Unique Identification of 50,000+ Virtual Reality Users from Head & Hand Motion Data

Vivek Nair *UC Berkeley*
Wenbo Guo *UC Berkeley*
Justus Mattern *RWTH Aachen*
Rui Wang *UC Berkeley*
James F. O'Brien *UC Berkeley*
Louis Rosenberg *Unanimous AI*
Dawn Song *UC Berkeley*

Abstract

With the recent explosive growth of interest and investment in virtual reality (VR) and the "metaverse," public attention has rightly shifted toward the unique security and privacy threats that these platforms may pose. While it has long been known that people reveal information about themselves via their motion, the extent to which this makes an individual globally identifiable within virtual reality has not yet been widely understood. In this study, we show that a large number of real VR users (N=55,541) can be uniquely and reliably identified across multiple sessions using just their head and hand motion relative to virtual objects. After training a classification model on 5 minutes of data per person, a user can be uniquely identified amongst the entire pool of 55,541 with 94.33% accuracy from 100 seconds of motion, and with 73.20% accuracy from just 10 seconds of motion. This work is the first to truly demonstrate the extent to which biomechanics may serve as a unique identifier in VR, on par with widely used strong biometrics like facial or fingerprint recognition.

1 Introduction

The emergence of affordable standalone virtual reality (VR) devices, such as the Meta Quest 2, has allowed VR to reach mass-market adoption in recent years, with nearly 10 million VR headsets sold in 2022 alone [21]. Coinciding with this dramatic increase in VR usage is a wave of new academic research revealing a range of unique security and privacy threats associated with these devices [11].

Gaming has thus far been the predominant driver of VR adoption, with 91 of the 100 most popular VR applications being games as of early 2023 [29]. While gaming is typically perceived as a fairly innocuous class of applications from a privacy standpoint, the opposite may actually be true in VR. In this paper, we examine the extent to which spatial telemetry captured during VR gaming sessions can be used to uniquely identify an otherwise anonymous player.

We obtained a novel dataset containing over 2.5 million recordings of users playing "Beat Saber," a VR rhythm game that today is by far the most popular VR application [31]. Using a unique combination of context-aware featurization and hierarchical machine learning, players can be identified out of a pool of over 50,000 candidates with 94.33% accuracy from 100 seconds of head and hand motion data, or with 73.20% accuracy from just 10 seconds of movement.

It has long been understood that individuals exhibit distinct biomechanical motion patterns that can be used to identify them or infer their personal attributes [5, 13, 16, 17, 23, 25]. However, the extent to which the subset of this information that is observable in VR can be used to uniquely identify users is less well understood. Although prior research has been conducted on the personal identifiability of VR tracking data [19, 22, 24, 24, 30], existing works have utilized data from small lab studies with 16 to 511 participants. By contrast, our dataset is not only more than 100 times larger than the largest prior result, but is also far more representative of a realistic use case, comprising 55,541 real VR users across over 40 countries and using over 20 different types of VR devices.

Despite the difficulty of identification growing in proportion to the number of users, we achieve comparable identification accuracy to prior works. We show that while identifying users in smaller sets (≤ 511) can be accomplished just by learning static attributes like height, actual behavioral differences in movement patterns must be utilized to identify users within our substantially larger dataset. As such, our work is the first to truly demonstrate the extent to which motion can be an identifying feature in VR.

Contributions:

1. We have identified the largest and most representative dataset to date of virtual reality telemetry recordings (§3).
2. Our featurization technique uses VR application context information to enhance VR user identification (§5).
3. Our hierarchical classification approach allows us to build a scalable identification model with 50,000+ classes (§6).
4. We achieve 94.33% identification accuracy across 55,541 users (§7) and provide detailed explainability results (§8).

2 Background

A virtual reality device uses an array of sensors to generate a stream of information about its user, which is consumed by an onboard or external computer to render stimuli for the user, thereby creating an immersive experience. In the case of multi-player (or "metaverse") applications, the generated data is also shared with a variety of external systems, which could then use it to infer private user information.

The 2023 VR privacy SoK by Garrido et al. [11] presents a standard model of VR information flow and threat actors, which we will briefly recount below and use to position our study within the broader landscape of VR privacy research.

2.1 VR Information Flow

A typical consumer-grade virtual reality system comprises at least a head-mounted display (HMD) and two hand-held controllers. The system uses either external or onboard sensors to measure the position and orientation of these devices in 3D space, providing six degrees of freedom (6DoF) per tracked object. These six measurements per object are taken for the user's head and hands, constituting 18 tracked dimensions in total. The data are captured at a usual rate of between 60 and 144 times per second, resulting in a "telemetry stream."

Many VR devices contain additional sensors, such as microphones, cameras, and eye or full-body tracking devices. In this paper, we focus entirely on the basic motion telemetry data noted above, so as to investigate the question of how users can be identified by their motion alone.

The telemetry stream generated by a VR device is first used by a client-side application running on an onboard or connected computer to render a separate series of visual stimuli (or "frames") for each eye, along with auditory and haptic stimuli, creating an immersive 3D virtual world.

In the case of a multiplayer or metaverse experience, the application also forwards the telemetry stream to an external game server. The server, in turn, forwards this data to other connected users, so that a virtual representation (or "avatar") of each user can be rendered on the devices of other users.

2.2 VR Threat Model

Per the Garrido VR threat model, each entity in the above information flow that can view the VR device telemetry of a target user is considered a potential adversary. Specifically, the attackers generally considered in VR privacy research are VR hardware (I), VR applications (II), external servers (III), and external users (IV). Each of these adversaries receives a view of the telemetry stream, which it could use to make adversarial inferences of private VR user information instead of (or in addition to) its intended purpose of facilitating application functionality. However, because the data can be reduced and compressed at each stage of the information flow, adversaries in higher tiers are considered "weaker" in this model.

Fig. 1 illustrates the general information flow and threat actors discussed thus far. In this paper, we are particularly interested in the game server (III) and other users (IV) as potential adversaries. These parties receive data processed by and filtered through the prior entities, meaning that attacks available to them can often be performed by other entities with even greater precision. They are also amongst the hardest attacks to detect due to their remote nature. This study exclusively analyzes data sent from a popular VR game to a remote server or other users, meaning that our attacks represent the hardest and most pernicious realistic threats in VR.

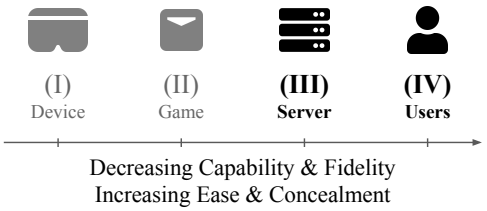

Figure 1: Selected VR threats relevant to this work.

2.3 VR Threat Scenarios

Consider a public figure who frequently uses a VR system with their corporate credentials to do professional work. In the evening, they log on with a different account for multiplayer VR gaming (where they might not behave in the most professional way), and later in the evening, they use a third account for adult VR experiences. Most individuals in this situation would reasonably prefer that the adversaries outlined above not be able to tie these accounts together. However, if a user can be uniquely identified by their VR motion patterns, any observer (or potentially even a group of colluding adversaries) could quickly link all of these accounts to them simply by observing their movement in each context.

On the web, "browser fingerprinting," which uses subtle differences between browsers to link people across web services, is highly analogous and is regarded as a significant privacy concern [9]. However, while one can replace their browser, they cannot easily change the distinct physiology and muscle memory that dictates their apparent movements, making motion identification a particularly challenging privacy threat.

As discussed in the following section, the data used in this paper originates from a single, popular VR game. Thus, we cannot yet demonstrate the ability to track users across applications, which we hope to see attempted in future work.

2.4 Beat Saber

"Beat Saber" [10] is an award-winning virtual reality rhythm game where players slice blocks representing musical beats with a pair of sabers they hold in each hand. With over 6.2 million copies sold, Beat Saber is the most popular and highest-grossing VR application of all time [31].

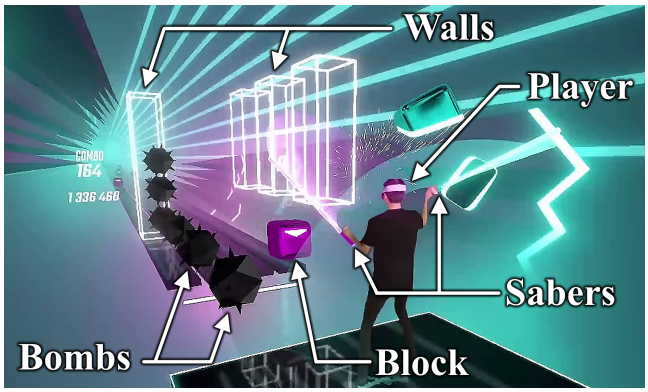

Figure 2: Interactive objects in Beat Saber game.

