ClenchClick: Hands-Free Target Selection Method Leveraging Teeth-Clench for Augmented Reality

XIYUAN SHEN, YUKANG YAN*, CHUN YU, and YUANCHUN SHI, Department of Science and Technology, Tsinghua University, China

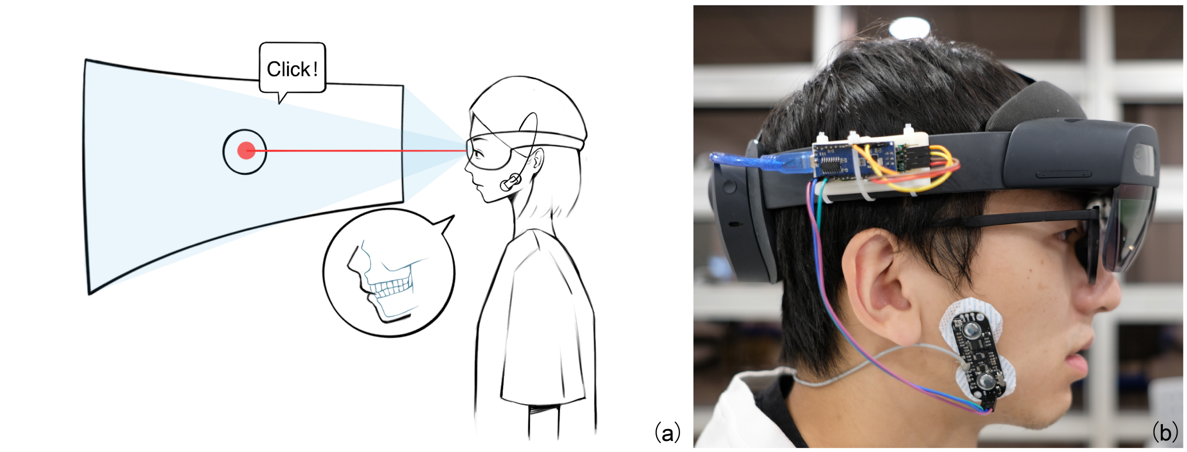

Fig. 1. ClenchClick is a clench-based target selection system for AR devices: (a) The user performs the ClenchClick action by first moving the head pointer to the target and then completing a brief teeth clench. (b) An EMG sensor is worn on the cheek.

We propose to explore teeth-clenching-based target selection in Augmented Reality (AR), as the subtlety in the interaction can be beneficial to applications occupying the user's hand or that are sensitive to social norms. To support the investigation, we implemented an EMG-based teeth-clenching detection system (ClenchClick), where we adopted customized thresholds for different users. We first explored and compared the potential interaction design leveraging head movements and teeth clenching in combination. We finalized the interaction to take the form of a Point-and-Click manner with clenches as the confirmation mechanism. We evaluated the taskload and performance of ClenchClick by comparing it with two baseline methods in target selection tasks. Results showed that ClenchClick outperformed hand gestures in workload, physical load, accuracy and speed, and outperformed dwell in work load and temporal load. Lastly, through user studies, we demonstrated the advantage of ClenchClick in real-world tasks, including efficient and accurate hands-free target selection, natural and unobtrusive interaction in public, and robust head gesture input.

CCS Concepts: • **Human-centered computing → Gestural input**; **Mixed / augmented reality**.

*denotes as the corresponding author.

Authors' address: Xiyuan Shen, shenxiyuan_864@163.com; Yukang Yan, yanyukanglwy@gmail.com; Chun Yu, chunyu@mail.tsinghua.edu.cn; Yuanchun Shi, shiyc@tsinghua.edu.cn, Department of Science and Technology, Tsinghua University, Beijing, China.

2474-9567/2022/9-ART139 $15.00
https://doi.org/10.1145/3550327

Additional Key Words and Phrases: target selection, hands-free interaction, augmented reality, EMG sensing

ACM Reference Format:
Xiyuan Shen, Yukang Yan, Chun Yu, and Yuanchun Shi. 2022. ClenchClick: Hands-Free Target Selection Method Leveraging Teeth-Clench for Augmented Reality. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 6, 3, Article 139 (September 2022), 26 pages. https://doi.org/10.1145/3550327

1 INTRODUCTION

Target selection is one of the most basic tasks in Augmented Reality (AR). As AR headsets continue to gain popularity as both productivity tools in the industry[5] and as personal assistants in daily life[55], a variety of their usage scenarios show the need for hands-free interaction and subtle interaction. For example, professional workers on the assembly [58] often have their hands occupied for long periods at work. In public spaces, users favor interactions that are achievable with subtle movements to avoid disturbing bystanders.

We argue that teeth clenches have the potential to serve as a hands-free and subtle confirmation method of target selection in AR to fulfill the current needs. Compared to existing methods, including dwell-based [27, 65] and head gesture-based [62] selection confirmation, a clench provides explicit proprioceptive feedback of teeth collision, which strengthens the sense of control, and does not require large-scale head movements. There exists prior work leveraging different clench force levels [61] and tongue-teeth contacts [43] as input methods, but we found that methods to combine the usage of head movement based cursor control and teeth-clench based confirmation in target selections in AR scenarios to be under explored. Thus, we present ClenchClick, with which users control the pointer with head movement and confirm the selection by a clench when the pointer is within the target. The clench action is detected based on the EMG signals collected by the add-on sensors, which we designed a container for and attached to the headset in a compact manner. In the implementation aspect, we implemented a real-time detection system and designed a calibration phase which provided a personalized threshold for different users to improve the performance.

Supported by the detection system, we thoroughly investigated the interaction design, user experience in target selection tasks, and user performance in real-world tasks in a series of user studies. In our first user study, we explored nine potential designs and compared the three most promising designs (ClenchClick, ClenchCrossingTarget, ClenchCrossingEdge) with a hand-based (Hand Gesture) and a hands-free (Dwell) baseline in target selection tasks. ClenchClick had the best overall user experience with the lowest workload. It outperformed Hand Gesture in both physical and temporal load, and outperformed Dwell in temporal and mental load. In the second study, we evaluated the performance of ClenchClick with two detection methods (General and Personalized), in comparison with a hand-based (Hand Gesture) and a hands-free (Dwell) baseline. Results showed that ClenchClick outperformed Hand Gesture in accuracy (98.9% v.s. 89.4%), and was comparable with Dwell in accuracy and efficiency. We further investigated users' behavioral characteristics by analyzing their cursor trajectories in the tasks, which showed that ClenchClick was a smoother target selection method. It was more psychologically friendly and occupied less of the user's attention. Finally, we conducted user studies in three real-world tasks which supported hands-free, social-friendly, and head gesture interaction. Results revealed that ClenchClick is an efficient and accurate target selection method when both hands are occupied. It is social-friendly and satisfying when performing in public, and can serve as activation to head gestures which significantly alleviates false positive issues.

In summary, we make the following contributions.

- We explored potential designs of combining head movements and teeth clenches in target selection, mainly considering user experience, and compared ClenchClick with two baseline methods. ClenchClick outperforms baselines and is more physically effortless, mentally relaxing, and subjectively faster.

- We investigated the usage of ClenchClick in real-world tasks (hands-free assembling, taking photos in public, clench-activated head gestures) and showed that advantages of interacting with ClenchClick included more efficient and accurate hands-free target selection, more natural and unobtrusive interaction in public, and more controlled head gesture input.
- A minor contribution is that we implemented an EMG-based clench detection algorithm that provides a general model applicable to average users, and a calibration phase to further improve the detection performance with personalized parameters.

2 RELATED WORK

2.1 In-mouth Interactions

In-mouth motions mainly involve movements of the user's tongue, teeth, bite, and throat [11], which are subtle and nearly unnoticeable to other people. Therefore extensive efforts have been undertaken in developing novel sensing methods to recognize in-mouth motions and to leverage them in human computer interaction. Qiao Zhang[67], Phuc Nguyen[43] and Takuro Nakao[42] proposed the use of EMG to detect teeth clenching and tongue movement. OsteoConduct[68], EarSense [46] and Touch-Touch[3] used vibration sensors or bone conduction microphones to detect the sound and vibration of tooth click. Bitey[3] further classifies clicks of different teeth pairs. Similarly, Byte.It[56] used gyroscopes to distinguish between different teeth collisions. Clench Interaction[61] and On the Tip of my Tongue[12] used pressure sensors to sense tooth clenching and tongue movement, respectively. TEMPO[51] and Tongue-in-Cheek[19] used non-invasive optoelectronic sensing and microwave sensing to identify tongue movements. Siyoung Lee[35] proposed an ultra-thin electronic skin attached to the larynx to sense the user's vocalizations, but the electronic skin remained expensive, fragile, and difficult to manufacture. Unlike previous work on recognizing in-mouth motions, we focused on improving user experience in target selection in VR, through the combined use of head movements and teeth-clenching in cursor control and selection confirmation, respectively. We acknowledge that this interaction manner has been raised in previous projects[61]. However, through a holistic process of exploring the design space, benchmarking the target selection performance, and evaluating in real-world applications, we expect to comprehensively provide more empirical insights that reflect how users interact and feel when interacting with this method in different cases.

2.2 Hands-free Target Selection Methods for HMD

Target selection is one of the most dominant input modes for head-mounted display devices [63, 66], which can be divided into target acquisition and target selection [10]. The hands-free interaction of HMDs serves as an alternative to gestures when users' hands are occupied or have limited mobility [41].

For target acquisition, HMD devices had a pretty uniform approach, mainly using head motion or eye-tracking [18, 20, 24] to control the cursor. Pinpointing[33] combined the above two modalities to achieve more accurate selection. For target selection, dwell was a commonly used method [6]. Nevertheless, it suffered from severe false-positive problems, especially when multiple targets exist. A dwell time of 450 to 1000 ms could counterbalance the false-positive problem and waiting time [65]. Orbits[16] defined a circular dwell trajectory to complete target selection. It solved the false-positive problem but inevitably reduced click efficiency.

Apart from dwelling, voice input[14, 15], head gestures[44, 62, 64], foot movements[14, 25, 40], and facial expressions[9, 28, 32, 49, 57, 57] were also used for target selection for HMDs. The vast majority of these input methods were ultimately applied in VR. HeadTurn[44] and HeadCross[62] used specific head movement patterns. Foot interactions such as GazeTap[14, 25, 40] used foot movements as confirmation actions to address the input need for surgeons in operating rooms. Voice commands [14, 15] have been applied by HoloLens[39] , Oculus[45], and other products, but one issue was that speech was easily affected by the environment and noise [47].

Target selection methods based on facial expressions leveraged frown [49, 57], blink [9, 28, 57], smile, mouth [32] and teeth clenching [49] as input. Jae Kwang Cha et al. [9] recognized facial skin deformation during blink movements, achieving 95.4% accuracy. Umur Aybars Ciftci[32] and others used cameras to identify the opening and closing of the user's mouth. Javier San Agustin[49] and Ker-Jiun Wang[57] designed VR input systems for mentally and physically challenged users. The sensing systems identified changing EEG signals due to facial expression changes, but they were bulky and not portable. Clench Interaction[61] leveraged an in-mouth pressure sensor to detect clenching to facilitate VR interactions. Although these studies proposed the concept of facial input in HMDs, few have either optimized the recall rate or considered the problem of false-positives. Our study improved both metrics through algorithmic design to facilitate daily interaction.

Unlike VR devices that emphasize personal immersion, users tend to wear AR glasses for a long time to accomplish various tasks in non-private life scenarios. Thus, we would like to propose a user-friendly [26, 31, 34] input action with high social acceptance in AR. ClenchClick has a private and subtle input action inside the mouth. Users endure a lower physical and psychological load during prolonged use.

