

# PHANTOM: Punch & Head Action Networked Tracking with Omni-device Motion Sensing

Geonho Kim

UNIST

Ulsan, Republic of Korea  
rlarjsgh02@unist.ac.kr

Joocho Sohn

UNIST

Ulsan, Republic of Korea  
tigerjoocho11@unist.ac.kr

Jiho Park

UNIST

Ulsan, Republic of Korea  
batiger00@unist.ac.kr

Jiseung Jeong

UNIST

Ulsan, Republic of Korea  
wjdwltmd1151@unist.ac.kr

## Abstract

Most boxing trackers measure punches but leave defensive movement unobserved. We present PHANTOM, a mobile sensing system that recognizes offense and defense during shadowboxing using three devices that many users already own: an Apple Watch, an iPhone, and AirPods Pro. PHANTOM aligns their inertial streams, detects active motion windows, extracts a fixed feature representation, and produces an ordered sequence of six punch classes and four slip or roll classes. All session analysis and personalization are performed locally on the iPhone. Because boxing style varies strongly across people, PHANTOM augments a subject-independent 1-nearest-neighbor model with one labeled demonstration per action. The demonstration is transformed by the frozen factory pipeline and appended to the reference set, requiring no on-device retraining. We evaluate 300 synchronized recordings from five participants using leave-one-subject-out and file-separated calibration protocols. The selected offline model improves from 0.637 to 0.916 pooled macro-F1 after one demonstration per action, while the deployment-oriented model improves from 0.581 to 0.887. Sensor ablation shows that wrist and hand motion provide most of the multi-device gain, with all three devices achieving the best overall score of 0.678. Continuous combinations remain harder than isolated actions, exposing co-articulation and segmentation as the main limitations of the current system.

## Keywords

boxing, wearable sensing, inertial measurement unit, activity recognition, personalization

## 1 Introduction

Boxing trackers see only half of a training session. Punch-oriented systems can report counts, speed, or power, but slips and rolls are usually absent from the record. Some products also depend on a dedicated bag or purpose-built sensor, making them unsuitable for unrestricted shadowboxing. A boxer can therefore receive detailed feedback about offense while the defensive half of the same movement sequence remains unmeasured.

This gap is partly a sensing problem. Punches create large signals at the wrist and hand, whereas slips and rolls are expressed through the head and torso. A single sensor cannot observe all of these body regions equally well. The project therefore asks the following

question: *Can a system measure both offense and defense, anywhere, using only devices the user already owns?*

PHANTOM addresses the question with an Apple Watch, an iPhone, and AirPods Pro. The Watch records lead-wrist motion, the iPhone is held in the rear hand and acts as the session hub, and AirPods Pro record head motion. Their accelerometer and gyroscope streams are aligned and analyzed after the session to recognize ten actions: lead and rear jab, hook, and uppercut, plus left and right slip and roll. The output is not only a class count but an ordered action sequence that can represent a complete shadowboxing session.

The second challenge is individual variation. Stance, handedness, trajectory, speed, range of motion, and device orientation all alter the observed signal. A model trained only on other participants consequently provides an incomplete factory model. PHANTOM uses one-shot personalization: the user demonstrates each action once, the captured feature vector passes through the same frozen preprocessing pipeline, and the vector is appended to a 1-nearest-neighbor (1-NN) reference set. This design trades a small amount of peak model accuracy for immediate, server-free adaptation on the phone.

The contributions of this project are:

- an end-to-end mobile system that combines Watch, phone, and earbud IMUs to recognize six offensive and four defensive actions;
- a synchronized, peak-centered recognition pipeline that runs locally and supports personalization without retraining;
- an evaluation over five participants that separates factory training, calibration, and testing by participant and recording file; and
- an analysis of sensor contribution, one-shot personalization, and the transfer from isolated actions to continuous combinations.

The central result tells a before-and-after story. For an unseen participant, pooled macro-F1 is 0.637 before calibration and 0.916 after one demonstration per action. The result shows that commodity devices can cover both offense and defense, but the lower performance on continuous combinations also shows that isolated demonstrations do not yet capture every condition encountered in real boxing.

**Table 1: Scope of representative boxing-sensing approaches.**

| Approach        | Sensing               | Action scope                  | Adaptation          |
|-----------------|-----------------------|-------------------------------|---------------------|
| BoxerSense      | Wearable IMUs         | Punches                       | Fixed model         |
| Earable defense | Earbud IMU            | Defensive gestures            | Fixed model         |
| PHANTOM         | Watch, phone, earbuds | Six punches and four defenses | One demo per action |

## 2 Related Work

### 2.1 Wearable Activity Recognition

Wearable activity recognition commonly converts inertial time series into window-level statistics and applies a conventional or learned classifier [4]. Multiple commodity devices can provide complementary observations of different body regions. IMUPoser, for example, combines IMUs embedded in phones, watches, and earbuds to estimate full-body pose [3]. PHANTOM uses a similar class of devices but targets sparse boxing-action segmentation and discrete action recognition rather than continuous pose reconstruction.