The Beat Saber game is segmented into "maps" which consist of a music track and a series of challenges presented to the user in time with the music. These challenges include "blocks," which the player must hit with both the correct saber (as indicated by the color) and correct angle (as indicated by the arrow), "bombs," which the player must avoid hitting with their sabers, and "walls," which the player must avoid with their head (see Fig. 2). If the player completes these actions successfully, they are awarded points according to their level of accuracy. On the other hand, if the player performs poorly, the map may terminate early with a "level failed" message.

While users can play and compete for high scores on hundreds of maps included in the base game or as purchasable extensions, over 100,000 user-created maps for popular songs are available by installing open-source game modifications.

3 BeatLeader Dataset

"BeatLeader" [26] is an open-source Beat Saber extension that maintains third-party leaderboards for over 100,000 custom Beat Saber maps. Beat Saber players may choose to install the BeatLeader extension in order to compete with other players to achieve a higher "rank" on the BeatLeader leaderboards. After playing a Beat Saber map with the BeatLeader extension enabled, scores are automatically uploaded to a globally-visible leaderboard. Since May 2022, over 50,000 users have posted over 2.5 million scores to the BeatLeader platform.

When uploading a score to BeatLeader, a recording (or "replay") of the user's performance is automatically captured by the BeatLeader extension and attached to their submission. The replay is made available to other BeatLeader users, who can use it to verify the authenticity of the submitted score.

In partnership with the administrators of BeatLeader, we obtained a 3.96 TB dataset consisting of 2,669,886 replays from 55,541 users across 713,013 separate play sessions. The dataset has between 1 and 4,509 replays per user, with a median of 14. The replays range in length from 5 seconds to over an hour,[1] with a median length of 2 minutes and 56 seconds.

3.1 Replay Format

Replays in the dataset are encoded in the "Beat Saber Open Replay" (BSOR) [27] format. BSOR files comprise four parts:

1. **Metadata.** Device information and the values of all user-configurable game settings are included in the replay.
2. **Telemetry.** The position and orientation of the player's head and hands is recorded every time a frame is rendered by the game, usually 60 to 144 times per second.
3. **Context.** Replays encode the type, location, and timing of in-game stimuli, such as "blocks," "walls," and "bombs," which the player is responding to throughout the replay.
4. **Performance.** BSOR files also score the validity and accuracy of the user's response to each in-game stimulus.

3.2 User Attributes

As noted above, Beat Saber replay files include metadata that reveals a number of user-specific data attributes. These attributes primarily consist of device information, such as the VR platform, runtime environment, software version numbers, and the make and model of the VR headset and controllers. They also contain the settings chosen by the user, including self-selected height and handedness. Finally, the number of replays present for each user indicates their level of "experience," while their performance relative to other users playing the same map provides a measure of "skill." The distribution of the users across these attributes is included in §B.

Because the goal of this paper is to uniquely identify VR users across sessions based on their motion alone, we do not incorporate any of the provided metadata into our identification models. However, we do use this information in §7.3 to identify which attributes correlate with higher or lower identification accuracy and contextualize the results accordingly.

3.3 Dataset Access

Researchers can access the data and source code necessary to replicate the results of this paper via the BOXRR-23 dataset or our open-source repository on Zenodo. The repository includes detailed documentation and instructions on how to use the data to replicate each result presented in this paper.

https://rdi.berkeley.edu/metaverse/boxrr-23
https://zenodo.org/record/7935034

[1] Some maps are longer than a single song; e.g., an entire film soundtrack.

3.4 Ethical Considerations

Because our work involves data derived from human subjects, significant attention was given to ethics throughout the study. We note that no original data collection was performed by the authors; we used an existing dataset from an external source. All data utilized in this study was already broadly, publicly available, to any person in the world with an internet connection, without the need for permissions, credentials, authentication, or any special tools or applications. We made no attempt in this work to infer user data that could be particularly sensitive, such as health information, and voluntarily followed the strictest PII data handling standards and guidelines offered by our institution throughout the research process.

Regarding consent, BeatLeader users must go out of their way to voluntarily modify their Beat Saber installation to add the BeatLeader extension and share their Beat Saber replay data. They are fully aware of the nature of the data being shared, as uploading and sharing Beat Saber replay data is the explicit purpose of the extension. They also consent to their replay data being used for a variety of purposes, including for data analysis, in the BeatLeader Privacy Policy, which states that "Replays may contain personally identifiable information... Your data, including associated personally identifiable information, will be broadly publicly available to anyone with an internet connection via the BeatLeader website."[2]

We submitted a detailed research proposal to our institution's IRB as protocol #16120, in which we described precisely the BeatLeader telemetry data and its potentially sensitive nature, as well as our PII handling procedures, and our research goals. Since no original data was collected from human subjects, and BeatLeader data is already public, the protocol was deemed IRB-exempt under 45 C.F.R. § 46.104(d)(4)(i) and was and issued a Notice of Approval by our IRB.

Despite the public nature of the data and the IRB approval, we chose to obtain written permission from BeatLeader before proceeding out of an abundance of caution and respect for the community from which this data originates. After obtaining our initial results, we worked with BeatLeader to notify users of our findings via their official social media channels.

Overall, we believe this research constitutes a net benefit to society by highlighting the magnitude of the VR privacy threat and motivating future work on defensive countermeasures. It further benefits the Beat Saber users whose data was utilized by highlighting the possible implications of the telemetry data which they had already made public, and also enabling the potential future development of anti-cheating tools.

In the interest of user privacy, we have not publicly released the entire raw dataset used to produce this work, despite its already public nature. However, with the permission of BeatLeader and our IRB, we have released enough de-identified and normalized data to replicate our results (see §3.3).

[2] See https://www.beatleader.xyz/privacy

4 Related Work

4.1 Motion

Since as early as the 1970s, researchers have demonstrated that people reveal identifying information about themselves via their motion. In a 1977 study of 6 participants, Cutting and Kozlowski demonstrated that individuals can identify their friends with 38% accuracy by viewing the motion of 8 tracked objects affixed to the body [5], and that the gender of the 6 participants could be identified by a stranger with 63% accuracy using the same 8 tracked objects [17].

More recently, Pollick et al. (2005) [25] have used statistical techniques to achieve 79% accurate identification of gender from motion. In a study of 8 participants, Jain et al. (2016) [13] found that the motion of children can be differentiated from that of adults with 66% accuracy.

In two further related works, O'Brien et al. (2000) [23] and Kirk et al. (2005) [16] demonstrated the ability to use motion data to infer a person's skeletal structure. O'Brien et al. used 16 sensors recorded with 6 degrees of freedom, while Kirk et al. used 30 to 40 optical markers captured with 3 degrees of freedom. Although not the explicit purpose of these works, the skeletal models could be used for user identification.

Virtual reality is somewhat distinct from the situations described above in that only 3 tracked locations are typically provided rather than the 8 to 40 used in the mentioned studies. Until the relatively recent proliferation of VR technology, the applicability of these results to VR has been uncertain.

4.2 Virtual Reality

The 2023 Garrido et al. VR privacy SoK [11] provides a recent survey of the VR privacy research landscape. Our work would fall under the "geospatial telemetry" attribute class in the SoK's taxonomy. Here, we summarize the works listed in the same category which are most relevant to our own.

First, Pfeuffer et al. (2019) [24] performed a laboratory study of 22 users, who were instructed to perform a variety of tasks in VR (pointing, grabbing, walking, typing) across two sessions. Using a random forest model, they were able to identify a user within the set of 22 with up to 40% accuracy.

Next, Miller et al. (2020) [20] conducted a lab study of 511 users, whose telemetry was captured while they watched a series of 360-degree videos in VR. With a random forest model, their system correctly identifies users within the pool of 511 with 95% accuracy from 5 minutes of telemetry data.

Liebers et al. (2021) [19] conducted a similar lab study of 16 users, who were asked to play archery and bowling games in VR. They were able to identify users within the set of 16 using an LSTM model with 90% accuracy.

Finally, Tricomi et al. (2022) [30] demonstrated the profiling of AR and VR users with laboratory studies of 34 and 35 users, respectively. They uniquely identify 30 users in VR with 95% accuracy using a logistic regression model.

Overall, Miller et al. is the largest known study of VR user identifiability, with 511 users across 5,110 sessions. Our study, with 55,541 users and 713,013 sessions, is thus at least two orders of magnitude larger than the largest existing result.

Furthermore, while all of the above works involve data collected from a highly-controlled laboratory setting with 1 to 3 device types, our dataset originates from real VR users in 40+ countries, and includes 20+ types of VR devices in a wide variety of heterogeneous physical environments.

Despite the significantly harder task of identifying users amongst tens of thousands of possibilities and in uncontrolled environments, we achieve comparable or better accuracy to the prior works. We believe this is the first study to truly demonstrate the staggering scale of the VR privacy threat.

4.3 Machine Learning

Classical ML. As summarized above, existing VR privacy studies model user identification as a classification problem and leverage machine learning to classify users based on feature vectors of extracted data. Given that the existing studies process the telemetry data into a relatively small tabular dataset, these works usually leverage classical ML techniques (such as random forest [2] and gradient boosting [3]).