2.3 EMG Signal Detection

Since the discovery by A. Galvani that human muscles produce electrical signals when they contract [29], the precise detection of discrete EMG pulse signals has been an essential issue in the analysis and understanding of the human motor system [48]. The most intuitive computer detection method is to compare the EMG signal with a fixed threshold [52]. The accuracy of signal recognition rose rapidly after considering the influence of environmental noise [37] and the standard deviation of the EMG intensity distribution [38] on the selection of the threshold. P. Bonato proposed the double-threshold method [8, 60], and achieved a detection accuracy of 95%.

Several works have used statistical distributions to optimize the selection of thresholds. J.A. Gurrero fitted a normal distribution of signal intensities [21] and simulated the Bernoulli distribution consisting of the number of detected signals under different threshold conditions [22]. He then used the Bayesian approach [21] to estimate more accurate thresholds. R. Merletti used adjacent electrodes to form an array [36] with multi-channel EMG information to improve the detection confidence. Due to the significant individual differences in EMG signals [2, 23], the rational use of statistical features of individual EMG signals helped improve the accuracy of threshold detection. However, all the algorithms mentioned above proposed offline recognition models based on big data statistics or iterations. Our work applied EMG sensing to real-time human-machine interaction. Therefore, we used a two-step "calibration-detection" real-time detection algorithm, first optimizing the model parameters using a small amount of user data, then executing the detection.

In addition to threshold detection, the energy fluctuations during the rise and fall of the EMG signal are useful to detect the onset and offset points of the signal. TEKO (Teager-Kaiser Energy Operator)[4, 53] and MUAP (Motor Unit Action Potential)[17] are representative effective edge features used for real-time signal detection. In our ClenchClick algorithm, we specifically target the patterns of EMG signals generated by the bite muscle. We combine threshold detection with TEKO-based edge detection and use a voting mechanism to improve the robustness of clench detection.

3 TEETH-CLENCHING DETECTION

Driven by a large amount of user data, we implemented a pipeline to detect the user's clenches. We introduced an optional calibration phase which predicted the personalized detection threshold based on the statistical features of the user's clench signal segments. Our personalized detection method achieved an offline accuracy of 98.0% and a false-positive count of 4.26 times/hour.

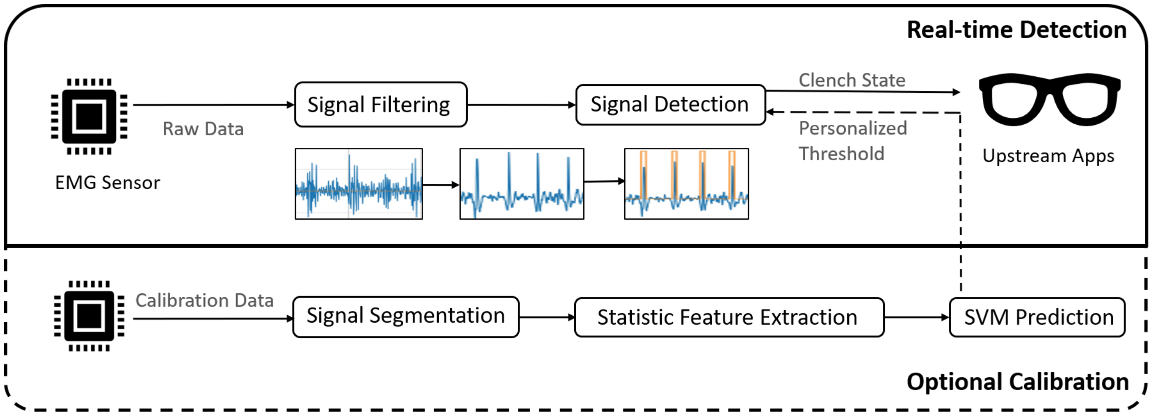

Fig. 2. The system overview. Clenching detection system is composed of an EMG sensor and a pair of AR glasses. We implemented a detection pipeline and an optional calibration phase to identify user input intention to support upstream applications.

3.1 Data Collection

To better understand the characteristics of the EMG signals of different users performing intentional clenches, we collected clench data from 20 participants for algorithm design, model training, and offline testing. Participants aged from 20 to 24 years, of whom 7 were females and 13 were males. All were enrolled students recruited from university campuses. Eight participants had experience using VR/AR.

We used a Myoware AT-04-001 electromyography(EMG) sensor to measure the electrical signals generated by muscle contraction through electrodes attached to the human skin. Referring to previous work[67], we conducted pre-experiments and decided to attach the chip and electrodes to the masseter muscle on the outside of the user's cheek, where clenching signals are robustly sensed without involving much noise caused by other facial expressions. AT-04-001 was connected to an Arduino Nano with Dupont lines. An HC-05 chip was connected to transmit EMG data through Bluetooth communication to a Dell G3 laptop. Users wore a pair of Microsoft HoloLens 2 AR glasses. A HoloLens 2 had IMU sensors inside and provided the real-time position of the user's head. We 3D-printed a plastic case and fixed the prototype on the lateral side of HoloLens2.

We collected clench data from participants when they sat, stood, and walked to simulate various possible body postures when they used clench input to control the AR glasses in their daily lives. Each participant was required to complete six rounds of data collection. In each round, the participant continuously moved the head-motion pointer to appearing objects and performed clenches 20 times. We recorded 60 frames of EMG signals and head positions per second.

3.2 Detection

The pipeline and design procedure for clench detection are presented in this section. Based on the collected data, we designed a filter and rule-based model to detect the onset, peak point, and offset of EMG signals. We optimized an adjustable **threshold** to determine the state of each frame.

3.2.1 Signal Filtering. Signals from the analog filter circuits suffered from high-frequency noise and serious signal zero-point drift, so we added a digital bandpass filter. We applied a Butterworth bandpass [13] filter online with a lower cutoff frequency of 1 Hz and a upper cutoff frequency of 30 Hz. It could resolve the issue of zero drift while retaining most energy and information.

3.2.2 Rule-based Model for Detection. We observed that the waveform of the transient clenching signal contains distinct and strong "rising, reaching the peak and falling" processes. We used the Teager-Kaiser Energy Operator [4, 53] for signal change point detection to detect onsets and offsets. We added peak detection to mask EMG signals of very small amplitudes. We linearly mapped filtered EMG data to [-1023,1023], then set the following rules to determine the state of each frame. Let the signal amplitude of the k_0^{th} frame be $f(k_0)$. We first defined the energy calculated using the **TEKO operator** for EMG signal as $\tau(k)$:

$$\tau(k) = f(k)^2 - f(k-1) * f(k+1) \tag{1}$$

The state of the k_0^{th} frame was defined as $s(k_0)$.

$$s(k_0) \in \{peak\ point, onset, offset, resting\} \tag{2}$$

We used a **detecting threshold** to determine the state of each frame. We set the threshold to be T (acquisition of T is mentioned in the next subsubsection) and defined the detection conditions of the following three states.

Peak point: If the signal is at the local maximum and exceeds the threshold T, the state is "peak point".

$$S(k_0) = Peakpoint \Leftrightarrow \{k_0 \mid f(k_0) > T, f(k_0) > f(k), \forall \mid k - k_0 \mid < 3, k \in N, k_0 \in N\} \tag{3}$$

Onset: The algorithm calculated the **TEKO operator** of the neighboring frames. We utilized a voting mechanism to ensure a relatively tolerant and discriminative scale to preserve atypical clench signals and eliminate weak EMG signals from other muscles. If the signal maintains a high rising rate, the state of k_0 is the "Onset".

$$S(k_0) = Onset \Leftrightarrow Card(\{k \mid \tau(k) > 0.5 * T^2, f(k-1) < f(k), \mid k - k_0 \mid < 3, k \in N, k_0 \in N\}) \geqslant 3 \tag{4}$$

Offset: Similarly, if the signal maintains a high falling rate, the state of k_0 is the "Offset".

$$S(k_0) = Offset \Leftrightarrow Card(\{k \mid \tau(k) > 0.5 * T^2,, f(k-1) > f(k), \mid k - k_0 \mid < 3, k \in N, k_0 \in N\}) \geqslant 3 \tag{5}$$

The detection algorithm determined the EMG state of each frame and maintained a temporal state sequence. When there was a subsequence of "onset,..., onset, wave, offset,..., offset" in the state sequence, the algorithm determined that an input signal was detected. Change point detection effectively reduced high-frequency fluctuating noise cases.

3.2.3 Calculating the Best Threshold T. To obtain the best general detection threshold T, we used labelled intentional clench signals and noisy signals to train our model. We defined the loss function as:

$$\begin{cases} Loss \ = \ -Recall \ + \ False\ positive\ rate \\ Recall \ = \ N_{intentional}/N_{total} \\ False\ positive\ rate = \ N_{noisy}/N_{total} \end{cases} \tag{6}$$

We used dichotomous traversal of thresholds locally to minimize the loss function and ensure that the threshold converged to the optimum. We applied the optimized threshold as the best detection threshold for general clench detection.

3.3 Personalized Calibration

Further, we put forward an optional calibration phase to generate **the personalized detection threshold** for a specific user to achieve more accurate clench detection.

3.3.1 Statistical Feature Extraction. We tested the clench detection algorithm. Results showed significant differences in the magnitude of the EMG signal among individuals, which is a common phenomenon[54] for physiological parameters of the human body. Thus, a general threshold cannot guarantee good detection performance for all users. Therefore, we proposed to involve user calibration to collect a small amount of clench data from each first-time user and predict a more suitable detection threshold.

We divided the data into groups of 20. Each group contained several signal segments from the same user. We treated each group of data as a sample, and calculated the mean value and variance of each data segment's peak value, mean value, maximum rise rate, and maximum fall rate in the group as the model input vector $\overrightarrow{X}$.

$$\overrightarrow{X} = (\overline{Peak}, Peak_{std}, \overline{Mean}, Mean_{std}, \overline{Riserate}, Riserate_{std}, \overline{Fallrate}, Fallrate_{std})$$

3.3.2 Model Training. For each group of data, we used the statistical features $\overrightarrow{X}$ as input. We used the traversal method to obtain the best detection threshold of each group as the model output Y. About 2400 clench signals were collected to train an SVM regression model for personalized threshold prediction.

3.3.3 Number of User Calibrations. In the actual calibration process, a higher number of clenches during calibration provided a more accurate estimation of signal features but imposed additional time and physical load on the user. We referred to the elbow method[30], commonly used in clustering algorithms. The result showed that the cost-effectiveness decreased sharply when the number was around 13. Therefore, we chose N = 13 as the final calibration time.

4 STUDY 1: CLENCHCLICK INTERACTION DESIGN

The primary motivation of ClenchClick is to leverage the clenching action as the confirmation of target selections. However, as there are various confirmation mechanisms (e.g., Point-and-Click [7], Crossing-based confirmation [1]), it is worth exploring in which manner the users feel most comfortable and convenient in performing clenching actions. Therefore, we designed nine selection methods using teeth clenching actions with different confirmation mechanisms. We conducted two iterations in deciding on the ClenchClick interaction design. We narrowed down nine methods to the three most favored candidates through a user interview in the first step. Then, we implemented a proof-of-concept prototype of the candidate methods and conducted a user study to compare their user experiences. We adopted a Fitts'-like target selection task to enable the participants to experience the interaction with different methods and added hand gesture and dwelling-based confirmation as the baselines to compare. The subjective feedback of participants was collected with the NASA TaskLoad scale and interviews.