### 2.2 Boxing Motion Recognition

Most boxing sensing research emphasizes punches. BoxerSense uses wearable IMUs and statistical features to detect and classify punch types [1]. Its main evaluation focuses on three rear-hand punches under a controlled collection protocol. This establishes that IMUs can identify boxing technique, but does not provide a complete offense-and-defense vocabulary.

Defensive motion has been studied separately. Sepanosian and Incel use an earable IMU and time-series matching to recognize defensive boxing gestures [5]. Their work motivates head sensing, while PHANTOM asks a different systems question: whether head, wrist, and hand signals can be combined into one action stream containing both punch and defense labels.

### 2.3 Personalization

Subject-independent activity recognition often degrades for unseen users because anatomy, technique, sensor orientation, and execution speed change the input distribution. On-device few-shot learning has been proposed as a bridge between generalization and personalization [2]. PHANTOM adopts a deliberately simple instance-based form of this idea. Factory preprocessing remains fixed, and each user demonstration becomes another reference in the transformed feature space. Unlike a parametric model, 1-NN requires neither an optimizer nor complete model retraining when a new demonstration is added.

### 2.4 Positioning

Table 1 exposes three gaps that motivate PHANTOM. Punch-oriented work provides detailed offense labels but omits defense; earable work observes defense but does not integrate it with bilateral punch recognition; and both directions use a fixed model despite strong

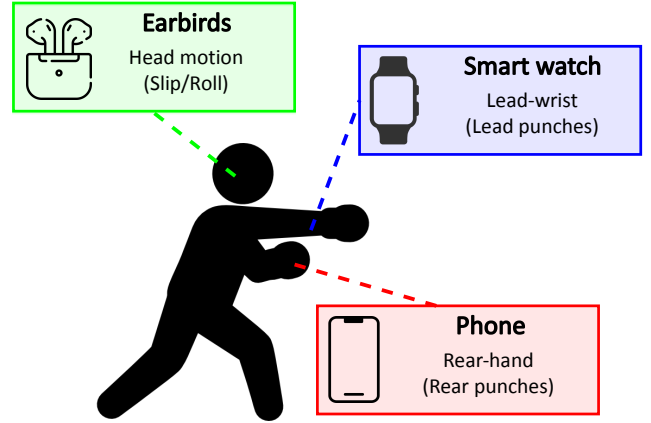


**Figure 1: PHANTOM device placement.** A smart watch captures lead-wrist motion, a phone captures rear-hand motion, and motion-sensing earbuds capture head motion for offensive and defensive action recognition.

cross-user motion variation. Among the reviewed systems, PHANTOM is the only one that combines all three design goals: offense-and-defense coverage, complementary sensing from devices already carried or worn, and one-demonstration local adaptation.

PHANTOM does not claim that peak segmentation, statistical IMU features, or nearest-neighbor personalization are individually new. Its position is a systems contribution: integrating these known components into an end-to-end mobile tracker and evaluating where the integration succeeds or fails, particularly for an unseen user and for continuous combinations.

## 3 System Design and Data Collection

PHANTOM is designed around three devices that observe complementary body regions: an iPhone 11, an Apple Watch SE (3rd generation), and AirPods Pro (3rd generation). Figure 1 shows the device roles and the placement on the body. The Watch is worn on the lead wrist, the iPhone is held securely in the rear hand, and the AirPods are worn normally. The iPhone also coordinates the session and executes the final recognition pipeline.

### 3.1 Sensing and Communication

Each device supplies three-axis acceleration and angular velocity. At session start, the iPhone broadcasts an anchor timestamp. Device-local timestamps are converted into elapsed time from this anchor, giving the three streams a common session timeline. The Watch maintains an active workout session so that motion collection can continue reliably. It uses `WCSession` live messages for immediate delivery and file transfer as a reliable fallback. The iPhone obtains its own device motion and headphone motion through Core Motion.

All raw samples are retained until the session ends. This is important for two reasons. First, intermittent Watch connectivity does not invalidate an otherwise usable recording. Second, the complete session can be reanalyzed after changes to segmentation or classification without recollecting motion data. No cloud service is required for inference or calibration.