Underlying these models are decision trees, which construct a tree-based rule structure for a learning problem. A random forest ensembles multiple decision trees to improve the model's capacity, and thus is capable of handling more sophisticated learning problems. Gradient boosting takes this a step further by iteratively optimizing the set of trees rather than simply aggregating them. During the training process, gradient boosting actively updates the trees and their weights based on the current prediction results, allowing it to generally achieve a better performance than random forests alone [4]. We observe similar results in our study, with gradient boosting models providing by far the best performance.

Deep Learning. Interestingly, only one of the existing studies (Liebers et al. [19]) has used deep learning-related techniques for user identification, and its results are amongst the least accurate at 90% accuracy with 16 users. This is counterintuitive, as deep learning has become a mainstream technique in the machine learning community. Research in different application domains has demonstrated that deep learning algorithms (e.g., Multi-layer Perceptrons), outperform traditional (e.g., tree-based) ML models in dealing with tabular data [12].

However, this may not be the case in VR user identification. This application has a very large number of users, which means that the classifier has to distinguish a large number of classes. It is challenging for deep learning models to train and converge under these conditions because they require a multi-class classifier to contain a large number of neurons in the output layer. In fact, most existing benchmark datasets where deep learning demonstrates a superior performance have a small number of classes. For example, the widely used image classification datasets MNIST [7] and CIFAR-10 [18] have ten classes, and some widely used text classification datasets only have 20 classes (Newsgroups [1]). The dataset with the most classes is ImageNet [6], which has 1,000 classes.

We found that deep learning empirically fails to perform well in our study, which requires more than 50,000 classes. Still, it is likely that larger and more sophisticated deep learning models could achieve strong performance in the future.

5 Featurization

In this section, we describe our method for converting the time-series replay telemetry data into a flat feature vector which can be consumed by a basic non-sequential model. The featurization techniques described in this section are used in the identification models discussed later in this paper.

We determined the best-performing model architecture and featurization method through a complex multi-parameter optimization in which we evaluated a variety of different featurization approaches together with a variety of classification model architectures and hyperparameters. In this process, more than 1,000 separate models were trained and tested using a validation set. However, we have chosen to use the single best-performing model architecture throughout this section to simplify the explanation of our feature selection.

Specifically, in this section, we use a 500-user identification model to validate our featurization choices and compare the resulting classification accuracy to the Miller et al. approach. For each proposed featurization approach, we randomly chose 500 users from our dataset and generated 150 training and 15 testing samples per user, using the train/test split discussed above. The features were then standardized using Z-score normalization before being used to train a 500-class LightGBM classification model. The identification accuracy on a per-sample and per-user basis is used to evaluate each approach.

We define a "session" as a continuously-recorded sequence of replays from a single user where no more than 10 minutes have elapsed between each replay. Our dataset contains an average of 13 such sessions per user. For each user, we reserve 70% of the sessions for training, 10% for validation, and 20% for testing, with a minimum of 1 session per set. As such, our models always perform true cross-session user identification rather than merely learning session-specific features, such as the exact position of a user within their room.

We begin with the best-performing existing method of featurizing VR telemetry data, which is that of Miller et al. [20], achieving 95% accuracy on 511 users. We describe this method in §5.2, and improve upon it in subsequent parts.

5.1 Guiding Principles

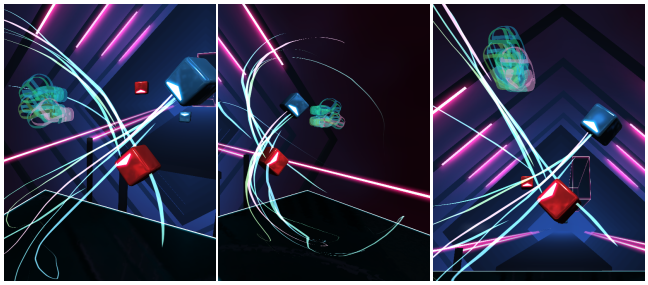

Figure 3: Five Beat Saber users hitting the same block pattern.

Fig. 3 shows, from several perspectives, the path taken by five Beat Saber users when slicing the same pair of blocks. As is clearly visible by the depictions, different users exhibit distinct motion responses even when presented with identical stimuli. These differences may be the result of physiology, learned motion patterns ("muscle memory"), random variance, or a combination thereof. The goal of the identification models presented in this paper is to learn a set of motion characteristics that uniquely represent a user. Accordingly, the featurization techniques of this section aim to reduce the dimensionality of the telemetry stream to the extent possible while retaining the ability to differentiate between users.

5.2 Motion Features

Motion data (telemetry) is the primary source of data for user identification and inference in VR. Fig. 4 shows a one-second segment of the head and hand motion of a Beat Saber user.

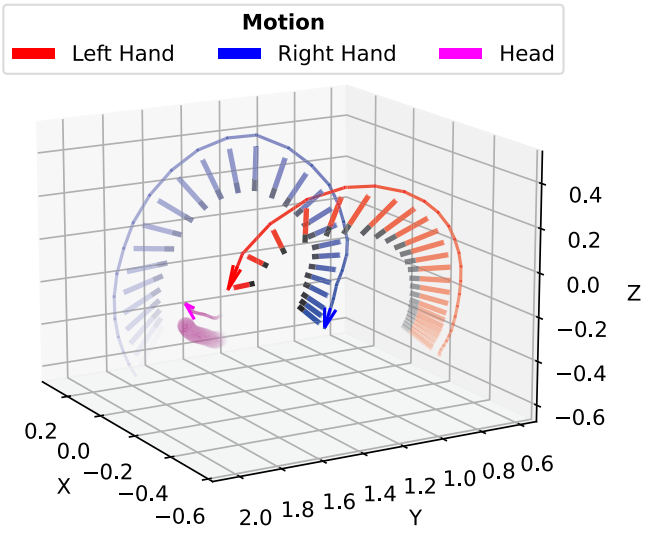

Figure 4: Head and hand motion from one second of telemetry.

As is visible in Fig. 4, each frame of telemetry data encodes 3D position and orientation coordinates across each of the three tracked objects. The Miller et al. method of motion data encoding suggests summarizing each of these 18 data streams using five summary statistics, namely the minimum, maximum, mean, median, and standard deviation, resulting in a 90-dimensional output vector. Using this approach with the Beat Saber data yields a 69.3% accurate per-sample identification and 93.4% accurate per-user identification using the evaluation method described above. This is comparable to the 95% accuracy reported by Miller et al. with their dataset.

In practice, we found that better performance is achieved by providing orientation measurements as four quaternion elements instead of three Euler angles. This modification alone resulted in an improved per-sample identification accuracy of 80.1% and per-user identification accuracy of 96.6%. Thus, our best-performing motion featurization can be represented as a 105-dimensional vector constructed as follows:

$$\{pos_x, pos_y, pos_z, rot_i, rot_j, rot_k, rot_1\}$$
$$\times$$
$$\{min, max, mean, med, stdev\}$$
$$\times$$
$$\{head, left\_hand, right\_hand\}$$

5.3 Context Features

While motion alone may be sufficient to identify 500 users, additional information is needed when dealing with significantly larger datasets. In particular, models can benefit from knowing the activity-specific context in which a motion segment is captured such that different users can be compared directly when performing similar actions.

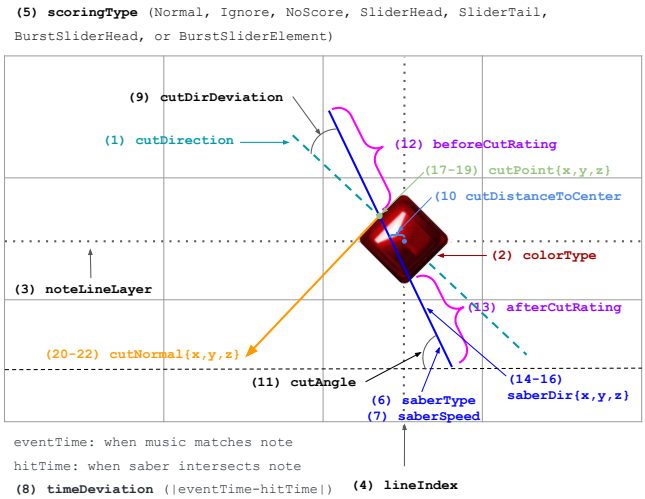

Figure 5: The 22 contextual features of a Beat Saber block.

In the case of Beat Saber, the activity chosen was the act of slicing an approaching block with a saber held in either hand. Specifically, we found 22 features that most accurately characterize movement relative to a single block, as shown in Fig. 5. These features include, for example, the position,

orientation, type, and color of the block, the angle, speed, location, and accuracy of the cut, and the relative error of the cut in both space and time.