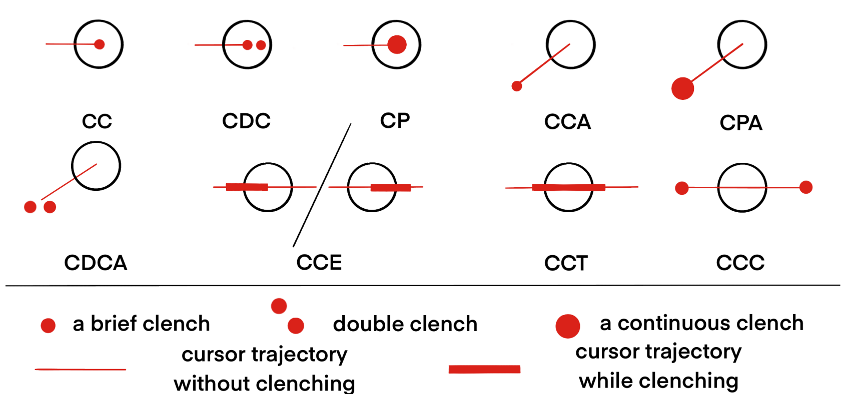

Fig. 3. Schematics of clench-based target selection methods.

Table 1. Text descriptions of clench-based target selection methods.

Input Method	Explanation
Clench-Click (CC)	The user first moves the head pointer to the target area and then completes a brief transient teeth clench in the target area.
Clench-Press (CP)	The user first moves the head pointer to the target area and then completes a continuous teeth clench in the target area for a while.
Clench-Double-Click (CDC)	The user first moves the head pointer to the target area and then completes two transient teeth clenches in the target area.
Clench-Click-Activation (CCA)	The user first completes a brief transient teeth clench to activate all the selectable targets in the interface in the next few seconds. The user then moves the head pointer to the target for selection.
Clench-Press-Activation (CPA)	The user first completes a continuous clench for a while to activate all the selectable targets in the interface in the next few seconds. The user then moves the head pointer to the target for selection.
Clench-Double-Click-Activation (CDCA)	The user first completes two transient teeth clenches to activate all the selectable targets in the interface in the next few seconds. The user then moves the head pointer to the target for selection.
Clench-Crossing-Target (CCT)	The user moves the head pointer to cross the target. From the pointer entering the target area to leaving the target area, the user needs to keep clenching his teeth.
Clench-Crossing-Edge (CCE)	The user moves the head pointer to cross the target. The user needs to keep clenching his teeth when the pointer crosses the edge of the target. Crossing the edge includes entering the target area from the outside and leaving the target area.
Clench-Click-While-Crossing (CCC)	The user moves the head pointer to cross the target and completes a brief transient teeth clench when the pointer is about to enter the target and after the pointer has just left the target.

4.1 Interview of Clenching-based Target Selection Design

4.1.1 Nine Design Candidates. When interacting with headset-based AR interfaces, users use their heads to control the pointer's movement. After the user moves the pointer to the target, they need a mechanism to confirm the selection or to indicate the pointer movement as an intentional action. We propose to leverage the action of clenching as confirmation. However, there are multiple confirmation mechanisms applicable to the clenching action. For example, the user can perform a clench while the pointer is within the target as an indicator of selection, or perform a clench before the pointer enters the target for the same purpose. Therefore, we selected three mostly used mechanisms, which were Point-and-Click (move the pointer into the target and do confirmation), crossing (move the pointer across the target boundary), and mode switch (first activate the target, and then move the pointer into it). We designed Clench-Click(CC), Clench-Press(CP) and Clench-Double-Click(CDC) based on the target-pointing paradigm. Their interaction mechanisms are similar to that of a traditional computer mouse. Three input actions that use clench as the activation for selection are also based on the pointing paradigm, redefining the clenching and head movement sequence. In addition, we devised three target selection methods based on the crossing paradigm. Detailed descriptions of individual designs is listed in Table 1 and demonstrated in Figure 3.

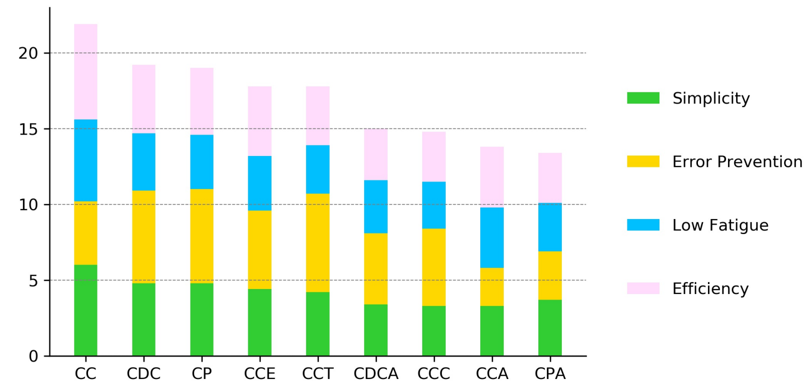

Fig. 4. Subjective ratings of nine clench-based input methods in terms of Simplicity, Error Prevention, Fatigue and Efficiency.

4.1.2 User Interview with Gedanken Experiments. We recruited nine participants from campus and conducted a preliminary user experiment. The participants' age ranged from 19 to 28 years old. Four of them were females, and five were males. All participants had no experience in using AR glasses. During the warm-up phase, participants were free to experience AR glasses (HoloLens 2) to gain a basic understanding of interactions in the AR scene. After that, we interviewed them with a preliminary Gedanken experiment. Participants attempted to perform the task with the nine target selection methods without wearing AR glasses only according to the experimenter's description. We provided a list of paper instructions as guidance. No detection algorithms were applied, and no feedback was provided in this experiment. The purpose was to allow the participant to focus on the user experience instead with the assumption of perfect clench detection. After completing the experience of the nine candidates, each participant rated them according to the following criteria along a 7-point Likert scale (1-strongly disagree, 4-neutral, 7-strongly agree).

We added up the scores of the four dimensions and arranged the clench-based input methods in reverse order (Figure 4). The three target-pointing input methods that completed the click inside the area had the highest score, among which ClenchClick was rated highest in all four dimensions. Seven participants consented that transiently clenching their teeth once was more intuitive and had a shorter selection time. Clench-Crossing-Edge and Clench-Crossing-Target, the two selection methods based on the goal-crossing paradigm, ranked fourth and fifth. However, there were still two participants who preferred selecting with a stroke. They felt that the pointer could move smoother and faster during the crossing. They need not slow down or stop in the process. Methods which use clench as the activation method had the lowest score. Participants felt that this two-step selection method was less efficient than clicking directly in the button area. Furthermore, it was easy to cause wrong selection when there were multiple buttons in the GUI. **In the end, we selected three input actions, including one based on the click paradigm (Clench-Click) and two based on the crossing paradigm (Clench-Crossing-Target, Clench-Crossing-Edge).**

4.2 User Experience Evaluation

We adopted a Fitts'-like task and compared the user experience of the three clench-based input methods with a hand-based(*hand gesture*) and a hands-free(*dwell*) baseline. The goal was to obtain the user experience of different clenching inputs in AR and select the target selection method with the best user experience. In this study, we present the following hypotheses:

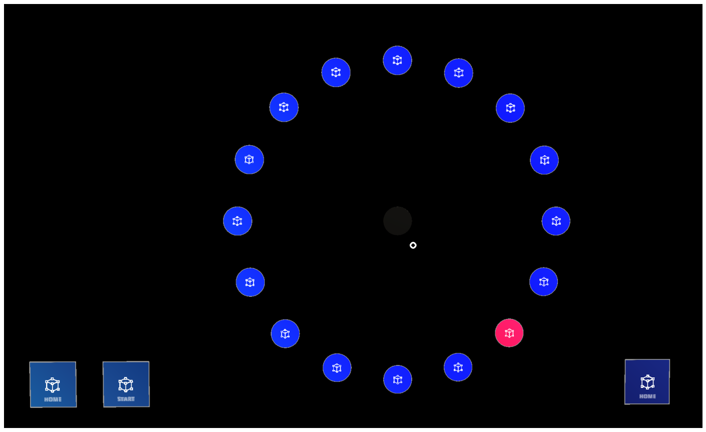

Fig. 5. The experimental interface of user experience evaluation. The interface is a Fitt's-Like scene with 16 buttons arranged in a circular array. The sizes of the buttons and their distances from the circle's center change during the experiment.

H1: Three clench-based techniques may have different workload and user experience in the target selection task due to different paradigms.

H2: As clench action is subtle, the user may perceive clench-based input methods as less physically demanding. Therefore, they may have a lighter workload than hand gesture and dwell.

4.2.1 Participants and Apparatus. We recruited 10 participants. Their ages ranged from 18 to 28 years old (4 females, 6 males). All participants were students recruited from a local campus and had experience in using a computer mouse and using a tablet or smartphone. Two participants had experience using AR or VR devices. Five participants wore glasses during the experiment, but the glasses did not affect the experiment process.

We used the same hardware devices as during the data collection phase in the implementation section.

4.2.2 Interaction Methods. We compared the five target selection methods. Besides three clench-based methods ($ClenchClick$,$ClenchCrossingEdge$, and $ClenchCrossingTarget$), a hand-based ($handgesture$) and a hands-free ($dwell$) baseline were involved.

- **ClenchClick**: The user moves the pointer into the target and then completes a brief teeth clench.
- **ClenchCrossingEdge**: The user moves the head pointer to cross the target. The user needs to keep clenching his teeth when the pointer crosses the edge of the target.
- **ClenchCrosssingTarget**: The user moves the head pointer to cross the target. From the pointer entering the target area to leaving the target area, the user needs to keep clenching.
- **Hand Gesture**: The HoloLens 2 default target selection method, AirTap, which is the gold standard for hand-based target selection in AR/VR, ws used. The participant first moves the pointer into the target, then raises one hand, points up the index finger, quickly pinches down on to the thumb, and returns to the initial hand posture.
- **Dwell**: As the most widely used hands-free target selection method, dwell serves as the baseline for hands-free input. The participant moves the pointer into the target and stares at it for 600 milliseconds. Prior research[65] has verified that this time is neither too short to cause false triggers nor too long to sustain.

4.2.3 Procedure. Each participant used five different input methods to do a Fitts'-Like button selection experiment in the AR interface (See Figure 5). For each input method, we arranged the buttons in a circular array. The size of

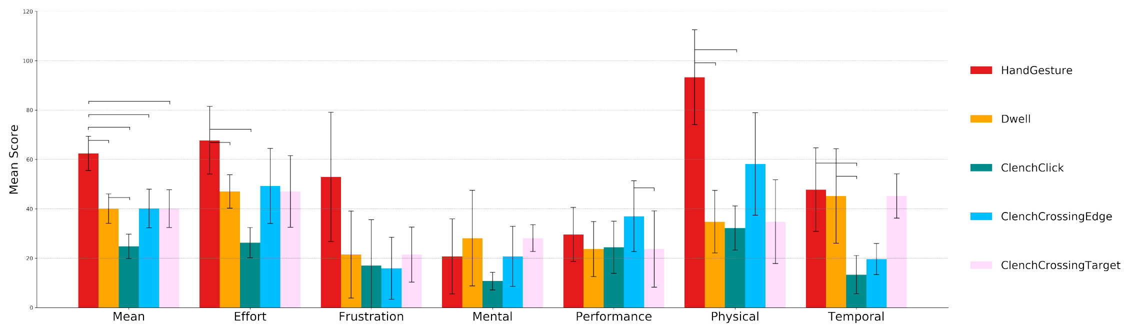

Fig. 6. The mean responses for Task Load. The error bars represent the standard deviations. The statistically significant differences are marked as connecting lines.

buttons and their center distance varied. The participant was required to complete 2 sizes x 2 center distances * 16 positions = 64 trials. Each participant had five seconds to complete each trial. After selection, the participant was asked to move the pointer back to the center of the button array. If the system did not detect an input after the timeout, the system would prompt the participant to continue the experiment and select the next button. The experiment order of different input methods and the order in which the buttons appeared were preset to eliminate the influence of the learning effect on the experiment results. Each participant had a 5-10 minute rest time after completing each input method, and was asked to complete a NASA Taskload questionnaire and briefly talk about their experience of each target selection method.