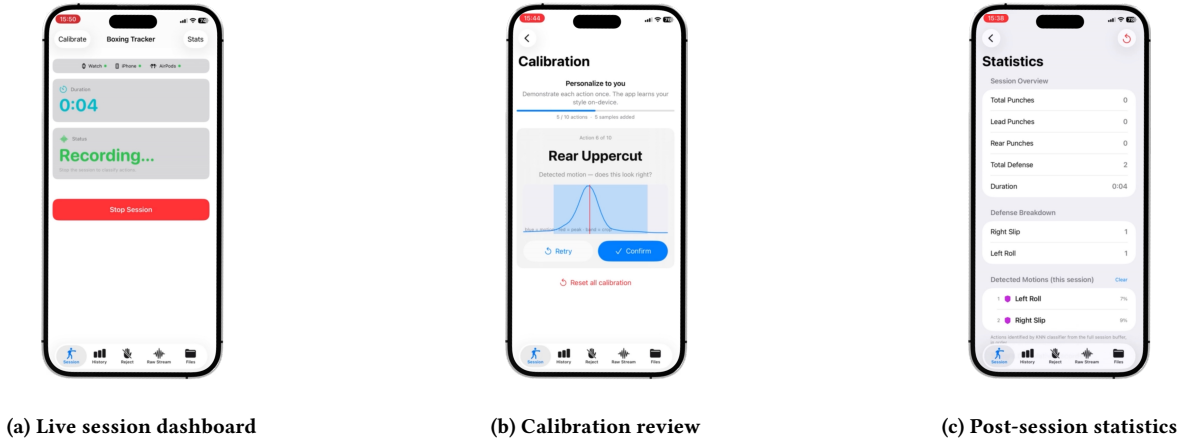


Figure 2: PHANTOM application workflow. (a) The session dashboard displays device connectivity and recording status. (b) The calibration interface lets the user review the detected peak and crop before adding a personal reference. (c) The statistics view presents offense and defense summaries and the ordered action sequence.

Table 2: PHANTOM action vocabulary.

| Group   | Base actions        | Direction or side |
|---------|---------------------|-------------------|
| Offense | Jab, hook, uppercut | Lead, rear        |
| Defense | Slip, roll          | Left, right       |
| Total   | 10 classes          |                   |

### 3.2 Action Vocabulary

The recognition vocabulary contains ten direction-sensitive actions. The offense set consists of jab, hook, and uppercut for the lead and rear side. The defense set contains left and right variants of slip and roll. This vocabulary is more demanding than a binary offense/defense decision because the model must separate punch type, body side, and defensive direction.

### 3.3 Collection Protocol

Five independent participants contributed the final dataset. Each recording began with an intentional jump used by the offline collection pipeline as an alignment and quality-control cue. This jump is part of the data collection protocol; the deployed application uses the shared anchor timestamp instead and does not require a jump during ordinary use.

For isolated actions, each participant recorded three files for every class, with five repetitions in each file. For continuous motion, each participant recorded ten predefined combinations three times. The combinations contained two or three actions and mixed offense with defense, such as a lead jab followed by a slip and rear hook. In total, the retained cohort contains 150 isolated-action files and 150 combination files.

## 3.4 Application Workflow

The user checks device availability, starts the session from the iPhone, and performs shadowboxing. When the session stops, PHANTOM aligns the buffered streams, recognizes the action sequence, and groups the result into lead punches, rear punches, and defense. The current prototype performs post-session analysis only and does not provide real-time feedback. A separate calibration screen prompts one demonstration for each action and stores the resulting references locally across application launches.

Figure 2 shows this workflow in the implemented application. The session dashboard verifies device connectivity and recording status. During calibration, the user can inspect the detected motion peak and selected crop before accepting the demonstration. After a session, the statistics view reports offense and defense counts together with the recognized actions in temporal order.

## 4 Recognition and Personalization Pipeline

Figure 3 summarizes how PHANTOM turns three IMU streams into one action sequence. The main inference path collects raw head, lead-wrist, and rear-hand motion, synchronizes the streams, segments and transforms active windows, and assigns an action label using nearest-neighbor classification. Calibration, described later in this section, reuses the same representation path before adding a user reference.

### 4.1 Synchronization and Active Windows

PHANTOM first places the asynchronous device samples on the shared session timeline. For each device, it combines acceleration and angular-velocity magnitudes into a smoothed activity envelope and normalizes the envelope with robust session statistics. This reduces sensitivity to different signal scales, sensor placement, and occasional extreme values.

Let  $z_{\omega,d}^+(t)$  and  $z_{a,d}^+(t)$  denote the positive robust scores of angular velocity and dynamic acceleration for device  $d$ , where each signal is centered by its median, scaled using the median absolute

