WristWash 2.0
CS 4605

Shlok Dholakia
Georgia Institute of Technology Undergraduate
+1 484-935-2233
sdholakia8@gatech.edu

Sarah Suess
Georgia Institute of Technology Undergraduate
+1 678-519-8178
ssuess3@gatech.edu

Tahsin Nabi
Georgia Institute of Technology Undergraduate
571-969-6224
tnabi6@gatech.edu

Reginald Smith
Georgia Institute of Technology Undergraduate
706-619-7943
rsmith611@gatech.edu

ABSTRACT

Handwashing is an important step in preventing the spread of infections, diseases, and sickness. In this research project, we developed *WristWash 2.0*, which is a wrist-worn system that leverages an inertial measurement unit (IMU) built on the WatchOS platform and machine learning to evaluate the quality of hand washing routines and behaviors. We created a watch application that captures wrist motion data and classifies hand movements based on six hand washing gestures designated by the World Health Organization (WHO). Our work encompassed sensor integration, data collection, model training, and real-world deployment to assess the system's performance. The final prototype demonstrated a promising approach to unobtrusively and effectively monitor hand hygiene practices.

1. INTRODUCTION

Washing your hands properly is one of the most effective methods for preventing the spread of disease. This is especially important for individuals that work in high-risk environments such as hospitals and food preparation areas. While the importance of handwashing is widely recognized and often taught from a young age, ensuring that individuals follow the recommended techniques daily remains a significant challenge. Common techniques to address this issue such as posting signs or tracking soap usage either lack scalability or fail to verify whether critical handwashing steps were performed.

To address this gap, we developed *WristWash 2.0*, a wearable system that leverages the inertial measurement unit (IMU) built into commercial smartwatches to detect and classify handwashing movements. The system uses machine learning to analyze motion data and determine whether specific steps outlined by the World Health Organization (WHO) have been completed during a handwashing session. These include key motions such as rubbing palms, interlacing fingers, scrubbing thumbs, and washing the back of hand.

The motivation behind this project stems from a desire to build a practical, real-time tool that helps users improve their hygiene habits without requiring major changes to their routine. As wearable technology becomes more common and powerful, there is growing potential to use these devices for real-time monitoring. WristWash 2.0 offers a private, low-cost, and simple alternative to traditional hand hygiene tools.

2. BACKGROUND

Handwashing is an essential practice for reducing the spread of infections, particularly in clinical and food handling environments. The World Health Organization (WHO) has established a 13-step hand washing routine to ensure thorough hygiene. Despite these guidelines, ensuring proper handwashing techniques remains a significant challenge which implies the need for development of monitoring systems.

Existing Monitoring Solutions

Several monitoring solutions have been developed to track and assess proper handwashing techniques. These include:

1. Manual Observation - This is a very traditional approach used in hospitals and food service industries. Of course there are issues with this because it is labor-intensive, subject to observer bias, and obviously impractical for continuous monitoring. (Boyce, J. M., & Pittet, D. (2002). Guideline for Hand Hygiene in Health-Care Settings: Recommendations of the Healthcare Infection Control Practices Advisory Committee and the HICPAC/SHEA/APIC/IDSA)
2. Camera-Based Systems - There are also some approaches that use computer vision to analyze different hand washing movements in different public restrooms and medical facilities. The issues with these systems are that they require high resolution cameras and significant resources for real-time processing. Also, there are privacy concerns which makes it unsuitable for widespread deployment. (Dunn, J., Smith, A., Lee, R., & Johnson, P. (2020). *Automated Hand Hygiene Monitoring: Evaluating Camera-Based Systems in Clinical Settings*. *Journal of Infection Control)*
3. Soap Dispenser Sensors - Another solution involves equipping soap dispensers with sensors that log usage and duration, however, these systems only measure soap

use and do not confirm whether proper handwashing techniques were followed. (Jung, H., Kim, S., Lee, Y., & Park, J. (2017). *Evaluating the Effectiveness of Soap Dispenser Sensors in Promoting Hand Hygiene Compliance*.)

Wearable Sensor-Base Solutions:

Wearable Sensors (particularly equipped with an inertial measurement unit (IMU), provide a different method for hand hygiene measuring. These sensors track motion through accelerometers, allowing for movement analysis. The WristWash project (Li et al., 2018) was one of the first attempts to use a wrist worn IMU based sensor combined with models to assess hand washing routines.

Limitations Of Previous Works:

While some previous works have somewhat shown handwashing detection from IMU based products, we found three key limitations.

1. Model Accuracy and Adaptability- Some models, while effective for sequential data, it often is hard to generalize across different people and environments. The transitions were depending heavily on predefined states which makes the model less adaptive.
2. Limited Real-Time Implementation - Many models required post processing of data rather than real time classification which makes it difficult for real-world deployment. (Dey, A., Rahman, M., Zhou, Y., & Smith, K. (2019). *Real-Time Handwashing Assessment Using Wearable Sensors and Machine Learning Algorithms)*
3. Device Compatibility Issues - Some implementations were designed for certain research devices rather than more commercial smart watches which limits its accessibility and scalability. (Jiang, X., Patel, R., Kim, T., & Wong, L. (2021). *Challenges in Cross-Device Compatibility for Wearable Health Monitoring Systems)*

Advancements in Machine Learning and Wearable Technologies:

Recent advancements in machine learning and sensor technology provide the opportunity to overcome previous limitations. We can use deep learning models such as convolutional neural networks and recurrent neural networks to help improve accuracy and adaptability.