4.2.4 Result. We analyzed the results of the NASA Taskload (See Figure 5) by using the Wilcoxon signed rank test to assess the difference in workload of the various input methods. We presented the following results.

Hands-free target selection is more physically relaxing. *Hand gesture* had the highest physical load that was significantly larger than *ClenchClick* and *dwell* (Z = -2.889,-2.530, both $p < 0.05$). The two clench crossing methods had a medium load. We speculated that physical load was positively correlated with the intensity of the participant's physical actions. Gesture interaction required continuous movement of the arm, which led to a rapid increase in participant fatigue. Clench-crossing required the participant to keep the master muscle tense for a period, bringing additional body load. Relatively, short-term clenching and dwell clicking almost only involved subtle movement of the head.

Continuous confirmation motions are more mentally exhausting than transient motions. *Dwell* had the highest mental load. Participant 3 and Participant 7 indicated that *dwell* required them to hover the pointer inside a tiny target and not be deflected, which caused psychological pressure. *ClenchCrossingTarget* also suffered a high mental load because the participant was required to focus more than *ClenchCrossingEdge*. *ClenchClick* had the lowest mental load. Compared with other confirmation mechanisms, short-term clenching was energy-efficient and required minimal attention from the participant.

The point-and-click clench manner was perceived faster by the participants. The temporal load of *ClenchClick* was significantly lower than *hand gesture* and *dwell* (Z= -2.210,-1.999, both $p < 0.05$). We further calculated the average selection time. Although *ClenchClick* (T_{avg} = 0.58s) was significantly faster than *hand gesture* (T_{avg} = 1.39s), it had about the same speed as *dwell* (T_{avg} = 0.90s), *ClenchCrossingTarget*(T_{avg} = 0.73s), and *ClenchCrossingEdge* (T_{avg} = 0.46s). Therefore, we inferred that a point-and-click clench is subjectively faster and smoother for the user than dwell and crossing clench methods.

Failure of system detection is the main cause of user frustration. *Hand gesture* had the highest frustration level. We blamed it on the fact that participants sometimes had to repeat the hand tap multiple times before succeeding. The low recall rate made participants irritable and impatient due to repetitive tasks.

In general, ClenchClick had the lowest taskload. Mean workload and effort level both extensively reflected the comprehensive taskload during the task. *ClenchClick* was significantly lower ($Z = -2.607, -2.514$, both $p < 0.05$) than *hand gesture* and *dwell*. In addition, *ClenchClick* had lower mean workload than both clench-crossing actions. The task load of *hand gesture* was significantly higher than that of the other four input methods ($p < 0.05$).

4.2.5 Subjective Feedback on User Experience. Most participants agreed that both *dwell* and *ClenchClick* had a better target selection experience. Although *ClenchCrossingEdge* interactions gave them a smooth one-and-done feeling, prolonged use was tiring (Participant 1,2,4,10). In comparison, ClenchClick was a more elegant and relaxing interaction (Participant 2, 7, 9). *Dwell* was described to be the easiest-to-use input method, for prolonged use of *ClenchClick* also led to occlusal fatigue problems. However, *dwell* also had its own issues. Participant 6 claimed that *dwell*'s long hovering of the pointer over the target area was nerve-wracking. In addition, the experiment did not consider the case of multiple target selection, so participants were oblivious to the wrong selection problem of dwelling, which might make *dwell* potentially overrated. All participants agreed that *hand gesture* was the most physically and mentally exhausting input method. The low accuracy resulted in a high frustration level.

4.2.6 Subjective Feedback on Interaction Habits and Preferences. Interaction preferences varied from person to person. Although the experimenter had introduced all the standard interaction methods, some of the participants adjusted their actions from the standard ones. We observed a variety of actions when the participant completed the clench-based inputs. The *ClenchClick* action only required minimal clench force in most participants, but participant 9 said he tended to perform more exaggerated teeth-clicking actions. For clench crossing actions, participant 3 clenched his teeth during the entire fast movement of the pointer, rather than starting to clench his teeth when the pointer was near the button. Participant 4 preferred the clench crossing paradigm and discovered that a *ClenchCrossingTarget* action could still activate button selection events in the *ClenchClick* task. We found that the bite is a body part that people rarely try to control actively. On the contrary, *dwell* has meager learning costs, and most people can use their hands proficiently. Depending on the person, the bite input could be mastered quickly, while others took longer time to learn and adapt. A longer learning time may correlate to higher input efficiency of clench-based target selection.

4.2.7 Summary. We verified **H1** by comparing the experience of different clench-based inputs in a longer task. Compared to *ClenchCrossingEdge* and *ClenchCrossingTarget*, *ClenchClick* was more physically effortless, mentally relaxing, and subjectively faster for long time use. Therefore, we believe that *ClenchClick* is the most usable clench input action for target selection. We verified **H2** by revealing that *ClenchClick*, as the representative of clench-based actions, significantly outperformed *hand gesture* and *dwell* in total workload. It had less temporal and physical load than *hand gesture* and less temporal and mental load than *dwell*.

5 STUDY 2: PERFORMANCE EVALUATION

To evaluate the performance of ClenchClick, we compared it with two baseline methods, both with the head control paradigm but different selection confirmation mechanisms. We tested hand gesture-based (*hand gesture*) and hands-free (*dwell*) confirmation methods. In addition, to verify the effectiveness of the clench detection algorithm and the calibration process, we compared two implementations of ClenchClick, which were the general detection pipeline for averaged users, and the personalized detection pipeline with data collected in an extra calibration process. Instead of Fitts' Law experiment used in the first user experiment, we decided to compare the selection methods in a more realistic application to obtain an evaluation that was more reflective of real usage. Thus, we designed an AR Whac-A-Mole game which requires frequent target selection. We recorded the completion time, selection accuracy, and false positive rates of participants as the metrics for evaluation. We also visualized and analyzed the participant's head cursor movement trajectory and revealed the behavioral pattern behind different input actions. In this section, we present the following hypotheses:

H3: Since ClenchClick has a very low temporal load and frustration level, it may also achieve a high accuracy or input speed in real-life usage.

H4: The general ClenchClick detection method can suppress false positives while assuring true positives. A calibration phase further reduces false positives.

H5: For different input methods, users may have different mental models and behavioral characteristics, which affect the overall user experience of target selection.

5.1 Participants and Apparatus

We recruited ten participants ranging in age of 19 to 24 years old, consisting of four females and six males. All were students recruited from local campuses. They did not participate in the above-mentioned tests. None of them had prior experience with AR devices. We used the same hardware devices as that of the data collection phase.

5.2 Interaction Methods and Detection Algorithms

Similar with the user experience study (Study 1), we compared three target selection methods. Besides *ClenchClick*, a hand-based (*handgesture*) and a hands-free (*dwell*) baseline were included.

In addition, we compared the following two algorithm implementations.

- **The general algorithm**: ClenchClick algorithm implementation, which contains a rule-based model. Threshold T was pre-calculated to ensure the best detection performance on average users.
- **The personalized algorithm**: ClenchClick algorithm implementation, which contains a rule-based model. Each participant carried out the the calibration. Threshold T was predicted by a pre-trained model.

5.3 Experiment Design

The user experiment was an AR application based on a Whac-A-Mole game. There were eight mole holes distributed in eight directions in the scene. Among them, the holes in the positive top, positive bottom, positive left, and positive right directions were located in a plane 3.0 meters away from the participant and 0.6 meters away from the plane's center. The upper-left, upper-right, lower-left, and lower-right oriented holes were located in a plane 4.0 m away from the participant and 0.6 m away from the plane's center. We designed different directions and distances of mole appearance locations to simulate the participants selecting targets in different locations in space during daily use. The mole was a 0.2m*0.2m*0.2m 3D model. The angle of the mole in the field of view was about 2.86° to 3.82°.

When the game started, the mole randomly poked its body out of one of the eight mole holes and stayed for 5 seconds to ensure that the participant had enough time to complete the selection. After the participant hit the mole by performing the target selection gesture, the mole emitted a sound effect and retracted back into the hole. If the system failed to detect inputs within five seconds, the mole also returned to the hole by itself.

Our system recorded the selection time, accuracy, and false-positives as metrics for evaluation. In our experiment design, participants were instructed only to perform the selection action once for each mole target, so the accuracy was equal to the true-positive rate.

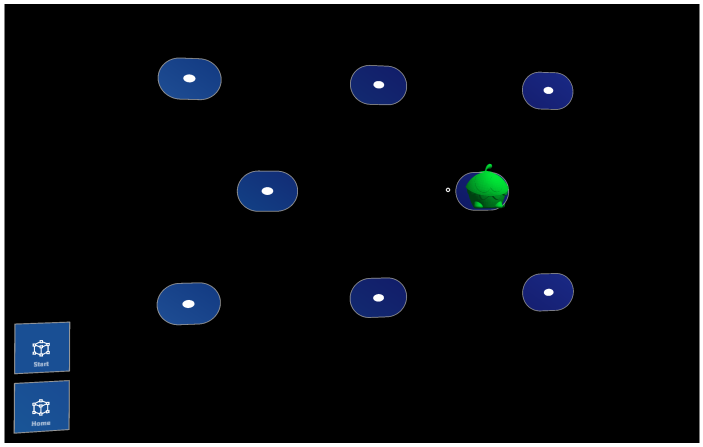

Fig. 7. The Whac-A-Mole Application. There are eight mole holes distributed in eight directions. The mole is about 0.2m*0.2m*0.2m, which randomly pokes its body out of one of the holes and waiting for the participant to hit it.

5.4 Procedure

Each participant completed the Whac-A-Mole target selection experiment using three input methods (*hand gesture*, *dwell*, and *ClenchClick*). For *ClenchClick*, participants had to repeat the task two times. Our system applied two different detection algorithms to recognize user input intentions.

Before the experiment began, we introduced the interactions in the AR glasses in a warm-up session. Participants practiced selecting objects in AR space until they thought they had mastered each of them. After that, participants completed four rounds (1**hand gesture*, 1**dwell*, and 2**ClenchClick*) of Whac-A-Mole. For each round, they went through 5 sessions. Each participant finished a total of 4 rounds*5 sessions*24 times = 480 target selections. Each participant completed the experiments in a different order to offset learning effects and fatigue. After finishing all experiments, they were asked to briefly describe the experience of all input methods.

5.5 Result

For the Whac-A-Mole task, we separately ran one-way RM-ANOVA tests for three interaction methods on the metrics of accuracy and selection time to verify **H3**. We used the results of the personalized detection algorithm as the representative for ClenchClick to compare with the other two input methods. We performed Tukey HSD tests as the post-hoc tests to identify significant differences between input methods and different algorithms.

5.5.1 Accuracy. Repeated measures analysis of variance implied a significant difference in the mean value of the accuracy of the different input methods ($F_{2,27} = 17.22$, $p < .00001$). Figure 8(a) shows the mean value of target selection accuracy. The recognition accuracy of *ClenchClick* (AVG = 98.89%, STD = 2.04%) and *dwell* (AVG = 99.72%, STD = 0.56%) were both high and significantly higher ($p < 0.01$) than that of *hand gesture* (AVG = 89.35%, STD = 6.46%). Participants felt that sometimes hand gestures were not detected, while they were pretty satisfied with the recognition accuracy of other target selection methods. They felt that their intentional inputs were at most not detected once or twice during the entire user experiment.

