017 56th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE)*. IEEE, IEEE, New York, NY, USA, 1082–1087.

[30] Jan-Philipp Tauscher, Fabian Wolf Schottky, Steve Grogorick, Paul Maximilian Bittner, Maryam Mustafa, and Marcus Magnor. 2019. Immersive EEG: evaluating electroencephalography in virtual reality. In *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. IEEE, IEEE, New York, NY, USA, 1794–1800.

[31] Dorin Ungureanu, Federica Bogo, Silvano Galliani, Pooja Sama, Xin Duan, Casey Meekhof, Jan Stühmer, Thomas J. Cashman, Bugra Tekin, Johannes L. Schönberger, Pawel Olszta, and Marc Pollefeys. 2020. HoloLens 2 Research Mode as a Tool for Computer Vision Research. *CoRR* abs/2008.11239 (2020). arXiv:2008.11239 https://arxiv.org/abs/2008.11239

[32] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017).

[33] Peng Wang, Xiaoliang Bai, Mark Billinghurst, Shusheng Zhang, Xiangyu Zhang, Shuxia Wang, Weiping He, Yuxiang Yan, and Hongyu Ji. 2021. AR/MR remote collaboration on physical tasks: A review. *Robotics and Computer-Integrated Manufacturing* 72 (2021), 102071.

[34] Hongyi Wen, Julian Ramos Rojas, and Anind K. Dey. 2016. Serendipity: Finger Gesture Recognition Using an Off-the-Shelf Smartwatch. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (San Jose, California, USA) *(CHI '16)*. Association for Computing Machinery, New York, NY, USA, 3847–3851. https://doi.org/10.1145/2858036.2858466

[35] Stefan Wiedenmaier, Olaf Oehme, Ludger Schmidt, and Holger Luczak. 2003. Augmented reality (AR) for assembly processes design and experimental evaluation. *International journal of Human-Computer interaction* 16, 3 (2003), 497–514.

[36] Erwin Wu, Ye Yuan, Hui-Shyong Yeo, Aaron Quigley, Hideki Koike, and Kris M. Kitani. 2020. Back-Hand-Pose: 3D Hand Pose Estimation for a Wrist-worn Camera via Dorsum Deformation Network. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology (UIST '20)*. Association for Computing Machinery, New York, NY, USA, 1147–1160. https://doi.org/10.1145/3379337.3415897

[37] Xuhai Xu, Jun Gong, Carolina Brum, Lilian Liang, Bongsoo Suh, Shivam Kumar Gupta, Yash Agarwal, Laurence Lindsey, Runchang Kang, Behrooz Shahsavari, Tu Nguyen, Heriberto Nieto, Scott E Hudson, Charlie Maalouf, Jax Seyed Mousavi, and Gierad Laput. 2022. Enabling Hand Gesture Customization on Wrist-Worn Devices. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems* (New Orleans, LA, USA) *(CHI '22)*. Association for Computing Machinery, New York, NY, USA, Article 496, 19 pages. https://doi.org/10.1145/3491102.3501904

[38] Dongseok Yang and Younggeun Choi. 2018. Palm glove: wearable glove based on palm-camera for thumb-to-finger tap recognition. In *2018 International Conference on Information and Communication Technology Convergence (ICTC)*. IEEE, New York, NY, USA, 549–551. https://doi.org/10.1109/ICTC.2018.8539616 ISSN: 2162-1233.

[39] Hui-Shyong Yeo, Erwin Wu, Juyoung Lee, Aaron Quigley, and Hideki Koike. 2019. Opisthenar: Hand Poses and Finger Tapping Recognition by Observing Back of Hand Using Embedded Wrist Camera. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology* (New Orleans, LA, USA) *(UIST '19)*. Association for Computing Machinery, New York, NY, USA, 963–971. https://doi.org/10.1145/3332165.3347867

[40] Xiaojing Yu, Zhijun Zhou, Mingxue Xu, Xuanke You, and Xiang-Yang Li. 2020. Thumbup: Identification and authentication by smartwatch using simple hand gestures. In *2020 IEEE International Conference on Pervasive Computing and Communications (PerCom)*. IEEE Computer Society, IEEE, New York, NY, USA, 1–10.

[41] Cheng Zhang, AbdelKareem Bedri, Gabriel Reyes, Bailey Bercik, Omer T. Inan, Thad E. Starner, and Gregory D. Abowd. 2016. TapSkin: Recognizing On-Skin Input for Smartwatches. In *Proceedings of the 2016 ACM International Conference on Interactive Surfaces and Spaces* (Niagara Falls, Ontario, Canada) *(ISS '16)*. Association for Computing Machinery, New York, NY, USA, 13–22. https://doi.org/10.1145/2992154.2992187

[42] Cheng Zhang, Anandghan Waghmare, Pranav Kundra, Yiming Pu, Scott Gilliland, Thomas Ploetz, Thad E Starner, Omer T Inan, and Gregory D Abowd. 2017. FingerSound: Recognizing unistroke thumb gestures using a ring. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 3 (2017), 1–19.

[43] Bing Zhou, Matias Aiskovich, and Sinem Guven. 2021. Acoustic Sensing-based Hand Gesture Detection for Wearable Device Interaction. *arXiv preprint arXiv:2112.05986* (2021).