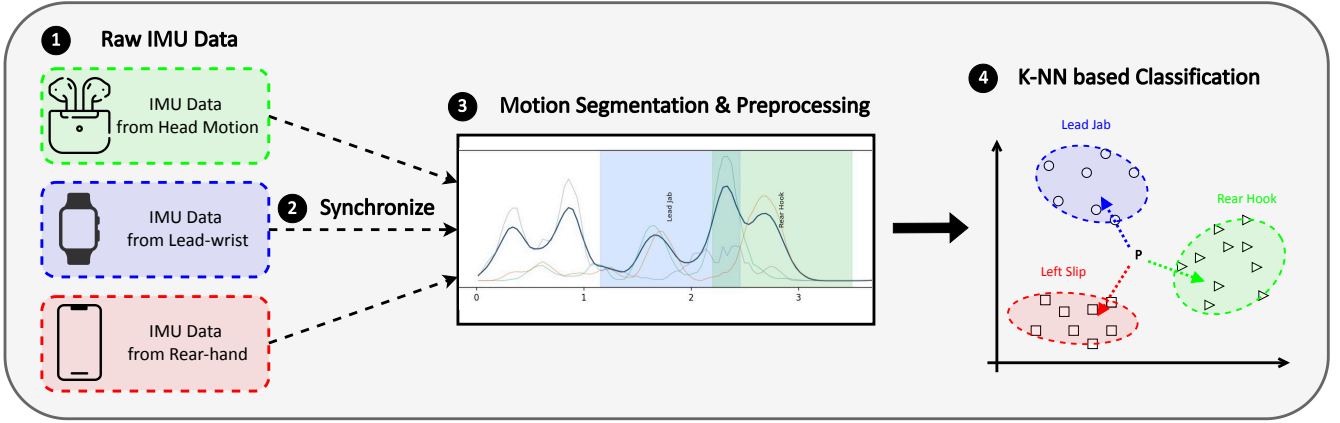


**Figure 3: PHANTOM inference pipeline.** Raw IMU streams from head, lead-wrist, and rear-hand motion are synchronized, segmented, converted into a frozen feature representation, and classified by nearest-neighbor matching.

deviation (MAD), and clipped at zero. The activity envelope of device  $d$  is

$$r_d(t) = \text{clip}\left(\alpha z_{\omega,d}^+(t) + (1 - \alpha) z_{a,d}^+(t), 0, C\right). \quad (1)$$

Here,  $\text{clip}(\cdot, 0, C)$  limits unusually large values to the fixed range  $[0, C]$  so that a single sensor spike does not dominate fusion, and  $\alpha$  controls the balance between angular velocity and acceleration. The device envelopes are fused as

$$s(t) = \beta \max_d r_d(t) + (1 - \beta) \text{mean}_d r_d(t). \quad (2)$$

The maximum term allows strong motion at one body region to identify an action, while the mean term retains supporting evidence shared across devices;  $\beta$  controls the balance between these two terms.

An adaptive threshold finds prominent peaks relative to the current session:

$$\tau = \max(Q_q(s), \text{median}(s) + \lambda \text{MAD}(s)), \quad (3)$$

where  $Q_q$  is a session-level quantile. Peaks below  $\tau$  are discarded, and among peaks separated by less than  $\Delta$ , only the strongest is retained by non-maximum suppression. A retained peak at  $t_p$  defines the action interval  $[t_p - T_{\text{pre}}, t_p + T_{\text{post}}]$ . The constants  $\alpha$ ,  $\beta$ ,  $C$ ,  $q$ ,  $\lambda$ ,  $\Delta$ ,  $T_{\text{pre}}$ , and  $T_{\text{post}}$  are fixed globally before evaluation rather than tuned for individual participants. Here,  $q$  and  $\lambda$  determine the adaptive peak threshold,  $\Delta$  is the minimum peak separation, and  $T_{\text{pre}}$  and  $T_{\text{post}}$  specify the crop around a peak. The output is therefore a sparse sequence of action windows instead of a classification decision at every sensor frame.

## 4.2 Feature Representation

For every window, PHANTOM computes statistical, percentile, directional-energy, and peak-aligned features from the raw axes and from acceleration and gyroscope magnitudes. The extraction stage produces 489 fields; removing label-irrelevant segmentation metadata leaves 482 model inputs. This representation captures both overall movement intensity and the direction or timing needed to distinguish lead from rear punches and left from right defense.

The selected offline analysis pipeline applies constant-value imputation, variance filtering, tree-importance selection at the median importance, robust scaling, and L2 normalization. Feature selection retains roughly half of the 482 inputs in each training fold. All transformation parameters are fit only on the factory-training participants and remain frozen for the held-out participant.

## 4.3 Reference-Based Classification

Classification uses distance-weighted 1-NN with Manhattan distance:

$$i^* = \arg \min_i \|\tilde{x} - \tilde{x}_i\|_1, \quad \hat{y} = y_{i^*}, \quad d_{\min} = \|\tilde{x} - \tilde{x}_{i^*}\|_1. \quad (4)$$

Here,  $\tilde{x}$  is the transformed query and  $\tilde{x}_i$  is a stored reference with label  $y_i$ . We report  $c = 1/(1 + d_{\min})$  as a relative confidence score rather than a calibrated probability. Predictions are sorted by peak time to form the session's ordered action sequence.

The exported iOS model intentionally omits tree-based feature selection to keep the mobile feature mapping fixed and easy to audit. It starts from the same 482 inputs, retains 469 dimensions after variance filtering, and stores 748 isolated-action references. The selected offline and deployed models are therefore reported separately in Section 6.