Although these 22 features provide a comprehensive yet succinct parameterization of a user's response to an individual block, they are insufficient to identify users without accompanying motion features. Using these features alone with the previously-established evaluation method yields just 14.8% accuracy per sample and 43.8% accuracy per user. While this is still highly statistically significant relative to the 0.2% accuracy one would achieve by attempting to identify one of the 500 users at random, it under-performs even the basic Miller et al. approach. Still, it demonstrates the potential to aid identification when combined with motion features.

5.4 Hybrid Featurization

Finally, we describe the inclusion of both motion and context features within a single feature vector, thus allowing models to interpret motion data specifically in relation to other users performing the same or similar actions. By combining the 22 context features of §5.3 with the 105 motion features of §5.2 corresponding to one second of motion centered on the moment of contact, a 127-dimensional hybrid feature vector can be produced. Using this feature set with our established evaluation approach yields 83.8% accurate per-note user identification, with 98.2% accurate identification per user.

While this hybrid feature set now outperforms either the motion or the contextual features alone, some useful information is still excluded. In particular, it is useful to explicitly separate the motion features from before and after a target event. For example, different information can be learned from a user's "in swing" and "out swing" relative to a block.

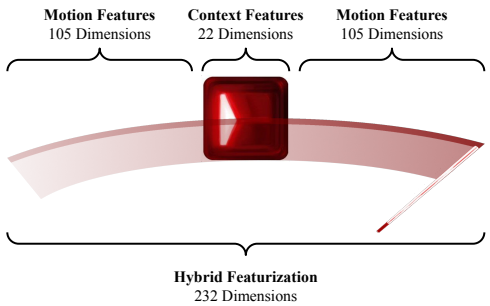

Figure 6: Hybrid featurization of a Beat Saber block.

Fig. 6 shows a full hybrid featurization of a Beat Saber block, including 22 contextual features for the block and 105 motion features corresponding to the one-second intervals before and after the block, totalling 232 dimensions. When evaluating this featurization with the same machine learning approach as before, 93.2% accurate identification is achieved per sample, with perfect (100.0% accurate) per-user identification of 500 users. The results of all approaches discussed in this section are summarized in Table 1.

Featurization Approach	Features (#)	Accuracy (Per Sample)	Accuracy (Per User)
Motion (Euler Angles)	90	69.3%	93.4%
Motion (Quaternion)	105	80.1%	96.6%
Contextual	22	14.8%	43.8%
Light Hybrid	127	83.8%	98.2%
Full Hybrid	232	93.2%	100.0%

Table 1: Accuracy of identifying 500 users using LightGBM with each of the discussed featurization methods.

In summary, the combination of rich contextual information about an event with separate features summarizing motion before and after said event is effective at achieving accurate identification for datasets significantly larger than 500 users. This is in part because the motion segments can be understood in the context of the corresponding stimuli, and in part because it begins to simulate a small sequential model; that is, it allows the model to ascertain which motion features are consistent and which change across two consecutive time slices. As such, we use this 232-dimension hybrid featurization method in all subsequent models for the remainder of this paper.

6 Model Architecture

Having established the above featurization technique, we next describe our selected machine learning model architecture for identifying users. This remains a non-trivial problem in practice, as it requires a 50,000-class classification model, a use case that many existing machine learning algorithms are not designed to handle (see §4.3). Therefore, after selecting a performant algorithm and preprocessing method, we describe a hierarchical approach for constructing the overall classification model out of several smaller classifiers.

6.1 Algorithm Selection

Using the best-performing feature set from §5, we tried to construct an identification model using 6 popular classical machine learning classification algorithms with the same sample of 500 users. For each algorithm, we began by using the default hyperparameters and then ran up to 25 rounds of tuning to obtain the below results, which show the best per-sample identification performance achieved by each algorithm.

- LightGBM: **93.2%**
- XGBoost: 80.0%
- Logistic Regression: 72.2%
- Support Vector Machines: 67.13%
- Extreme Random Trees: 35.5%
- Random Forest: 32.1%
- Naive Bayes: **1.2%**

As discussed in §4.3, gradient boosting models are known to outperform other tree-based classification algorithms on tabular datasets, which matches our observations above. In particular, LightGBM [15], an industry-leading gradient boosting framework, exhibited by far the best performance.

We also tried multiple sequential and non-sequential deep learning approaches with limited success. As summarized below, the deep learning attempts far underperformed the classification accuracy of the best classical ML algorithm.

- GRU: **84.0%**
- LSTM: 83.0%
- MLP: **72.0%**

Overall, we conclude that simple deep learning algorithms empirically failed to perform as well as LightGBM for the large multi-class classification task at hand. Moving forward, we use LightGBM for our identification models in view of the performance results and the fundamental factors favoring gradient boosting for this type of application.

6.2 Preprocessing Method

Using the hybrid featurization and LightGBM model with optimized hyperparameters (see §A), we evaluated five potential preprocessing methods, the results of which are shown below.

- StandardScaler: **93.2%**
- MinMaxScaler: 89.8%
- MaxAbsScaler: 86.4%
- SparseNormalizer: 83.5%
- TruncatedSVD: **66.5%**

The preprocessing approach with best results is standard scaling (Z-score normalization), whereby each feature is transformed by removing the mean and scaling to unit variance.

6.3 Hierarchical Approach

For smaller datasets, the above methods would be adequate. Indeed, if up to 5,500 classes are present, a single LightGBM classification model, deployed with our described featurization and preprocessing method, demonstrates strong performance in identifying users. Unfortunately, training a single LightGBM model with 50,000 classes would be infeasible with our dataset. We found that the training time and memory consumption of training a LightGBM classifier scales quadratically with the number of classes, as shown in Fig. 7.

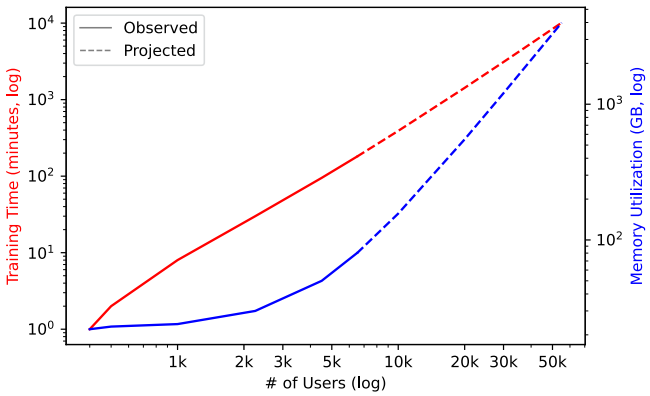

Figure 7: Observed and projected time and memory required to train an increasingly large LightGBM classifier.

According to a polynomial projection of our attempts to train classifiers with as many as 5,000 users, training a single classifier with all 55,000 users would take over 7 days and consume nearly 4 TB of RAM. While still within the realm of possibility when using server-grade hardware, the prospect of even larger datasets over the horizon motivates us to find a more efficient and scalable architecture.

We ultimately chose to construct a multi-layer hierarchical classifier. Our overall identification model is composed of three layers of smaller classifiers, each of which are only trained on a small set of available classes.

Figure 8: Hierarchical structure with 5 models per layer.

Fig. 8 illustrates the principle method by which the first two classification layers are constructed. In the first layer, N classifiers are each trained on 1/N of the available classes. In practice, we train 10 classification models with about 5,000 users each. This single layer already provides better performance than one may expect. Although each of the models will output a classification when identifying a user, regardless of whether that user is actually contained within their training set, the classification probability is usually highest in the model actually containing the target user.

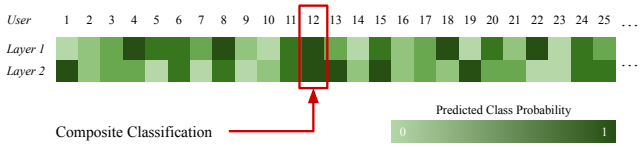

Figure 9: Class probabilities output by hierarchical classifier.

Further accuracy can be obtained by adding a second layer, also containing N classifiers each trained on 1/N of the available classes, with an even class redistribution from the first

layer. Now, when querying each layer to identify a user, the layers are likely to agree on the correct user while disagreeing about false classifications (see Fig. 9). The overall classification can now be obtained by taking the highest logarithmic sum of the class probabilities output by both layers.

Adding more layers at this stage via random redistribution provides diminishing returns. Instead, a separate clustering set (independent of the train, validate, and test sets) can now be used to cluster users based on their class confusion using the existing two layers. The method for doing so using connected components in a graph is illustrated in Fig. 10.