Since the accuracy was equal to the true-positive rate in our experiment, we found that both *ClenchCick* detection methods could robustly detect EMG signals and achieved a high detection recall of around 99%.

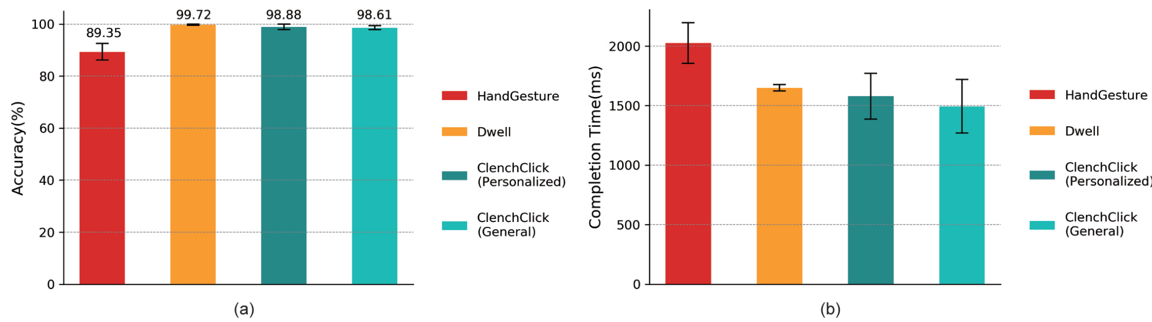

Fig. 8. Mean accuracy and completion time for different input methods and detection algorithms. The error bars represents the standard deviations.

5.5.2 Selection Time. Figure 8(b) shows the mean value of target selection time. There was a significant difference in the mean target selection time for different input methods ($F_{2,27}$ = 5.14, p < 0.05). *ClenchClick* (AVG = 1.58s, STD = 0.38s) and *dwell* (AVG = 1.65s, STD = 0.05s) were both fast. In addition, we observed that although the completion time of *ClenchClick* was slightly lower than *dwell*, the mean completion time distribution of the *dwell* was very concentrated. On the contrary, the completion time of *ClenchClick* had a more significant variance. Several participants subjectively felt *Clenchclick* was the fastest method. For participants who clenched faster (top 7 participants among all), the mean completion time was around 25% faster than dwell and reached 1317 ms. This indicated that clench input efficiency varied considerably between participants. *ClenchClick* was a more efficient input method for subjects who could control their occlusal muscles more comfortably. Additional training may improve the input efficiency of other users.

5.5.3 False-positive rate. In our experiments, we did not find any false-positive detection for gestures. False detection for dwell input was hard to define because the participants' pointers stayed somewhere on the screen at all times. Thus, we only compared the false-positive rates among algorithms. We defined one false detection as the system detecting the participant's clench action when the pointer was not in the target area. In real-life scenarios, false-positive triggers also require the user's pointer to happen to be in the target area, which occurs far less frequently than the false-positive rate obtained in this section.

There were significant differences in the number of false-positive detections per hour for different algorithms ($F_{2,27}$ = 8.62, p < 0.01). False detections per hour for *the personalized algorithm* (AVG = 9.3/h, STD = 10.6/h) algorithm was significantly less than that for *the general algorithm* (AVG = 15.3/h, STD = 13.9/h) (p<0.01). Results showed that the calibration phase significantly reduced the false-positive cases.

According to the subjective feedback, seven participants felt a minimal difference between the two algorithms, probably because their clench signals were already detected well enough using the general threshold. Four participants reported a significant difference between the two algorithms: two subjects reported occasional false positives before calibration. One subject often felt unable to complete the target selection task before calibration unless she clenched very hard. The phenomenon might be because their EMG signals were significantly stronger or weaker than the mean value of the vast majority. Our proposed calibration phase enhanced the interaction experience especially for these users.

5.6 Head Cursor Trajectory

To investigate participants' behavioral patterns of using different target selection methods, we analyzed participants' head cursor movement trajectory when performing *hand gesture*, *dwell*, and *ClenchClick*. Figure 9 shows

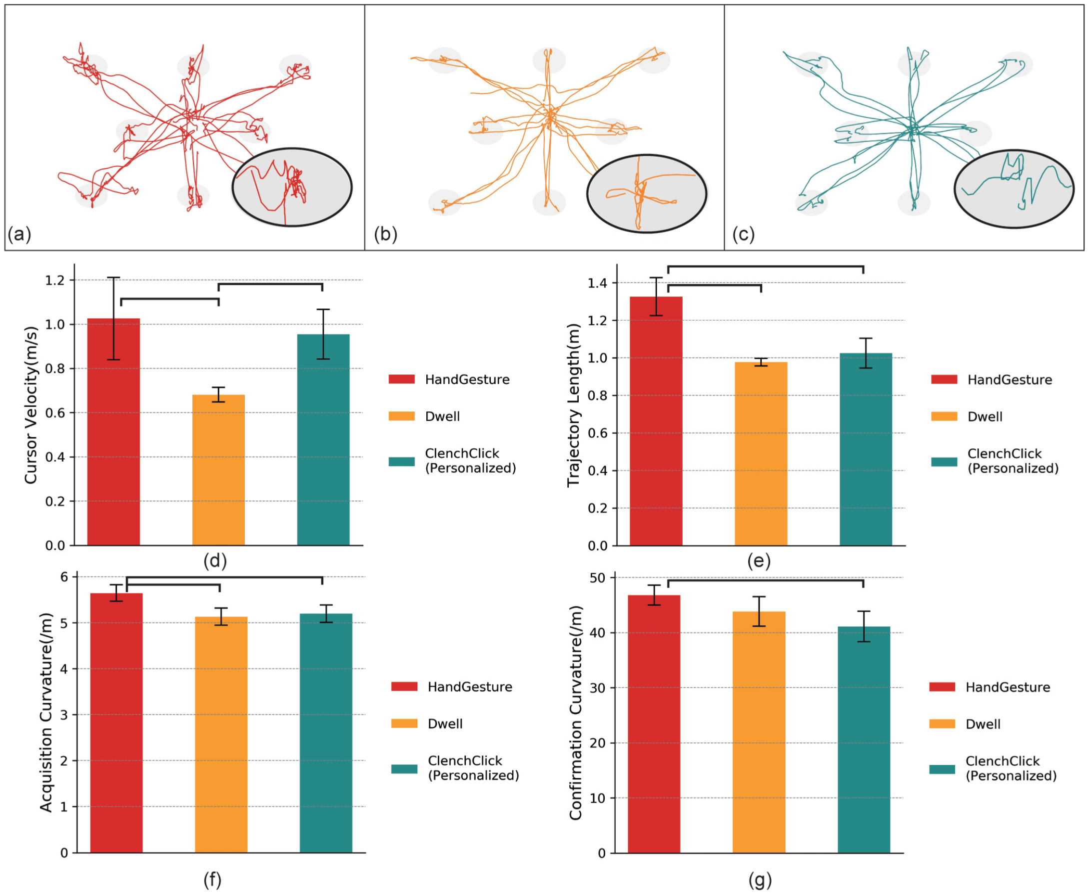

Fig. 9. The above three pictures show the participant's head cursor in one round of Whac A Mole game using different target selection methods. The trajectories magnified in an ellipse are trajectories in the right-bottom area during target confirmation. (a):head trajectory of hand gesture input (b):head trajectory of dwell (c):head trajectory of ClenchClick. Figure(d)-(g) show the mean length of trajectories, mean velocity of cursor and the mean curvature of target acquisition and confirmation trajectories.

the cursor trajectory examples, representing the participant's head orientation on the AR GUI. We divided the trajectories into two steps: moving the cursor to the target (**acquisition**) and confirming the click (**confirmation**), and analyzed them separately.

5.6.1 Acquisition. For the target acquisition period, we analyzed the average velocity of the head cursor, the average curvature of the acquisition trajectory, and the average length of the acquisition trajectory. We separately ran one-way RM-ANOVA tests for all metrics mentioned above, with the technique as the independent variable.

Participants felt more carefree in target acquisition when using ClenchClick and hand gesture. Although *ClenchClick* and *dwell* shared similar trajectory length, the cursor movement velocity was significantly greater ($p < 0.05$, $F_{1,18} = 12.07$) for *ClenchClick* and *hand gesture* input than for *dwell*. We speculated that dwelling required the cursor to remain in the button area for a sustained time after reaching the area. Therefore, the participant was wary of moving the cursor accidentally outside the target which cancelled the selection and dealt with target acquisition more cautiously and slowly. In contrast, the cursor moved about 40% faster when gesturing and clenching.

The participant's head cursor shook more frequently with hand gesture interaction, which we believed was a sign of high mental load. We used averaged curvature to quantitatively investigate the degree of bending of the cursor trajectory during acquisition. The curvature was significantly greater for *hand gesture* input than for *dwell* ($p < 0.05$, $F_{1,18} = 6.82$)and *ClenchClick*($p < 0.05$, $F_{1,18} = 7.81$), which we hypothesize was because participants needed to lift their hands to complete the gesture while moving the cursor, thus distracting them from cursor movement. The greater curvature reflected greater distraction and mental burden.

Taken together, with ClenchClick, participants were able to move the cursor quickly and smoothly with light concern, and had a more psychologically friendly experience during the target acquisition process.

5.6.2 Confirmation. Similarly, we used the average curvature to determine how distorted the head cursor trajectory during target confirmation was within the button. We separately ran one-way RM-ANOVA tests on confirmation curvature with the technique as the independent variable.

ClenchClick had the cleanest confirmation trajectory. Ideally, the head cursor should be relatively stationary when the participant performs a confirmation action. However, different target confirmation methods resulted in different subtle cursor movement patterns. When using *dwell* and *ClenchClick* as confirmation, the curvature of the head cursor trajectory was significantly smaller ($p < 0.05$, $F_{1,18} = 4.41$) than *hand gesture*. From Fig 9(b), the trajectory of *ClenchClick* during confirmation was more like a series of folded segments, which was caused by the subtle head nods that accompanied the clenching of teeth. Participant 8 believed that the slight head nod was a more relaxing movement than holding still for a while. In contrast, dwell trajectories and hand gesture trajectories (Fig 9(b)(c)) included more irregular curves resulting from the jitter when the participant's head tried to remain still in air.

5.7 Summary

H3 was partially valid as *ClenchClick* was more accurate and faster than *hand gesture*, while being comparable to *dwell* in accuracy and speed. We argued that the input speed of *ClenchClick* was significantly faster than *dwell* for users more used to clenching. For other users, speed may be improved after a bit of training. We verified **H4** by presenting the high true positive rate (99%) and acceptable false positives (15/h) of our general detection method. False positives reduced by 60% after a calibration. We verified **H5** by revealing facts on user behavior. *ClenchClick* allowed users to interact quickly and smoothly without heavy concerns and have a more psychologically friendly experience.

6 STUDY 3: EVALUATION IN REAL-WORLD TASKS

After evaluating the performance of ClenchClick in target selection, we decided to test its applicability in real-world applications. We tested three scenarios assumed to be beneficial from the features of hands-free, subtle and convenient interaction that ClenchClick potentially provides. We aimed to reveal more insights on how users interact with ClenchClick and verify how beneficial it can be in real-world tasks.

6.1 Apparatus and Participants

We recruited 12 participants aged 18 to 25 years old (4 females and 8 males) from campus and asked them to experience the tasks in three scenarios. Apart from *ClenchClick*, they also used *hand gesture* and *dwell* on completing the same tasks as control. After the experiment, they were asked to rate their subjective feelings along a 5-point Likert scale.