## 4.4 One-Shot Personalization and Rejection

During calibration, the user demonstrates a labeled action. PHANTOM segments the capture, extracts the same features, applies the frozen factory transform, and appends the vector to the reference set. No model parameter is optimized. This is the reason for choosing 1-NN: personalization becomes a local data operation rather than a training job.

Nearest-neighbor classification otherwise assigns every query to some class. PHANTOM therefore uses two complementary rejection mechanisms. A low-motion gate removes windows that do not contain enough activity to represent a deliberate action. For energetic non-boxing movements, users can record examples such as walking or raising an arm as *none* references. Let  $E(x)$  be the motion intensity of query  $x$ , and let  $d_B(x)$  and  $d_N(x)$  be its distances

**Table 3: Final evaluation dataset after quality control.**

| Subset           | Files | Evaluation units |
|------------------|-------|------------------|
| Isolated actions | 150   | 748 windows      |
| Combinations     | 150   | 150 sequences    |
| Total            | 300   | –                |

to the nearest boxing and *none* references. The rejection logic can be summarized as

$$\hat{y} = \begin{cases} \text{reject,} & E(x) < \tau_{\text{idle}}, \\ \text{none,} & d_B(x) > \rho d_N(x), \\ y_B, & \text{otherwise,} \end{cases} \quad (5)$$

where  $y_B$  is the nearest boxing label and  $\rho > 1$  prevents an ambiguous *none* reference from suppressing a valid boxing action. If the user has not recorded a *none* reference,  $d_N(x)$  is treated as infinite and the standard boxing decision is preserved. Here,  $\tau_{\text{idle}}$  is the minimum activity required for classification, and  $\rho$  is the margin required to prefer *none*. Both are fixed for all users. This combines a general activity check with user-specific negative examples without requiring a separate rejection model.

## 5 Evaluation Methodology

### 5.1 Dataset and Quality Control

The final cohort contains five independent participants. An additional barefoot recording was excluded because it came from a participant already represented in the cohort and was also collected barefoot, confounding subject identity with environment. Including it would have violated the leave-one-subject-out assumption. Every retained recording contains Watch, iPhone, and AirPods data.

The isolated set contains three files for each of ten actions and five participants, giving 150 files. Each file was intended to contain five repetitions, or 750 windows in total. Peak selection recovered 748 valid windows because two recordings contained only four usable repetitions. The continuous set contains ten predefined combinations, three recordings per participant, for another 150 files and 150 ordered sequences.

Some participants interpreted left and right slip instructions from different viewpoints. Before model evaluation, these labels were normalized according to the AirPods yaw convention while preserving the raw recordings. This correction is reported because left/right defense confusion is one of the main outcomes.

### 5.2 LOSO and File-Separated Calibration

Macro-F1 is the primary metric because it weights all ten classes equally. Cross-user performance is measured with leave-one-subject-out (LOSO) evaluation. In each of five folds, four participants form the factory training set and the remaining participant is excluded from every factory fit and used only for testing. Imputation, feature selection, scaling, and the initial reference set are therefore learned without samples from the test participant.

Because the cohort contains only five participants, we emphasize absolute effect sizes and protocol-controlled comparisons rather

**Table 4: Leakage-control rules used throughout evaluation.**

| Component        | Rule                                       |
|------------------|--------------------------------------------|
| Factory model    | Fit on four participants only              |
| Test participant | Excluded from all factory preprocessing    |
| Calibration      | One of three files for each action         |
| Test data        | The other two files for that action        |
| Preprocessing    | Frozen before user references are added    |
| Repetition       | Rotate calibration file and selected peaks |

than statistical significance claims. Pooled and subject-mean aggregations are identified explicitly wherever both are reported.

Personalization uses a second separation inside the held-out participant. The three files available for each action are split into one calibration file and two test files. At  $m = 1$ , one peak from the calibration file is appended for each action. Values  $m = 2, \dots, 5$  add more peaks from the same calibration file. The other two files remain a fixed test set and never contribute references. Each of the three files is used as the calibration source in turn, and peak choices are rotated before averaging.

### 5.3 Compared Configurations

The primary analysis uses 1-NN with fold-local tree-importance feature selection. Sensor ablation repeats LOSO with head, wrist, hand, all pairwise combinations, and all three devices. A separate deployment configuration removes feature selection and matches the exported iOS resource.

For comparison with punch-only work, we also reproduce the BoxerSense feature and model configuration as closely as the available data permit: seven statistics over six IMU axes, default scikit-learn SVM, random forest, and KNN models, and leave-one-person-out evaluation. Its Butterworth dual-threshold segmenter and upper-back sensor cannot be reproduced, so the comparison uses PHANTOM’s segmentation and is interpreted as a reference point rather than a direct ranking.

Combination transfer is evaluated in two settings. The first factory set contains only isolated windows. The second augments it with high-confidence windows extracted from the other participants’ combination files. Calibration still uses isolated demonstrations, allowing us to test whether an isolated example transfers to co-articulated motion.