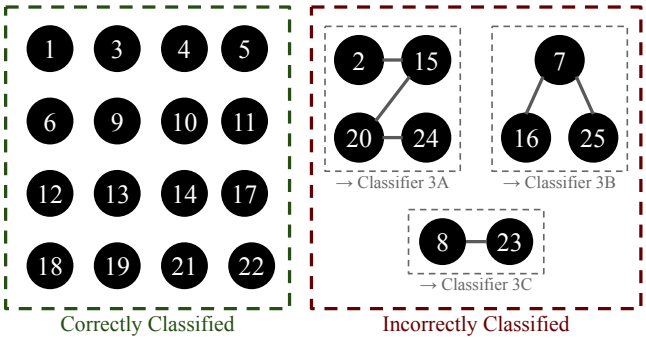

Figure 10: Graph-based method of selecting layer 3 groups.

As illustrated in Fig. 10, an undirected graph is constructed with a node for each user. Every time a user is incorrectly classified using the clustering set, an edge is added between the user and up to five apparently similar users. The connected components of this graph now represent sets of users who are likely to be misidentified as each other. In a third layer, one additional model can be trained for each component C in the graph (where $|C| > 1$), containing the users of C.

When ultimately identifying a user, the logarithmic sum of the first two layers is used to obtain an initial identity. If the resulting user is present in one of the connected components, the corresponding model in the third layer is used to produce the final classification. Otherwise, the initial classification is directly returned as the predicted identity. Given limited computational resources, this approach increases the odds that similar classes are directly compared in at least one model.

6.4 Scalability

While motivated initially by the infeasibility of training a single multiclass classification model of insufficient size, the proposed hierarchical architecture also presents a number of important scalability and practicality improvements over a monolithic approach. Each model in a layer can be trained in parallel, allowing for a 10-20x reduction in training time when using a cluster. Testing and inference can similarly be parallelized by evaluating each model separately.

Finally, the cost of adding a new user is significantly reduced by the hierarchical approach. When a new user is added, only one model on each layer must be retrained, rather than retraining the entire classifier. Given that most platforms where such an identification model may practically be deployed are constantly receiving new users, this alone constitutes a major improvement in the practicality of deployment.

6.5 Methodological Novelty

The primary contribution of this paper is in identifying a novel application for a new dataset, which is more than 100x larger than the next largest study in this field. Nevertheless, the unique challenges of this dataset have led us to make advances in the techniques used for identification. For instance, the hybrid featurization of §5.4 offers a significant performance advantage over the motion featurization of Miller et al., while our hierarchical model architecture in §6.3 provides a necessary improvement in scalability. To the best of our knowledge, neither of these techniques have been disclosed in prior work. We later obtained the Miller dataset (N=511), and found that these techniques improved their identification accuracy from 95.0% to 99.8%, demonstrating the significant practical improvement offered by our methods.

7 Evaluation

We evaluated our identification technique using a distributed machine learning cluster of 10 nodes, each with 16 vCPU cores and 128 GB of RAM. The replays of each user were separated into 4 or more distinct sessions, which were reserved for training, clustering, validation, and testing at a ratio of 70-10-10-10. For each user, 150 samples were generated from the training set using the full hybrid featurization method of §5.4. The features of all users were then z-score normalized, and used to train the hierarchical model described in §6.3.

The training process was completed in about 3 hours each for the first and second layers and about 6 hours for the third layer. The final testing process, which required over 90 million classifications to be made, took about 8 hours; an individual user identification requires less than a second.

7.1 Results

Layer	# of Models	Accuracy (per Model)	Accuracy (per Layer)
Layer 1	10	93.1%	90.2%
Layer 2	10	93.1%	90.2%
Layers 1 & 2	20	93.1%	91.0%
Layer 3	5	84.0%	84.0%
Layers 1, 2, & 3	25	91.3%	94.3%

Table 2: Accuracy of each hierarchical model layer per model (i.e., 5.5k users) and per layer (i.e., 55k users).

Table 2 shows the identification accuracy of each layer in the hierarchical model when evaluated using 50 test samples

(100 seconds) per user. An identification accuracy of 90.1% can be achieved using a single layer, with the hierarchical architecture boosting the overall accuracy to 94.3%.

Of course, the accuracy of identification is highly dependent on the number of samples (and thus seconds of data) used to identify a user. Fig. 11 illustrates the identification accuracy in relation to the number of seconds used.

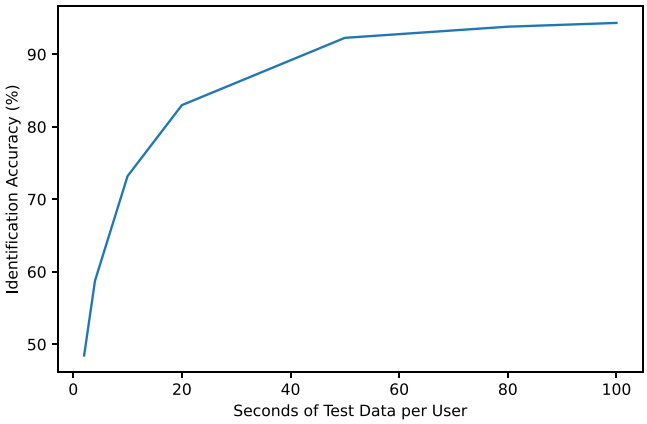

Figure 11: Impact of test sample size on accuracy.

Even with a single sample generated from just 2 seconds of telemetry data, the correct user out of 50,000 is identified about 48.45% of the time. Using 5 samples (10 seconds) of data increases this accuracy to 73.20%, which implies that only a short period of motion information is actually needed to uniquely characterize a user. A single minute of data yields 92.78% identification accuracy, and the full 94.33% accuracy is achieved when 50 samples (100 seconds) of data are used, with rapidly diminishing returns for each sample thereafter.

In some applications, it may be sufficient to output a small number of candidate identities rather than exactly identifying a user. In our evaluation, the correct user is amongst the top 3 candidates identified by the model in 97.25% of all instances.

7.2 Open-World Setting

Thus far, we have evaluated our models under the closed-world assumption, in which we are only concerned with classifying users that have already been seen in the training phase. However, in any realistic deployment, models will often be faced with users that have not previously been encountered. In the open-world setting, models should be able to detect the unseen classes rather than incorrectly identifying them as a previously-seen user. Ideally, the model can then be updated over time to incorporate the new users into the system.

Thankfully, it is well known that statistical techniques can be used to detect instances of concept drift in classification models. For example, Transcend [14] uses a statistical comparison of samples to identify concept drift in malware classification models. Using a similar principle, our hierarchical classification approach is already well suited to detect and reject users not previously seen during training.

To understand the performance of our models in an open-world setting, we performed a second evaluation using 10% of the existing users (5,554) and an equal number of new BeatLeader users not previously seen in training. Each of these users was classified using the first two layers of the hierarchical model. Fig. 12 shows the output confidence of both layers for new and existing users.

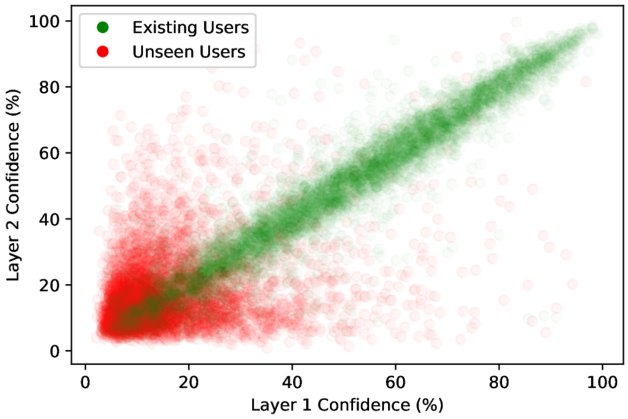

Figure 12: Correlation of layer 1 and layer 2 confidence values for existing and unseen users in the open-world setting.

As illustrated in Fig. 12, users present in the training set demonstrate a high correlation between the confidence of both layers, while previously unseen users show less correlation and have significantly lower confidence overall. Thus, a simple logistic regression model can be trained to determine whether a given user was previously seen. We chose to allocate 90% of the 5,554 new and 5,554 existing users to training, with the remaining 10% for testing. Thus, we trained the model using 4,999 existing users and 4,999 new users, and subsequently tested it using 555 existing users and 555 new users, the results of which are shown in Tab. 3. For each user, the inputs consisted of the max, argmax, and standard deviation of classification confidence values from each layer.

	Existing Users	**Unseen Users**
Classified as Existing	518 (93.3%)	45 (8.1%)
Classified as Unseen	37 (6.7%)	510 (91.9%)

Table 3: Binary classification of existing and unseen users in the open-world setting using logistic regression.

Overall, the logistic regression model was 92.6% effective at determining whether a given user had previously been seen in the training phase. This result should be interpreted in light of the fact that the accuracy of identifying and rejecting new

users cannot reasonably be expected to out-perform the overall 94.3% accuracy of user identification. Thus, our approach could reasonably be deployed in the open-world setting.

7.3 Impact Factors

As explained in §3.2, our dataset contains labeled metadata for a number of user attributes, including device information and some basic demographics. While we avoided using this data in our identification model in order to achieve purely motion-based identification, we later used all of this information to perform a key factor analysis so as to better understand which attributes affect the identifiability of a user. The 15 most important factors are summarized in Fig. 13. This summary evaluates the impact of each factor on the accuracy of layers 1 and 2, as not all users are present in layer 3.