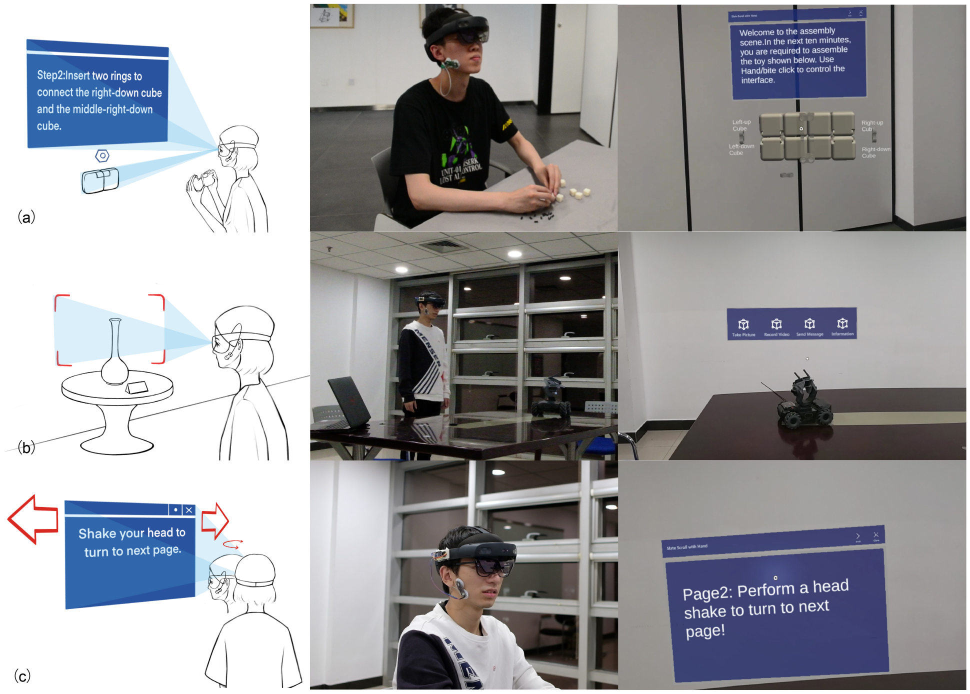

Fig. 10. ClenchClick application scenarios. (a) hands-free requirements: industrial assembly (b) social-friendly interaction: taking pictures (c) gestures beyond selection: clench activated head gesture input

6.2 Hands-free Input: Industrial Assembly

When users have occupied hands or physical limitations that prevent them from using their hands, we expect the hands-free feature of ClenchClick can supplement and provides efficient and feasible AR input with a low workload. In this sense, we tested the first application as the industrial assembly. The application shows a 2×4 decompression tesseract as an example of a part assembly teaching application often used in worker training scenarios (Figure 7 (a)). The part assembly assistant transformed 2D instructions into 3D part animations to more visually assist workers in learning the assembly process. Each step in the scene contained an animated demonstration, a text description, and a page turn button. The animation demonstration included eight steps, each progressively showing the relative position and installation method between the magic cubes and the connecting rings. Participants used *hand gesture*, *dwell* and ClenchClick to click buttons and do the page flip. Each participant needed to control the page turn and complete the assembly using three input methods. The experiment order of different participants was preset to eliminate the influence of the learning effect.

6.2.1 Result. Results are shown in Figure 11. We analyzed and used the Wilcoxon signed rank test in this section to test the significant difference in subjective evaluation metrics of different input methods.

When hands were occupied, ClenchClick and dwell were preferred by most participants because they eliminated the need for the switch between assembly work and interaction. Both *ClenchClick* and *dwell* had a smaller physical load (Z = -2.124, -2.058, both $p < 0.05$) and faster speed (Z = -2.546, -2.061, both $p < 0.05$) than *hand gesture*. All participants agreed that task switching was exhausting. Participant 1, 3, 4, 5, and 11 attributed this to the complexity of the gesture interaction itself and the muscle fatigue it brought. They

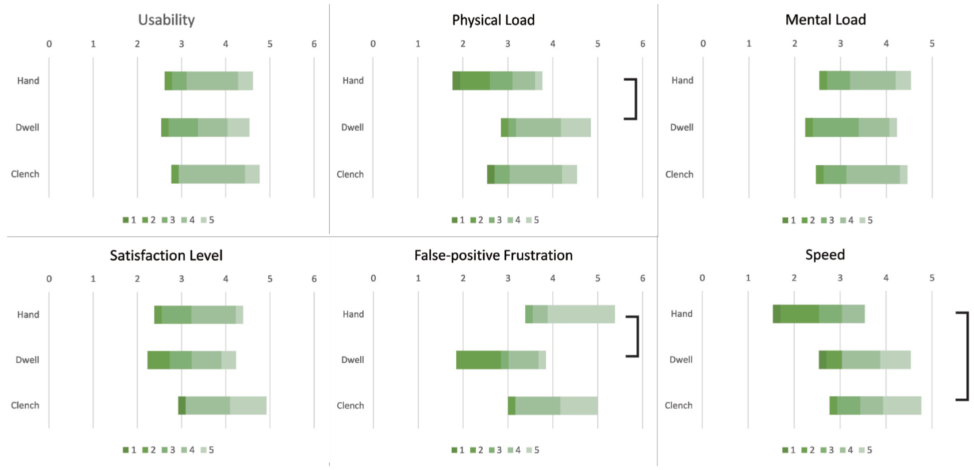

Fig. 11. The usability, physical load, satisfaction level, false-positive frustration and speed rating for different input methods when assembling parts. The error bars represent the standard deviations. The position of the bars in the stacked chart represents the average ratings of participants. Different colors show the distribution of participant ratings.

appreciated hands-free interaction that allowed them to do the assembling without putting down parts from time to time.

Between *ClenchClick* and *dwell*, **ClenchClick had a better user experience because of the lower frustration caused by false positives.** We did not discuss the false positives of *dwell* in previous studies because computers could not tell whether the dwell is intentional. Subjective feedback in real-world applications found that *dwell* suffered much more severe false positive problems than *ClenchClick*. *ClenchClick* had a significantly lower frustration level compared to *dwell* ($Z = -2.842$, $p < 0.05$). Several participants reported that if they stared at a button, *dwell* might accidentally trigger button events in succession, thereby directly skipping parts of the assembly animation. Subsequent undo operations brought a terrible user experience. In comparison, few participants reported clench false positives, even if they worked with both hands while talking to the interviewer from time to time. ClenchClick had a slightly smaller mental load than dwell. Participant 3 and 7 believed that keeping the head still for a long time is more mentally exhausting than clenching the teeth.

These findings suggested that ClenchClick was a more balanced input method that combines fast input speed with low error and frustration levels in assembly tasks.

6.3 Social-friendly Interaction: Photo-taking

An interaction method's social friendliness can drastically affect the user's mental load when completing the input action. Given that the holographic projection in the AR scene is not visible to onlookers, a large amount of motion input by the AR glasses user can be absurd and incomprehensible to them. The psychological load on the user is also significantly increased as a result. Taking pictures and recording videos are standard functions supported by AR glasses that users often use in public. We used photo-taking as the second task to explore whether ClenchClick provides a more subtle and private user experience, making it more social-friendly in public compared to hand gestures and gaze.

Considering that users frequently move around in space and adjust their view when taking images, we designed a 1×4 button toolbar that follows the user's view and hovers below the view (Figure 7 (b)). The toolbar was kept at a certain distance in front of the user at all times. It was fixed vertically and horizontally, allowing the

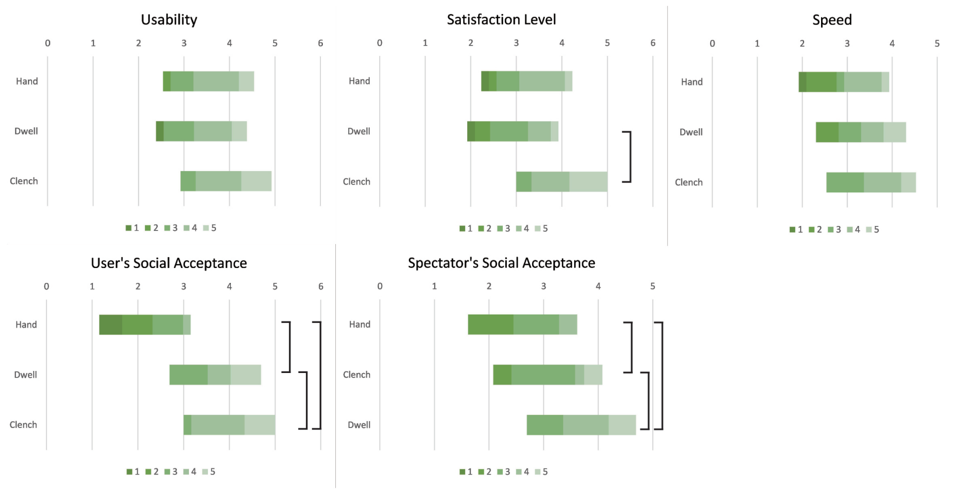

Fig. 12. The usability, satisfaction level, speed and social acceptance for different input methods when taking pictures in public. The error bars represent the standard deviations.

user's pointer to move around the toolbar to click on the buttons. After the user selected the "Picture" button, the system took a picture after the user performed a second clench. The user could also click the "Video" button to start or stop the camera. After taking the picture, the user could preview the image and choose a shortcut to send and share the picture.

In the user experiment session, participants stood in a public place and clicked the "shutter" button on the floating toolbar using three input methods: *hand gesture*, *dwell*, and *ClenchClick*. They were asked to complete taking still life, people (a photo of the other subjects), and landscape photos. They were also required to be photographed as models by other subjects or observed as bystanders. After completing the experiment, they completed the ratings in the questionnaire as both a user and a bystander.

6.3.1 Results. In the photo-taking scene, the usability and speed ratings were similar. Significant differences in social acceptance ($Z = -2.969, -2.000$, both $p < 0.05$) from both users' and spectators' perspectives led to different satisfaction levels. *ClenchClick* had the best satisfaction level.($Z = -2.412, -1.983$, both $p < 0.05$)

From the user's perspective, **ClenchClick could relieve the awful feelings that performing gestures on or staring at somebody brought about.** (Participant 2, 3, 4, 5, 7, 8, 10, 11) Participant 2, 3, 7 and 11 felt that performing mid-air gestures in public was bound to attract public attention. They described it as weird and silly. For *dwell*, participant 3 said it was very impolite to stare at someone or a place for a long time, and clenching confirmation could shorten such embarrassment. At the same time, the teeth-clenching action was very subtle and did not draw additional attention.

From the spectator's perspective, **ClenchClick was a more natural, unobtrusive interaction.** Several participants believed that *ClenchClick* was more inconspicuous and did not interfere with other people in public. On the other hand, comments on *hand gesture* were neutral. Spectators thought that they would feel intrigued and assumed that the participant was manipulating an advanced device. Participant 10 thought that if he had not been exposed to AR, it would remind him of the futuristic images in science fiction movies.

In addition, participants believed that gesture tapping distracted attention from models and objects, thus causing the framing to shift and affecting the final shot. Participant 9 reported that dwell's false triggers could cause him to inadvertently press the shutter by mistake, which lowered his satisfaction.

6.4 Convenient Control: Media Player

ClenchClick not only provides a hands-free target selection input method in AR scenarios, but can serve as activation for other gesture-based input in AR as well. Participants can combine teeth-clenching actions with gestures to send shortcut commands to AR glasses. We implemented a system which uses *Clenchclick* as the activation for head gesture recognition and designed a set of clench-activated micro-head gestures in AR scenarios. The user first performed the clench to activate the head gesture recognition system and then completed a small head gesture to send shortcut commands. We used the one-dollar algorithm to classify four predefined micro head gestures: nodding, shaking to the left, shaking to the right and drawing circles. The accuracy of the system recognition reached over 95%.