### 5.4 Research Questions

The evaluation addresses three questions:

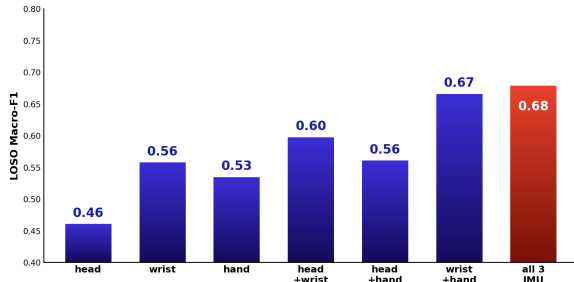
- **RQ1:** How much does multi-device sensing improve recognition over a single device?
- **RQ2:** How much does one demonstration per action improve recognition for an unseen participant?
- **RQ3:** How well does the isolated-action model transfer to continuous combinations?

## 6 Results

Table 5 summarizes the three research questions before the detailed analysis. Multi-device sensing and one-shot personalization

**Table 5: Summary of the main evaluation findings. Values are macro-F1.**

| Question          | Main finding                                                                             |
|-------------------|------------------------------------------------------------------------------------------|
| RQ1: devices      | Wrist 0.557 $\rightarrow$ wrist+hand 0.665 $\rightarrow$ all three 0.678                 |
| RQ2: calibration  | Pooled 0.637 $\rightarrow$ 0.916 with one demonstration per action                       |
| RQ3: combinations | Isolated-only 0.301 $\rightarrow$ combo-augmented 0.553; isolated calibration then 0.502 |

**Figure 4: LOSO macro-F1 by sensor combination for offline 1-NN with feature selection. Wrist and hand provide most of the multi-device gain, while all three devices form the best configuration.**

produce clear gains, while continuous combinations remain the principal failure case.

### 6.1 Why Multiple Devices?

Figure 4 answers RQ1. Wrist-only sensing reaches 0.557 macro-F1 and hand-only sensing reaches 0.534. Combining wrist and hand raises the score to 0.665, a gain of 0.108 over wrist alone. All three devices achieve the highest score, 0.678, although the 0.013 improvement over wrist plus hand is modest. Head-only sensing is weakest at 0.460.

The result supports multi-device sensing but also contextualizes the role of AirPods. Their current average contribution is small and varies by action. They are retained because slips and rolls are intrinsically head movements and because they provide a path toward more trajectory-specific defense features. The present accuracy gain, however, is driven primarily by the wrist-and-hand pair.

### 6.2 One Demonstration Personalizes the Model

Figure 5 gives the main before-and-after result. Under pooled file-separated evaluation, macro-F1 increases from 0.637 without user data to 0.916 after one demonstration per action. Lead hook and lead uppercut, which are strongly confused before calibration, reach F1 values near 0.85 and 0.91. Left/right slip confusion also decreases substantially. The average calibrated F1 over the four defense classes is approximately 0.86.

Figure 6 reports subject-mean rather than pooled aggregation. It rises from approximately 0.605 at  $m = 0$  to 0.912 at  $m = 1$  and approaches 0.954 at  $m = 5$ . The small numerical difference from

**Table 6: LOSO macro-F1 for isolated and combination windows. Calibration uses one isolated demonstration per action.**

| Factory references | Test windows | $m = 0$ | $m = 1$ |
|--------------------|--------------|---------|---------|
| Isolated           | Isolated     | 0.599   | 0.889   |
| Isolated           | Combination  | 0.301   | 0.337   |
| Isolated+combo     | Isolated     | 0.605   | 0.894   |
| Isolated+combo     | Combination  | 0.553   | 0.502   |

Figure 5 is due to the aggregation protocol. Both views support the same answer to RQ2: one example captures most of the useful person-specific information.

### 6.3 Comparison with Punch-Only Tracking

The most faithful BoxerSense reproduction uses one wrist IMU and three rear-hand punch classes. On PHANTOM’s data, its best pooled LOSO macro-F1 is approximately 0.61, whereas BoxerSense reports about 0.91 on its own dataset. These values should not be read as a head-to-head ranking. The original study used ten participants, controlled pauses between punches, and collection and segmentation details that cannot be reproduced completely with our recordings. PHANTOM uses five participants and includes more natural continuous movement.

The comparison instead clarifies PHANTOM’s contribution. Its contribution is not a universally larger punch-classification accuracy. It is the broader action scope: one model recognizes lead and rear punches together with four directional defense classes, and one-shot calibration makes this scope usable for a new participant.

### 6.4 Offline and Deployed Models

The selected offline model includes tree-importance feature selection. The exported iOS model omits that stage, retains 469 of 482 dimensions after variance filtering, and stores 748 references. Its LOSO macro-F1 is 0.581 without calibration and 0.887 at  $m = 1$ . The difference from 0.637 and 0.916 is the cost of a simpler, fixed deployment transform.