Fig. 13 reveals some interesting trends with respect to the factors which most impacted identification accuracy. Some devices, such as Windows Mixed Reality, are less conducive to identification, perhaps due the device's overreliance on low-quality dead reckoning for tracking. Others, like Valve Index, yield better than average user identification, which may be due to its highly precise outside-in tracking system.

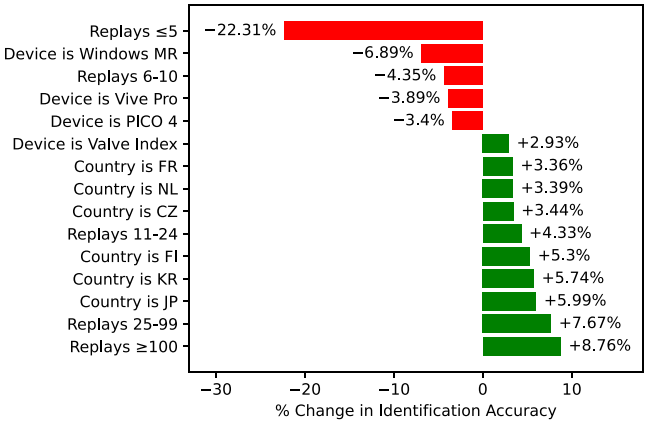

Figure 13: Impact of key factors on identification accuracy.

Users from certain countries, particularly Japan and South Korea, are significantly easier to identify, implying there may be detectable cultural differences in play style. This result is highly statistically significant, with over 99% identification accuracy for users from those two countries.

However, by far the most important factor in determining identification accuracy is the number of total replays observed from a target user, regardless of how many samples were actually used to train the model. Users with 5 or less total replays submitted were significantly harder to identify, while the 5,000 or so users with 100 or more replays could be identified with over 99.5% accuracy. The identification accuracy for users is charted against the number of replays in Fig. 14.

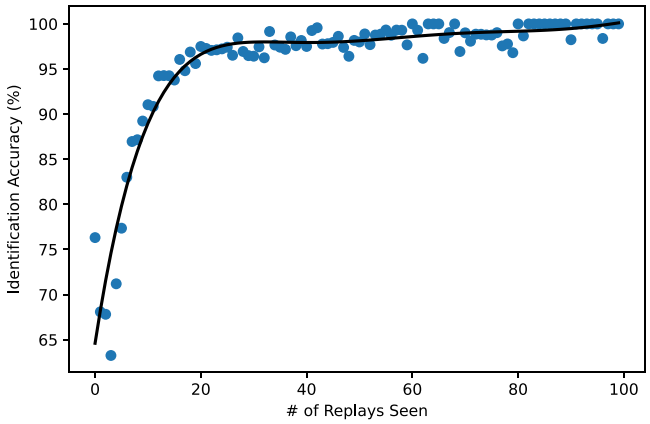

Figure 14: Replays per user vs. identification accuracy.

The clear trend of users with more replays (and thus more time spent in the game) being more easily identifiable is indicative of something other than more data being available, as the full 150 training features can easily be extracted from a single 5-minute session. Rather, it suggests that users with more experience are likely to develop a distinct play style (and reinforce the corresponding muscle memory) over time. Highly experienced players are thus more likely than novices to exhibit a repeatable response to the same stimulus, with veteran users becoming so consistent in their movements that they can be identified with near-perfect accuracy.

8 Explanations

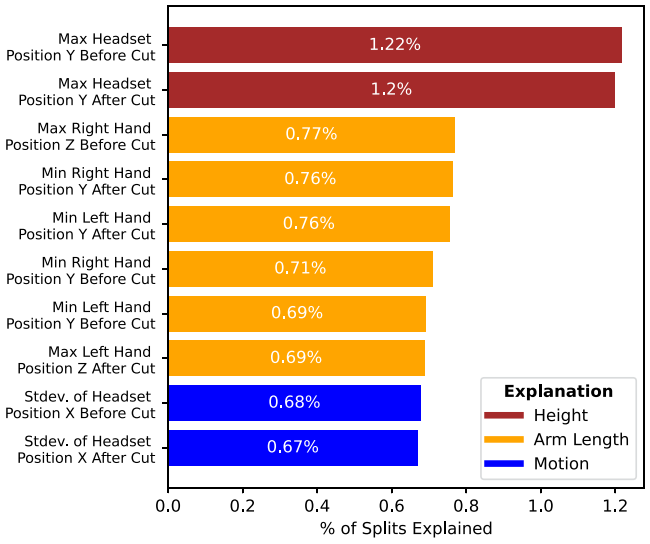

Figure 15: Explanation for 10 most important features.

An additional benefit of using a LightGBM model is the relative ease of explaining the importance of each feature. Fig.

15 shows the percentage of splits attributable to each of the 10 most important features (out of 232) in our final model.

As illustrated in Fig. 15, many of the most important features for identification correspond to obvious physical measurements. For example, the two most important features, which measure the maximum Y-position of the headset before and after the cut, are an obvious proxy for the user's height (and posture). Similarly, the next six most important features seemingly measure the length of the user's arms when furthest outstretched. These first eight features alone account for 6.8% of the splits and 10.2% of the gain of the identification model, providing about 12 bits of real entropy – enough information to accurately identify as many as 4,000 users.

It is no coincidence that these easily understandable features are by far the most important for identification. Unlike motion features, which are highly dependent on the specific action being taken, features that measure some static physical dimension of a user are highly consistent throughout a replay and across sessions. Thus, while the importance of any given motion feature may vary depending on the context of a sample, models can be sure to glean some information from the static features of every sample, regardless of context.

Still, these simple measurements alone hardly account for the identification of 50,000 users. A more complete picture is provided by Fig. 16, which shows the percentage of overall information gain explained by all 232 utilized features.

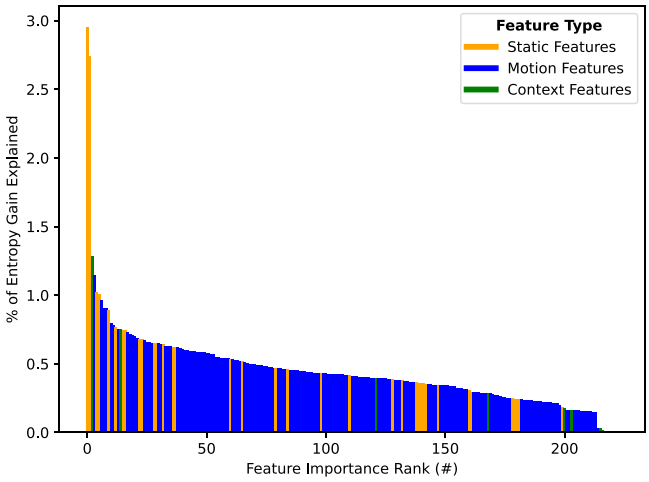

Figure 16: Entropy explained by all feature types.

As is evident in Fig. 16, motion features actually play a major role in identifying users. While static measurements comprise many of the most important features, they account for only 22.9% of the overall performance of the model. Motion features constitute 73.9% of all entropy gain, while contextual features compose the remaining 3.2%. Clearly, motion features actually represent the majority of information used by our identification model, and the task of identifying over 50,000 users would not have been possible without them.

9 Discussion

In light of the fundamental factors working against this result, the identification accuracy achieved by this paper may even be stronger than it initially appears. Unlike the laboratory studies with which this work can be most directly compared, our study endures many of the pitfalls associated with utilizing a dataset from "in the wild." Chief among them is the fact that many users may actually have more than one account or play on multiple devices, resulting in the presence of multiple distinct classes which are in fact identical.

Furthermore, our definition of a "session" is more rigorous than the previous work, with training and testing data for users originating from completely separate days in almost all cases. The largest comparable study (Miller et al. [20]) records 10 short sessions of a user on the same day. Therefore, our results represent the consistent identification of a user across wider periods of time, a task that is far more difficult than correlating motion segments recorded in close succession.

This rigorous session-based split method also provides assurances that player-map preferences are not being used for identification. One reasonable concern with the use of data from Beat Saber is that each player may have their own set of preferred maps, which could, in theory, be used by models as part of the identification process rather than motion alone. Indeed, learning a trivial relationship between a player and their favorite map would undermine the presented results. However, because our dataset consists of leaderboard high scores, we have, at most, only a single instance of a given player playing a given map. Since a replay must occur entirely within one session, our session-based split method ensures that a given player-map replay will be included in either the training or testing sets, but never both. Moreover, the hybrid featurization provides only a single note (2 seconds), from which the map cannot be inferred. Thus, it is certain, for multiple independent reasons, that player-map associations are not being used to artificially inflate identification accuracy.