We tested the system in a media player application. The application was a head-gesture-based picture and video viewer. The user performed a clench-activated head shake to switch to the previous or next image. The user performed a clench-activated head nod to play and pause the video when watching it and drew a circle to zoom in on the image in view quickly. Each participant needed to go through several videos and pictures during a short user experiment. They followed the AR text instructions and performed clench-activated head gestures to send shortcut commands. In the control group, each participant was required to complete the same experiment using the head gesture without the clench activation. Participants were asked to fill in a simple subjective questionnaire to rate the two input methods. The accuracy and false positives were also recorded.

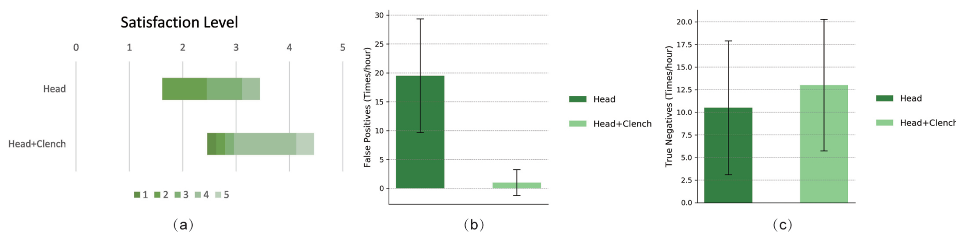

Fig. 13. (a): The satisfaction level for head gesture with/without clench activation. (b): False positives per hour for head gesture input with/without clench activation. (c): True negatives per hour for head gesture input with/without clench activation.

6.4.1 *Results.* **Clench activation gave participants a greater sense of control over head gesture interaction.**

Most participants liked to use head gestures to send short commands and preferred "head+clench" as a more easy-to-use and secure input method. Both input methods had similar accuracy. But with clench as activation, the false recognition of head movements without input intent was significantly reduced from 3.25 times/hour to about 0.17 times/hour. As a result, although both head gesture input with and without clench activation shared similar usability, the satisfaction level of clench-activated head gesture input was significantly higher than that of head gestures alone.

We concluded that combining the two modalities made the intent recognition more accurate and potentially allowed users to recognize more subtle head movements without many false positives, which lowered the user's physical load.

7 DISCUSSION AND LIMITATIONS

We have conducted three studies to evaluate the user experience and performance of ClenchClick in both laboratory and real-life scenarios. In this section, we discuss the limitations of our work and suggest new opportunities for future work.

7.1 User-centered Design of Clench Interactions

Our paper focused on the exploration and evaluation of clench-based input in target selection experiments. Therefore, we narrowed down from nine clench-based candidates through Gedanken experiments from various clench-based interactions to select actions that may be appropriate for the target selection task. We acknowledge that clench-based interactions with head action contain a considerable design space that can be used for both target selection and shortcut operations. The number of clenches (single, multiple) and the duration of clenches (instantaneous, continuous) can be combined with head actions (head pose, head direction, certain head gestures, the cursor's position with GUI elements) to define a larger action space. We only gave the clench-activated head gesture in the application as an example of combining the two. We hope future works may conduct a similar user elicitation study as Wobbrock[59] to collate more user-defined clench gestures and map them to actual input tasks in AR/VR.

7.2 Comfort

Although EMG sensing could support the whole experimental process of "Implementation-Design-Evaluation-Application" and ensured that the results of user experiments are not affected by accuracy and comfort, we still required the user to attach on-the-shelf fabric electrodes to the skin on the lateral side of the cheek. The electrode size is about $4cm \times 4cm$, which may bring a bit of foreign bodily sensation. Our ultimate vision is to make EMG sensors lighter and more comfortable to wear by using new electrode materials and designing suitable wearing methods. Two options, skin conformal polymer electrodes[50] and micro-structured metal electrodes, although not yet available in mass production, are comfortable and reusable dry electrodes that do not rely on the stickiness of conductive gel. An alternative that we are currently designing is an elastic stick-like structure on the side of the HMD. Smaller electrodes are placed at the end of the stick. By adjusting the stick's position, the user can make electrodes adhere correctly to the side of the cheek and generate little pressure.

7.3 Robustness

EMG sensing will only sense localized muscle contraction, and the torso movements such as walking and running will not affect signal detection. To test possible interference from facial muscles, we recorded 2 hours of EMG data when the user was having a conversation or laughing. Our detection algorithm was able to suppress most of the false positives. False detections in high-density talking scenes were about 20 times per hour. We believe that the chance that the user's head happened to be within the selected target was relatively low. In future wearable interaction scenarios, we believe that multimodal fusion with sensors of other channels can further improve sensing accuracy and reduce false positives. For example, IMU sensors can capture a small amplitude of head nod when users clench their teeth. Bone conduction and vibration sensors can recognize the sound and vibration when bumping teeth.

7.4 Limitation and Future Work

In the exploration phase, we did not technically implement various clench-based interactions but only used Gedanken experiments to obtain an imagined user experience. We acknowledge that the subjective ratings might not be completely precise, but we believe that the decision was valid, as we holistically considered the ratings, participant comments, and selection paradigms. We note that there remains more interaction exploration and technical implementation work to be done. All three of our studies were short-term experiments in laboratory or semi-live scenarios. It is worth exploring how users' behaviors adapt when using clench interaction in the long term. For example, there may be a learning effect for the clench interaction. There may exist fatigue issues during long-term use. Teeth clenching can also conflict with daily behavior such as eating. In addition, as technology advances, our sensing electrodes may become lighter and more comfortable. New low-cost, highly accurate, and comfortable sensing methods may also replace EMG as a better technical solution for clench interaction.

8 CONCLUSION

We explored teeth-clenching based target selection in Augmented Reality (AR), as the interaction subtlety can be beneficial to applications in which users' hands are occupied or in sensitive social contexts. We implemented an EMG-based teeth-clenching detection system (ClenchClick), where we adopted customized thresholds for different users. We first explored and compared the potential interaction design leveraging head movements and teeth clench in combination. We finalized the interaction to be a Point-and-Click manner with clenches as the confirmation method. We evaluated the task load and performance of ClenchClick by comparing it with two baseline methods in target selection tasks. Results showed that ClenchClick outperformed hand gestures in workload, physical load, accuracy and speed, and outperformed dwell in work load and temporal load. Lastly, through user studies, we revealed the advantage of ClenchClick in real-world tasks, including more efficient and accurate hands-free target selection, more satisfying interaction in public, and more controlled head gesture input.

ACKNOWLEDGMENTS

This work is supported by the Natural Science Foundation of China under Grant No.62132010, the Natural Science Foundation of China under Grant No.62102221, Beijing Academy of Artificial Intelligence (BAAI), and National Key R&D Program of China under Grant No.2020AAA0105200. Our work is also supported by Beijing Key Lab of Networked Multimedia, the Institute for Guo Qiang, Tsinghua University, and Institute for Artificial Intelligence, Tsinghua University (THUAI).

REFERENCES

[1] Johnny Accot and Shumin Zhai. 2002. More than dotting the i's—foundations for crossing-based interfaces. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. 73–80.
[2] AB Arsenault, DA Winter, and RG Marteniuk. 1986. Is there a 'normal'profile of EMG activity in gait? *Medical and Biological Engineering and Computing* 24, 4 (1986), 337–343.
[3] Daniel Ashbrook, Carlos Tejada, Dhwanit Mehta, Anthony Jiminez, Goudam Muralitharam, Sangeeta Gajendra, and Ross Tallents. 2016. Bitey: An exploration of tooth click gestures for hands-free user interface control. In *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services*. 158–169.
[4] Rabya Bahadur and Saeed Ur Rehman. 2018. A Robust and Adaptive Algorithm for Real-time Muscle Activity Interval Detection using EMG Signals.. In *BIOSIGNALS*. 89–96.
[5] Donna R Berryman. 2012. Augmented reality: a review. *Medical reference services quarterly* 31, 2 (2012), 212–218.
[6] Jonas Blattgerste, Patrick Renner, and Thies Pfeiffer. 2018. Advantages of eye-gaze over head-gaze-based selection in virtual and augmented reality under varying field of views. In *Proceedings of the Workshop on Communication by Gaze Interaction*. 1–9.
[7] Michael Bohan and Alex Chaparro. 1998. To click or not to click: A comparison of two target-selection methods for HCI. In *CHI 98 Conference Summary on Human Factors in Computing Systems*. 219–220.
[8] Paolo Bonato, Tommaso D'Alessio, and Marco Knaflitz. 1998. A statistical method for the measurement of muscle activation intervals from surface myoelectric signal during gait. *IEEE Transactions on biomedical engineering* 45, 3 (1998), 287–299.
[9] Jaekwang Cha, Jinhyuk Kim, and Shiho Kim. 2019. Hands-free user interface for AR/VR devices exploiting wearer's facial gestures using unsupervised deep learning. *Sensors* 19, 20 (2019), 4441.
[10] Ishan Chatterjee, Robert Xiao, and Chris Harrison. 2015. Gaze+ gesture: Expressive, precise and targeted free-space interactions. In *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*. 131–138.
[11] Victor Chen, Xuhai Xu, Richard Li, Yuanchun Shi, Shwetak Patel, and Yuntao Wang. 2021. Understanding the Design Space of Mouth Microgestures. *arXiv preprint arXiv:2106.00931* (2021).
[12] Jingyuan Cheng, Ayano Okoso, Kai Kunze, Niels Henze, Albrecht Schmidt, Paul Lukowicz, and Koichi Kise. 2014. On the tip of my tongue: a non-invasive pressure-based tongue interface. In *Proceedings of the 5th Augmented Human International Conference*. 1–4.
[13] SS Daud and R Sudirman. 2015. Butterworth bandpass and stationary wavelet transform filter comparison for electroencephalography signal. In *2015 6th international conference on intelligent systems, modelling and simulation*. IEEE, 123–126.
[14] David Dearman, Amy Karlson, Brian Meyers, and Ben Bederson. 2010. Multi-modal text entry and selection on a mobile device. In *Proceedings of Graphics Interface 2010*. 19–26.

[15] Denis V Dorozhkin and Judy M Vance. 2002. Implementing speech recognition in virtual reality. In *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Vol. 36215. 61–65.

[16] Augusto Esteves, Eduardo Velloso, Andreas Bulling, and Hans Gellersen. 2015. Orbits: Gaze interaction for smart watches using smooth pursuit eye movements. In *Proceedings of the 28th annual ACM symposium on user interface software & technology*. 457–466.

[17] Dario Farina, Ning Jiang, Hubertus Rehbaum, Aleš Holobar, Bernhard Graimann, Hans Dietl, and Oskar C Aszmann. 2014. The extraction of neural information from the surface EMG for the control of upper-limb prostheses: emerging avenues and challenges. *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 22, 4 (2014), 797–809.

[18] Anna Maria Feit, Shane Williams, Arturo Toledo, Ann Paradiso, Harish Kulkarni, Shaun Kane, and Meredith Ringel Morris. 2017. Toward everyday gaze input: Accuracy and precision of eye tracking and implications for design. In *Proceedings of the 2017 Chi conference on human factors in computing systems*. 1118–1130.

[19] Mayank Goel, Chen Zhao, Ruth Vinisha, and Shwetak N Patel. 2015. Tongue-in-cheek: Using wireless signals to enable non-intrusive and flexible facial gestures detection. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. 255–258.

[20] Scott W Greenwald, Luke Loreti, Markus Funk, Ronen Zilberman, and Pattie Maes. 2016. Eye gaze tracking with google cardboard using purkinje images. In *Proceedings of the 22nd ACM Conference on Virtual Reality Software and Technology*. 19–22.