1-NN is not the most accurate candidate in every experiment. After one demonstration, an RBF SVC reached 0.931 subject-mean macro-F1 compared with 0.912 for 1-NN. We selected 1-NN because adaptation requires only appending a vector. In the Python analysis environment, an append took about 0.07 ms, whereas refitting SVC and ExtraTrees took approximately 26 ms and 843 ms. These timings compare adaptation mechanisms; they are not measurements of iPhone latency.

### 6.5 Continuous Combinations

Continuous combinations introduce co-articulation, truncated recovery phases, and short inter-action transitions. The anchor-based combination segmenter placed the expected number of ordered windows in 147 of 150 files (98%). This is a placement-success rate, not boundary accuracy, because frame-level temporal ground truth was not collected.

Table 6 answers RQ3. An isolated-only factory model transfers poorly to combination windows. Adding combination references

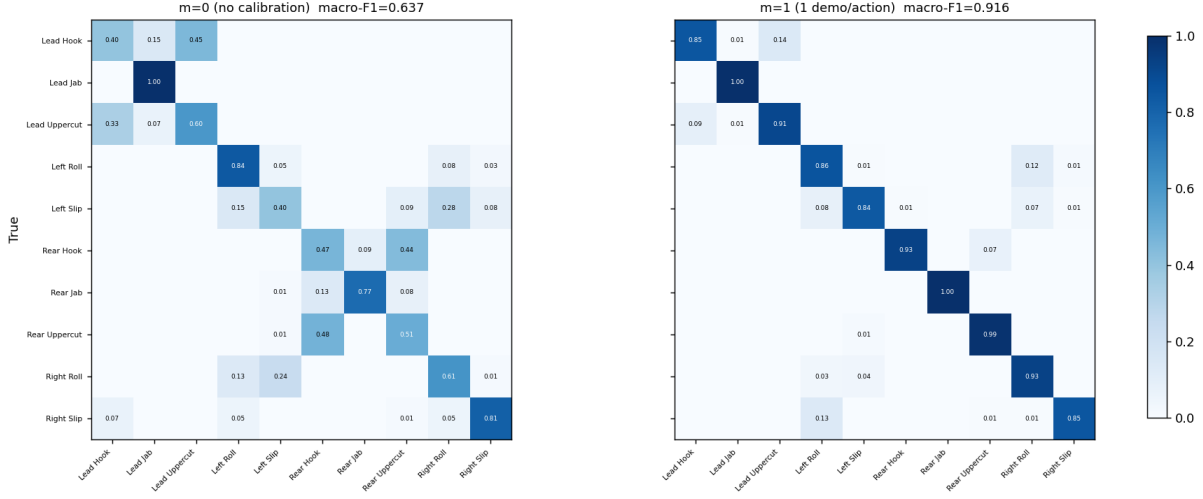


**Figure 5: Pooled confusion matrices before and after one file-separated user demonstration per action. Macro-F1 increases from 0.637 to 0.916, and left/right defense confusion is substantially reduced.**

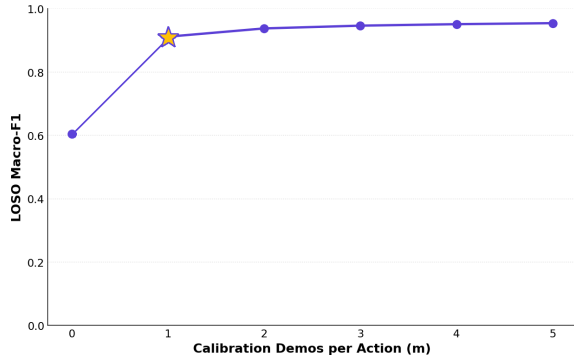


**Figure 6: Subject-mean macro-F1 versus calibration demonstrations per action. Most of the available improvement appears at  $m = 1$ , followed by diminishing returns.**

from training participants raises uncalibrated combination macro-F1 from 0.301 to 0.553 without reducing isolated performance. However, one isolated user example then lowers the combination score from 0.553 to 0.502. The nearest user reference can dominate even when its isolated context differs from the co-articulated query.

The category-level result is also informative. With calibration and all three sensors, defense macro-F1 decreases from about 0.85 on isolated windows to 0.74 when combination windows are included, while offense falls from about 0.95 to 0.60. Defensive movement remains comparatively robust, but fast punch transitions are the main source of failure.

## 7 Discussion and Limitations

### 7.1 Offense and Defense on Commodity Devices

PHANTOM demonstrates that a commodity-device stack can cover a broader range of boxing actions than punch-only tracking. The

important systems result is the combination of complementary sensing, local post-session inference, and lightweight personalization. Wrist plus hand supplies most of the current recognition accuracy, while the earbud stream makes head motion available for future defense-specific modeling.