Lastly, we argue that our work was the first to fully and demonstrably leverage actual movement for identification in VR. As demonstrated in §8, deriving simple measurements like height and arm lengths is sufficient for a model to identify tens or even hundreds of users, as is seen in Miller et al. [20]. This speculation is supported by the fact that users in that study were instructed to simply observe a number of 360-degree VR videos, a relatively static task that does not fundamentally involve much movement. By contrast, identifying 50,000 users would not have been possible without leveraging actual motion patterns, which was made possible by our featurization approach that contextualizes observed motion relative to relevant virtual objects involved in a repeatable activity. The model explainability results of §8 indicate that motion features played a key role in identifying users, accounting for a majority of the model's information gain.

As discussed in §2.3, one cannot easily change their motion patterns, creating the potential for users to be tracked throughout the metaverse. This may, in fact, paint an incomplete picture. Motion patterns are so intrinsically tied to our physical selves that they may soon be able to follow us out of the metaverse and into the real world. Machine learning models designed to extract 3D motion data from monocular video feeds are rapidly improving [28]. We can reasonably extrapolate that it will eventually be possible to match a person's VR movements to surveillance video, and unlike one's face, which can be covered with a mask, no physical countermeasure can reasonably obscure all of a person's movements from public view. While this threat is speculative today, the ability demonstrated in this paper to use motion in a way comparable to other biometrics indicates that we should begin considering the realistic possibility of such scenarios in the pursuit of a future secure and private architecture for the metaverse.

On the positive side, the relatively consistent nature of identifiable motion patterns could provide an unparalleled opportunity for passive authentication in future metaverse applications. Users could benefit from the convenience of having their motion data, fundamentally required for VR functionality, also be used to verify their identity rather than needing to authenticate explicitly. Unfortunately, the laissez-faire nature with which VR motion data is currently broadcasted and shared undermines its future use in authentication; the equivalent would be using fingerprint login on your accounts if pictures of your fingerprints were already made public on the internet. Thus, today's VR users may be paying a heavy early adoption penalty by sharing their motion data with the world before comprehensive defenses are in place.

9.1 Limitations

There are a few notable limitations to the work presented in this paper. Most importantly, several features were used to identify users that are arguably unique to the Beat Saber application. While Beat Saber is currently the most popular VR application in existence, it is not clear, without further investigation, whether these results will generalize to other types of VR applications. Furthermore, the "ground truth" values for some of the attributes reported in §B, namely height and handedness, are based on user-configurable settings, and as such, should be treated as self-reported. Indeed, many players are known to deliberately misconfigure their height setting to obtain a perceived performance advantage.

As described in §4 and quantified in §6.1, deep learning models, though broadly desirable, empirically underperformed tree-based models in our experiments. We found the identification performance of traditional ML models to be sufficient in light of the main focus of this paper, which is to shed light on the sheer magnitude of the privacy concerns implicated by collecting telemetry data in VR applications. An advantage of using LightGBM in this setting is the ability to generate rich model explanations, as presented in §8.

9.2 Future Work

For the reasons discussed above, our results rely on tree-based models rather than using deep learning. In the future, we hope to see deep learning models (especially advanced sequential models like transformer-based models [8]) applied to the same problem, perhaps enabled by a combination of distributed machine learning and more efficient techniques.

There are several interesting applications of our results to Beat Saber specifically, as well as VR gaming in general. These include advanced cheating detection, score prediction, skill-based matchmaking, and map recommendation engines.

By collecting surveys to measure ground truth, future work could aim to infer specific attributes from VR telemetry, including demographics, biometrics, and perhaps even medical conditions, turning VR into a useful measurement tool.

Finally, and perhaps most importantly, we hope to motivate future work into defensive applications and techniques. We hope to see future methods that intelligently corrupt VR telemetry to obscure identifiable properties without impeding their original purpose (e.g., scoring or cheating detection).

10 Conclusion

While perhaps not surprising to experts in biomechanics, the extent to which users can be uniquely identified by observing just a few seconds of motion of their head and hands may indeed be surprising to most. Though many don't presently think of movement patterns as a uniquely identifiable characteristic to the same extent as faces and fingerprints, results like those presented in this paper may serve to change this assumption. Researchers have long speculated that individuals might be identifiable by their movements on a much larger scale than lab studies are able to demonstrate, but datasets with motion from tens of thousands of users did not begin to emerge until the recent widespread adoption of VR.

As we slowly realize the increasing role that virtual reality and the "metaverse" may soon play in our lives, more attention should be given to the security and privacy implications of these platforms. The same telemetry streams which are essential to their operation should in fact be considered highly sensitive data that may reveal a plethora of information about an end user. We hope to motivate further research into privacy-preserving technologies which may be deployed to enable the use of VR without revealing private user information.

Availability

The featurization, normalization, training, and testing scripts used in this paper are available for review in our anonymized repository, along with detailed logs, outputs, and results:

https://github.com/MetaGuard/Identification

Acknowledgments

We would like to acknowledge and thank Xiaoyuan Liu, Charles Dove, Julien Piet, Mark Roman Miller, Gonzalo Munilla Garrido, Ines Bouissou, Beni Issler, Eric Wallace, and Yu Gai for their advice, support, and assistance. We additionally thank Beat Games, ScoreSaber, BeatLeader, and their respective teams, for providing access to the data used in this study. We particularly thank Viktor Radulov and Dziugas Ramonas for their ongoing guidance. We also appreciate the guidance of our anonymous USENIX reviewers and shepherd in helping us refine the paper. This work was supported in part by the National Science Foundation, by the National Physical Science Consortium, by the Fannie and John Hertz Foundation, and by the Berkeley Center for Responsible, Decentralized Intelligence. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of their employer or the supporting entities. We sincerely thank all of the VR users whose published data made this work possible.

References

[1] Khaled Albishre, Mubarak Albathan, and Yuefeng Li. Effective 20 newsgroups dataset cleaning. In *2015 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*, volume 3, pages 98–101. IEEE, 2015. doi:10.1109/WI-IAT.2015.90.

[2] Leo Breiman. Random forests. *Machine learning*, 45:5–32, 2001. doi:10.1023/A:1010933404324.

[3] Tianqi Chen, Tong He, Michael Benesty, Vadim Khotilovich, Yuan Tang, Hyunsu Cho, Kailong Chen, Rory Mitchell, Ignacio Cano, Tianyi Zhou, et al. Xgboost: extreme gradient boosting. *R package version 0.4-2*, 1(4):1–4, 2015. URL: https://cran.microsoft.com/snapshot/2017-12-11/web/packages/xgboost/vignettes/xgboost.pdf.

[4] Linda A Clark and Daryl Pregibon. Tree-based models. In *Statistical models in S*, pages 377–419. Routledge, 2017. doi:10.1201/9780203738535.

[5] James E. Cutting and Lynn T. Kozlowski. Recognizing friends by their walk: Gait perception without familiarity cues. *Bulletin of the Psychonomic Society*, 9(5):353–356, May 1977. doi:10.3758/BF03337021.

[6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. doi:10.1109/CVPR.2009.5206848.

[7] Li Deng. The mnist database of handwritten digit images for machine learning research [best of the web]. *IEEE signal processing magazine*, 29(6):141–142, 2012. doi:10.1109/MSP.2012.2211477.

[8] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. URL: https://arxiv.org/abs/1810.04805.

[9] Peter Eckersley. How Unique Is Your Web Browser? In Mikhail J. Atallah and Nicholas J. Hopper, editors, *Privacy Enhancing Technologies*, volume 6205, pages 1–18. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010. Series Title: Lecture Notes in Computer Science. URL: http://link.springer.com/10.1007/978-3-642-14527-8_1, doi:10.1007/978-3-642-14527-8_1.

[10] Beat Games. Beat Saber. https://beatsaber.com/. URL: https://beatsaber.com/.

[11] Gonzalo Munilla Garrido, Vivek Nair, and Dawn Song. SoK: Data Privacy in Virtual Reality, January 2023. arXiv:2301.05940 [cs]. URL: http://arxiv.org/abs/2301.05940.

[12] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016. URL: https://www.deeplearningbook.org/.

[13] Eakta Jain, Lisa Anthony, Aishat Aloba, Amanda Castonguay, Isabella Cuba, Alex Shaw, and Julia Woodward. Is the Motion of a Child Perceivably Different from the Motion of an Adult? *ACM Transactions on Applied Perception*, 13(4):1–17, July 2016. URL: https://dl.acm.org/doi/10.1145/2947616, doi:10.1145/2947616.

[14] Roberto Jordaney, Kumar Sharad, Santanu K. Dash, Zhi Wang, Davide Papini, Ilia Nouretdinov, and Lorenzo Cavallaro. Transcend: Detecting Concept Drift in Malware Classification Models. pages 625–642, 2017. URL: https://www.usenix.org/conference/usenixsecurity17/technical-sessions/presentation/jordaney.