[21] JA Guerrero and JE Macías-Díaz. 2019. An optimal bayesian threshold method for onset detection in electric biosignals. *Mathematical biosciences* 309 (2019), 12–22.

[22] José A Guerrero and Jorge Eduardo Macías-Díaz. 2020. A threshold selection criterion based on the number of runs for the detection of bursts in EMG signals. *Biomedical Signal Processing and Control* 57 (2020), 101699.

[23] Laura Guidetti, Gianfranco Rivellini, and Francesco Figura. 1996. EMG patterns during running: Intra-and inter-individual variability. *Journal of Electromyography and Kinesiology* 6, 1 (1996), 37–48.

[24] Hiroyuki Hakoda, Wataru Yamada, and Hiroyuki Manabe. 2017. Eye tracking using built-in camera for smartphone-based HMD. In *Adjunct Publication of the 30th Annual ACM Symposium on User Interface Software and Technology*. 15–16.

[25] Benjamin Hatscher, Maria Luz, Lennart E Nacke, Norbert Elkmann, Veit Müller, and Christian Hansen. 2017. GazeTap: towards hands-free interaction in the operating room. In *Proceedings of the 19th ACM international conference on multimodal interaction*. 243–251.

[26] Hwan Heo, Minho Lee, Sungjei Kim, and Youngbae Hwang. 2020. Gaze+ Gesture Interface: Considering Social Acceptability. In *2020 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*. IEEE, 690–691.

[27] Anke Huckauf and Mario H Urbina. 2011. Object selection in gaze controlled systems: What you don't look at is what you get. *ACM Transactions on Applied Perception (TAP)* 8, 2 (2011), 1–14.

[28] Jinhyuk Kim, Jaekwang Cha, Hojun Lee, and Shiho Kim. 2017. Hand-free natural user interface for VR HMD with IR based facial gesture tracking sensor. In *Proceedings of the 23rd ACM Symposium on Virtual Reality Software and Technology*. 1–2.

[29] RFM Kleissen, JH Buurke, J Harlaar, and G Zilvold. 1998. Electromyography in the biomechanical analysis of human movement and its clinical application. *Gait & posture* 8, 2 (1998), 143–158.

[30] Trupti M Kodinariya and Prashant R Makwana. 2013. Review on determining number of Cluster in K-Means Clustering. *International Journal* 1, 6 (2013), 90–95.

[31] Marion Koelle, Swamy Ananthanarayan, and Susanne Boll. 2020. Social acceptability in HCI: A survey of methods, measures, and design strategies. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. 1–19.

[32] Devender Kumar and Amit Sharma. 2016. Electrooculogram-based virtual reality game control using blink detection and gaze calibration. In *2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*. IEEE, 2358–2362.

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

[34] DoYoung Lee, Youryang Lee, Yonghwan Shin, and Ian Oakley. 2018. Designing socially acceptable hand-to-face input. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology*. 711–723.

[35] Siyoung Lee, Junsoo Kim, Inyeol Yun, Geun Yeol Bae, Daegun Kim, Sangsik Park, Il-Min Yi, Wonkyu Moon, Yoonyoung Chung, and Kilwon Cho. 2019. An ultrathin conformable vibration-responsive electronic skin for quantitative vocal recognition. *Nature communications* 10, 1 (2019), 1–11.

[36] Roberto Merletti and Silvia Muceli. 2019. Tutorial. Surface EMG detection in space and time: Best practices. *Journal of Electromyography and Kinesiology* 49 (2019), 102363.

[37] Andrea Merlo, Dario Farina, and Roberto Merletti. 2003. A fast and reliable technique for muscle activity detection from surface EMG signals. *IEEE transactions on biomedical engineering* 50, 3 (2003), 316–323.

[38] S Micera, Giancarlo Vannozzi, AngeloM Sabatini, and Paulo Dario. 2001. Improving detection of muscle activation intervals. *IEEE Engineering in medicine and Biology Magazine* 20, 6 (2001), 38–46.

[39] Microsoft. 2021. *Microsoft HoloLens: Mixed Reality Technology for Business*. https://www.microsoft.com/en-us/hololens

[40] Katsumi Minakata, John Paulin Hansen, I Scott MacKenzie, Per Bækgaard, and Vijay Rajanna. 2019. Pointing by gaze, head, and foot in a head-mounted display. In *Proceedings of the 11th ACM symposium on eye tracking research & applications*. 1–9.

[41] Pedro Monteiro, Guilherme Gonçalves, Hugo Coelho, Miguel Melo, and Maximino Bessa. 2021. Hands-free interaction in immersive virtual reality: A systematic review. *IEEE Trans. Vis. Comput. Graph.* 27, 5 (2021), 2702–2713.
[42] Takuro Nakao, Yun Suen Pai, Megumi Isogai, Hideaki Kimata, and Kai Kunze. 2018. Make-a-face: a hands-free, non-intrusive device for tongue/mouth/cheek input using emg. In *ACM SIGGRAPH 2018 Posters*. 1–2.
[43] Phuc Nguyen, Nam Bui, Anh Nguyen, Hoang Truong, Abhijit Suresh, Matt Whitlock, Duy Pham, Thang Dinh, and Tam Vu. 2018. Tyth-typing on your teeth: Tongue-teeth localization for human-computer interface. In *Proceedings of the 16th Annual International Conference on Mobile Systems, Applications, and Services*. 269–282.
[44] Tomi Nukarinen, Jari Kangas, Oleg Špakov, Poika Isokoski, Deepak Akkil, Jussi Rantala, and Roope Raisamo. 2016. Evaluation of HeadTurn: An interaction technique using the gaze and head turns. In *Proceedings of the 9th Nordic Conference on Human-Computer Interaction*. 1–8.
[45] Oculus. 2021. *Oculus|Headsets,Games and Equipments*. https://www.oculus.com
[46] Jay Prakash, Zhijian Yang, Yu-Lin Wei, Haitham Hassanieh, and Romit Roy Choudhury. 2020. EarSense: earphones as a teeth activity sensor. In *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking*. 1–13.
[47] Javier Ramirez, Juan Manuel Górriz, and José Carlos Segura. 2007. Voice activity detection. fundamentals and speech recognition system robustness. *Robust speech recognition and understanding* 6, 9 (2007), 1–22.
[48] Mamun Bin Ibne Reaz, M Sazzad Hussain, and Faisal Mohd-Yasin. 2006. Techniques of EMG signal analysis: detection, processing, classification and applications. *Biological procedures online* 8, 1 (2006), 11–35.
[49] Javier San Agustin, John Paulin Hansen, Dan Witzner Hansen, and Henrik Skovsgaard. 2009. Low-cost gaze pointing and EMG clicking. In *CHI'09 Extended Abstracts on Human Factors in Computing Systems*. 3247–3252.
[50] Flurin Stauffer, Moritz Thielen, Christina Sauter, Séverine Chardonnens, Simon Bachmann, Klas Tybrandt, Christian Peters, Christofer Hierold, and Janos Vörös. 2018. Skin conformal polymer electrodes for clinical ECG and EEG recordings. *Advanced healthcare materials* 7, 7 (2018), 1700994.
[51] Kazuhiro Taniguchi, Hisashi Kondo, Mami Kurosawa, and Atsushi Nishikawa. 2018. Earable TEMPO: a novel, hands-free input device that uses the movement of the tongue measured with a wearable ear sensor. *Sensors* 18, 3 (2018), 733.
[52] AJ Thexton. 1996. A randomisation method for discriminating between signal and noise in recordings of rhythmic electromyographic activity. *Journal of neuroscience methods* 66, 2 (1996), 93–98.
[53] Andrea Tigrini, Alessandro Mengarelli, Stefano Cardarelli, Sandro Fioretti, and Federica Verdini. 2020. Improving EMG Signal Change Point Detection for Low SNR by Using Extended Teager-Kaiser Energy Operator. *IEEE Transactions on Medical Robotics and Bionics* 2, 4 (2020), 661–669.
[54] Andries Van Der Bilt, Anneke Tekamp, Hilbert Van Der Glas, and Jan Abbink. 2008. Bite force and electromyograpy during maximum unilateral and bilateral clenching. *European journal of oral sciences* 116, 3 (2008), 217–222.
[55] DWF Van Krevelen and Ronald Poelman. 2010. A survey of augmented reality technologies, applications and limitations. *International journal of virtual reality* 9, 2 (2010), 1–20.
[56] Tomás Vega Gálvez, Shardul Sapkota, Alexandru Dancu, and Pattie Maes. 2019. Byte. it: discreet teeth gestures for mobile device interaction. In *Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems*. 1–6.
[57] Ker-Jiun Wang, Quanbo Liu, Yifan Zhao, Caroline Yan Zheng, Soumya Vhasure, Quanfeng Liu, Prakash Thakur, Mingui Sun, and Zhi-Hong Mao. 2018. Intelligent wearable virtual reality (VR) gaming controller for people with motor disabilities. In *2018 IEEE International Conference on Artificial Intelligence and Virtual Reality (AIVR)*. IEEE, 161–164.
[58] Xiangyu Wang, Soh K Ong, and Andrew YC Nee. 2016. A comprehensive survey of augmented reality assembly research. *Advances in Manufacturing* 4, 1 (2016), 1–22.
[59] Jacob O Wobbrock, Meredith Ringel Morris, and Andrew D Wilson. 2009. User-defined gestures for surface computing. In *Proceedings of the SIGCHI conference on human factors in computing systems*. 1083–1092.
[60] Lanyi Xu and Andy Adler. 2004. An improved method for muscle activation detection during gait. In *Canadian Conference on Electrical and Computer Engineering 2004 (IEEE Cat. No. 04CH37513)*, Vol. 1. IEEE, 357–360.
[61] Xuhai Xu, Chun Yu, Anind K Dey, and Jennifer Mankoff. 2019. Clench Interface: Novel Biting Input Techniques. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. 1–12.
[62] Yukang Yan, Yingtian Shi, Chun Yu, and Yuanchun Shi. 2020. Headcross: Exploring head-based crossing selection on head-mounted displays. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4, 1 (2020), 1–22.
[63] Yukang Yan, Chun Yu, Xiaojuan Ma, Shuai Huang, Hasan Iqbal, and Yuanchun Shi. 2018. Eyes-Free Target Acquisition in Interaction Space around the Body for Virtual Reality. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (Montreal QC, Canada) *(CHI '18)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3173574.3173616
[64] Yukang Yan, Chun Yu, Xin Yi, and Yuanchun Shi. 2018. HeadGesture: Hands-Free Input Approach Leveraging Head Movements for HMD Devices. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 2, 4, Article 198 (dec 2018), 23 pages. https://doi.org/10.1145/3287076
[65] Chun Yu, Yizheng Gu, Zhican Yang, Xin Yi, Hengliang Luo, and Yuanchun Shi. 2017. Tap, dwell or gesture? Exploring head-based text entry techniques for HMDs. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. 4479–4488.

[66] Difeng Yu, Hai-Ning Liang, Feiyu Lu, Vijayakumar Nanjappan, Konstantinos Papangelis, and Wei Wang. 2018. Target Selection in Head-Mounted Display Virtual Reality Environments. *J. Univers. Comput. Sci.* 24, 9 (2018), 1217–1243.

[67] Qiao Zhang, Shyamnath Gollakota, Ben Taskar, and Raj PN Rao. 2014. Non-intrusive tongue machine interface. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2555–2558.

[68] Lin Zhong, Dania El-Daye, Brett Kaufman, Nick Tobaoda, Tamer Mohamed, and Michael Liebschner. 2007. OsteoConduct: Wireless body-area communication based on bone conduction. In *Proceedings of the ICST 2nd international conference on Body area networks*. 1–8.