The AirPods result should not be overstated. All three devices give the best average score, but the gain over wrist plus hand is only 0.013 and is not consistent across every action. More data and features that encode head rotation direction, angle, and trajectory are needed before head sensing can be established as the primary driver of defense accuracy.

### 7.2 Personalization as a System Requirement

The gap between uncalibrated and calibrated performance indicates that personalization is not merely an optional improvement. Cross-user movement variation is a central source of error. A one-demonstration workflow is practical because 1-NN treats calibration as an append operation and keeps all data on the phone.

This choice has a clear trade-off. A retrained SVC can be slightly more accurate in the present analysis, but it would require an on-device training implementation or a server. PHANTOM instead chooses immediate local adaptation and a transparent reference set. This trade-off is appropriate for a prototype whose main user-facing requirement is fast personalization, not maximum offline benchmark accuracy.

### 7.3 Three Main Limitations

First, segmentation is automated but heuristic. Peak-centered windows work well for isolated actions, and combination placement succeeds in 147 of 150 files, but there are no frame-level boundary annotations. The project can therefore report window-count placement, not exact temporal localization.

Second, combinations shift the signal distribution. An isolated punch has a clear preparation and recovery phase, while the same punch inside a combination may begin before the previous action

has finished. Isolated calibration references can consequently harm a combo-augmented model. Future calibration should include short combinations or explicitly condition references on motion context.

Third, the dataset is small and primarily same-day. Five participants are enough to reveal strong user variation but not to establish robustness across days, fatigue, stance, handedness, environments, or device-placement drift. Calibration improves performance for the current held-out user; it does not prove generalization to a broader population.

## 7.4 Additional System Limits and Future Work

The current workflow assumes access to all three Apple devices and performs analysis after the session. End-to-end iPhone latency, battery use, memory, and missing-sensor behavior have not been benchmarked systematically. The *none* references and motion gate reduce non-boxing predictions, but a larger negative-motion dataset is needed for reliable open-world rejection.

Future work should collect multi-day data from more independent participants, add frame-level action annotations, and balance stance and handedness. Technical extensions include orientation-robust features, head-trajectory representations, confidence calibration, and context-specific references for combinations. A complete deployment study should also measure session-analysis time and energy directly on the phone.

## 8 Conclusion

PHANTOM tracks the offensive and defensive halves of shadow-boxing with an Apple Watch, an iPhone, and AirPods Pro. The system synchronizes their IMU streams, extracts peak-centered action windows, and performs reference-based recognition and personalization locally on the iPhone.

Three findings summarize the project. First, multi-device sensing improves LOSO macro-F1 from 0.557 with the wrist alone to 0.665 with wrist and hand and 0.678 with all three devices. Second, one file-separated demonstration per action raises the selected model from 0.637 to 0.916 pooled macro-F1; the deployment-oriented model similarly improves from 0.581 to 0.887. Third, the system recognizes four defensive classes with approximately 0.86 calibrated F1, extending the action coverage beyond punch-only tracking.

The remaining challenge is continuous motion. Combination-aware references improve uncalibrated combination recognition, but isolated personalization does not transfer cleanly to co-articulated actions. A larger multi-day dataset, temporally annotated combinations, and context-aware calibration are therefore the next requirements. Within these limits, PHANTOM establishes the feasibility of personalized offense-and-defense tracking using devices the user already owns.

## References

- [1] Yoshinori Hanada, Tahera Hossain, Anna Yokokubo, and Guillaume Lopez. 2022. BoxerSense: Punch Detection and Classification Using IMUs. In *Sensor- and Video-Based Activity and Behavior Computing*. Smart Innovation, Systems and Technologies, Vol. 291. Springer Nature Singapore, 95–114. doi:10.1007/978-981-19-0361-8\_6
- [2] Pixi Kang, Julian Moosmann, Mengxi Liu, Bo Zhou, Michele Magno, Paul Lukowicz, and Sizhen Bian. 2025. Bridging Generalization and Personalization in Human Activity Recognition via On-Device Few-Shot Learning. arXiv:2508.15413 [cs.LG] doi:10.48550/arXiv.2508.15413
- [3] Vimal Mollyn, Riku Arakawa, Mayank Goel, Chris Harrison, and Karan Ahuja. 2023. IMUPoser: Full-Body Pose Estimation Using IMUs in Phones, Watches, and Earbuds. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. ACM. doi:10.1145/3544548.3581392
- [4] Subhas Chandra Mukhopadhyay. 2015. Wearable Sensors for Human Activity Monitoring: A Review. *IEEE Sensors Journal* 15, 3 (2015), 1321–1330. doi:10.1109/JSEN.2014.2370945
- [5] Thomas Sepanosian and Ozlem Durmaz Incel. 2024. Boxing Gesture Recognition in Real-Time Using Earable IMUs. In *Companion of the 2024 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. ACM. doi:10.1145/3675094.3680524