[15] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. LightGBM: A Highly Efficient Gradient Boosting Decision Tree. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL: https://proceedings.neurips.cc/paper/2017/file/6449f44a102fde848669bdd9eb6b76fa-Paper.pdf.

[16] Adam G. Kirk, James F. O'Brien, and David A. Forsyth. Skeletal parameter estimation from optical motion capture data. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR) 2005*, pages 782–788, June 2005. URL: http://graphics.cs.berkeley.edu/papers/Kirk-SPE-2005-06/.

[17] Lynn T. Kozlowski and James E. Cutting. Recognizing the sex of a walker from a dynamic point-light display. *Perception & Psychophysics*, 21(6):575–580, November 1977. doi:10.3758/BF03198740.

[18] Alex Krizhevsky and Geoff Hinton. Convolutional deep belief networks on cifar-10. *Unpublished manuscript*, 40(7):1–9, 2010. URL: https://www.cs.toronto.edu/~kriz/conv-cifar10-aug2010.pdf.

[19] Jonathan Liebers, Mark Abdelaziz, Lukas Mecke, Alia Saad, Jonas Auda, Uwe Gruenefeld, Florian Alt, and Stefan Schneegass. Understanding User Identification in Virtual Reality

Through Behavioral Biometrics and the Effect of Body Normalization. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pages 1–11, Yokohama Japan, May 2021. ACM. URL: https://dl.acm.org/doi/10.1145/3411764.3445528, doi:10.1145/3411764.3445528.

[20] Mark Roman Miller, Fernanda Herrera, Hanseul Jun, James A. Landay, and Jeremy N. Bailenson. Personal identifiability of user tracking data during observation of 360-degree VR video. *Scientific Reports*, 10(1):17404, October 2020. Number: 1 Publisher: Nature Publishing Group. URL: https://www.nature.com/articles/s41598-020-74486-y, doi:10.1038/s41598-020-74486-y.

[21] Q. ai-Powering a Personal Wealth Movement. VR Headset Sales Underperform Expectations, What Does It Mean For The Metaverse In 2023? Section: Money. URL: https://www.forbes.com/sites/qai/2023/01/06/vr-headset-sales-underperform-expectations-what-does-it-mean-for-the-metaverse-in-2023/.

[22] Vivek Nair, Gonzalo Munilla Garrido, and Dawn Song. Exploring the Unprecedented Privacy Risks of the Metaverse, July 2022. arXiv:2207.13176 [cs]. URL: http://arxiv.org/abs/2207.13176, doi:10.48550/arXiv.2207.13176.

[23] James F. O'Brien, Robert E. Bodenheimer, Gabriel J. Brostow, and Jessica K. Hodgins. Automatic joint parameter estimation from magnetic motion capture data. In *Proceedings of Graphics Interface 2000*, pages 53–60, May 2000. URL: http://graphics.cs.berkeley.edu/papers/Obrien-AJP-2000-05/.

[24] Ken Pfeuffer, Matthias J. Geiger, Sarah Prange, Lukas Mecke, Daniel Buschek, and Florian Alt. Behavioural Biometrics in VR: Identifying People from Body Motion and Relations in Virtual Reality. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI '19, pages 1–12, New York, NY, USA, May 2019. Association for Computing Machinery. doi:10.1145/3290605.3300340.

[25] Frank E. Pollick, Jim W. Kay, Katrin Heim, and Rebecca Stringer. Gender recognition from point-light walkers. *Journal of Experimental Psychology: Human Perception and Performance*, 31:1247–1265, 2005. Place: US Publisher: American Psychological Association. doi:10.1037/0096-1523.31.6.1247.

[26] Viktor Radulov. BeatLeader. URL: https://www.beatleader.xyz/.

[27] Viktor Radulov. BS Open Replay, September 2022. URL: https://github.com/BeatLeader/BS-Open-Replay.

[28] Mingyi Shi, Kfir Aberman, Andreas Aristidou, Taku Komura, Dani Lischinski, Daniel Cohen-Or, and Baoquan Chen. MotioNet: 3D Human Motion Reconstruction from Monocular Video with Skeleton Consistency, June 2020. arXiv:2006.12075 [cs]. URL: http://arxiv.org/abs/2006.12075.

[29] SteamDB. Most played VR Games Steam Charts. URL: https://steamdb.info/charts/?tagid=21978.

[30] Pier Paolo Tricomi, Federica Nenna, Luca Pajola, Mauro Conti, and Luciano Gamberini. You Can't Hide Behind Your Headset: User Profiling in Augmented and Virtual Reality, September 2022. arXiv:2209.10849 [cs]. URL: http://arxiv.org/abs/2209.10849, doi:10.48550/arXiv.2209.10849.

[31] Jan Wöbbeking. Beat Saber generated more revenue in 2021 than the next five biggest apps combined, August 2022. URL: https://mixed-news.com/en/beat-saber-generated-more-revenue-in-2021-than-the-next-five-biggest-apps-combined/.

A LightGBM Hyperparameters

- objective='multiclass'
- boosting_type='goss'
- colsample_bytree=0.6933333333333332
- learning_rate=0.1
- max_bin=63
- max_depth=-1
- min_child_weight=7
- min_data_in_leaf=20
- min_split_gain=0.9473684210526315
- n_estimators=200
- num_leaves=33
- reg_alpha=0.7894736842105263
- reg_lambda=0.894736842105263

B Participant Distribution

Replays	**55,541**
≤ 5	14,945 (26.9%)
6–10	8,639 (15.6%)
11–24	12,495 (22.5%)
25–99	14,012 (25.2%)
≥ 100	5,450 (9.8%)
Platform	**55,541**
SteamVR	42,035 (75.7%)
Oculus	11,269 (20.3%)
Oculus PC	2,223 (4.0%)
Others	14 (0.0%)
Runtime	**55,541**
OpenVR	42,039 (75.7%)
Oculus	13,492 (24.3%)
Unknown	10 (0.0%)
Headset	**55,541**
Oculus Quest 2 (Standalone)	25,857 (46.6%)
Oculus Quest 2 (Quest Link)	4,124 (7.4%)
Valve Index	8,820 (15.9%)
Oculus Rift S	4,483 (8.1%)
HTC Vive	2,408 (4.3%)
Oculus Rift CV1	2,061 (3.7%)
Pico Neo 3	1,595 (2.9%)
Oculus Quest (Standalone)	1,453 (2.6%)
Oculus Quest (Quest Link)	313 (0.6%)
PICO 4	905 (1.6%)
HTC VIVE Pro	728 (1.3%)
HP Reverb G20	644 (1.2%)
HTC Vive Cosmos Elite	395 (0.7%)
HTC VIVE Pro 2	328 (0.6%)
Samsung Windows Mixed Reality	304 (0.5%)
HTC Vive Cosmos	226 (0.4%)
Others	897 (1.6%)
Controller	**55,541**
Oculus Quest Controller	16,449 (29.6%)
Oculus Touch Controller	11,240 (20.2%)
Valve Knuckles Controller	9,805 (17.7%)
Oculus Rift S Controller	3,202 (5.8%)
HTC Vive Controller	1,958 (3.5%)
Pico Neo 3 Controller	1,443 (2.6%)
Oculus Rift CV1 Controller	1,265 (2.3%)
Oculus Quest Controller	665 (1.2%)
HTC VIVE Pro Controller	602 (1.1%)
Others	8,912 (16.0%)
Handedness	**55,541**
Right	53,144 (95.7%)
Left	2,397 (4.3%)
Height	**55,541**
≤ 1.5 m	4,888 (8.8%)
1.5 m – 1.6 m	4,721 (8.5%)
1.6 m – 1.7 m	17,273 (31.1%)
1.7 m – 1.8 m	18,495 (33.3%)
1.8 m – 1.9 m	6,720 (12.1%)
≥ 1.9 m	3,444 (6.2%)
Countries	**55,541**
US	15,142 (27.3%)
DE	2,404 (4.3%)
GB	2,350 (4.2%)
CN	1,964 (3.5%)
CA	1,563 (2.8%)
JP	1,337 (2.4%)
AU	988 (1.8%)
FR	955 (1.7%)
NL	767 (1.4%)
RU	743 (1.3%)
PL	650 (1.2%)
HK	545 (1.0%)
BR	349 (0.6%)
CZ	344 (0.6%)
FI	335 (0.6%)
KR	304 (0.5%)
NO	297 (0.5%)
SE	288 (0.5%)
ES	282 (0.5%)
AT	277 (0.5%)
DK	255 (0.5%)
SG	241 (0.4%)
BE	201 (0.4%)
IT	188 (0.3%)
NZ	159 (0.3%)
TW	157 (0.3%)
MX	137 (0.2%)
CH	116 (0.2%)
HU	114 (0.2%)
CL	111 (0.2%)
IL	101 (0.2%)
TH	88 (0.2%)
AR	88 (0.2%)
IE	86 (0.2%)
UA	85 (0.2%)
PT	76 (0.1%)
Others	21389 (38.5%)