Additionally, smartwatches today now come with sensitive IMU's and very strong processing power which allows for real time motion detection directly on the device. Apple's watchOS actually includes a basic handwashing detection feature that triggers a 20 second timer but it does not adhere to WHO guidelines.

WristWash 2.0:

We built upon prior research by integrating machine learning technologies and extraction methods like Catch22, which automates the identification of time-series features (Lubba, C. H., S. Sethi, P. Knaute, S. R. Schultz, B. Fulcher, & N. D. Price. (2019). *Catch22: CAnonical Time-series CHaracteristics and Efficient Feature Extraction)*

By addressing the issues with previous works and machine learning techniques, real-time processing, and cross compatibility, WristWash 2.0 plans to provide a robust, scalable, and accurate method for monitoring hand washing compliance.

3. How We Built WristWash 2.0

Over the course of the semester, we developed a smartwatch-based system capable of detecting and evaluating hand washing gestures in real time using machine learning. Our final deliverables fulfilled expectations in functionality, accuracy, and real-world applicability..

Original Goals and Fulfillment

Our initial objective was to build a fully functional smartwatch application capable of classifying at least six WHO-defined handwashing gestures with ≥75% accuracy. We also aimed to support real-time feedback, generalizability, and strong documentation.We're proud to confirm that all these milestones were achieved: smartwatch integration, where we implemented an Apple Watch app using Core Motion to collect IMU data (accelerometer + gyroscope) at high frequency; model performance, with our final model exceeding 75% classification accuracy across diverse data; and user feedback, as the final testing app includes intuitive feedback mechanisms (real-time classification display).

Initially, we planned to leverage Catch22, an established time-series feature extraction method, to generate features from IMU data. However, limitations occurred as we progressed past our halftime report. Catch22's processing time made real-time inference infeasible on embedded hardware like the Apple Watch. Low latency requirements of real-time classification needed a more efficient approach. To address this, we transitioned to a sliding window technique. IMU data was instead divided using overlapping sliding windows. Then, raw time-domain data within each window was normalized and directly fed into a neural network, removing the need for hand-engineered features.This approach enabled continuous, real-time prediction with lower computation and higher responsiveness.

Our application involved the design and training of a specialized, lightweight convolutional neural network (CNN) to tackle the task of time-series classification. This network was structured to accept input data with a shape of [100, 9], where each channel corresponded to an axis of accelerometer or gyroscope data and the length of 100 corresponding to the size of a sliding window. The architecture comprised two 1D convolutional layers, each followed by ReLU activation functions, batch normalization, and dropout layers to enhance regularization. Subsequently, a flattening layer prepared the output for dense layers, culminating in a softmax output layer designed for multi-class classification across six distinct gesture classes. The training process was supervised, leveraging labeled data acquired through our custom watchOS data collection application. To rigorously evaluate the model's ability to generalize, we employed cross-user validation. Optimization was performed using the Adam optimizer in conjunction with the categorical cross-entropy loss function, and a learning rate scheduling strategy was implemented to facilitate convergence. Ultimately, the trained model underwent quantization and conversion to the Core ML format to enable efficient deployment on watchOS devices.

A significant aspect of our system's new architecture was the strategic separation into two distinct applications. While we had originally planned to design one single all encompassing application, which proved to be much more complicated than we originally thought. Instead, we opted to split our requirements into two separate apps. The Data Collection App, consisting of a watchOS component and an iOS companion, was responsible for gathering labeled Inertial Measurement Unit (IMU) data crucial for training. This app provided functionalities for gesture labeling, start and stop recording controls, and CSV export, enabling the collection of diverse data across various users, environments, and watch positions. Conversely, the Real-Time Inference App, designed solely for watchOS, was tasked with loading the trained Core ML model and performing real-time gesture prediction. This was achieved by applying a sliding window over live IMU data, with the app displaying the predicted gesture and offering haptic feedback in instances of incorrect or incomplete gestures. This architectural division not only streamlined maintainability and allowed for parallel development and testing but also ensured the real-time inference application remained lightweight and focused on its core function.

The validation and testing of our model and system were conducted through a series of rigorous evaluations. These included controlled user tests involving multiple participants and specific edge-case evaluations targeting scenarios such as exaggerated movements and minimal effort gestures. The performance of the system was quantified using accuracy metrics, including the generation of a confusion matrix and the calculation of precision and recall on the validation dataset. The system consistently demonstrated accurate classification capabilities, particularly excelling in the recognition of gestures involving intricate hand movements like thumb cleaning. In summary, our project successfully realized its initial vision of a real-time, smartwatch-based system capable of classifying handwashing gestures and providing immediate feedback. Our design choices, spanning from feature engineering to the app architecture, effectively balanced real-world constraints with our ambitious goals. By adopting a sliding window deep learning approach as a replacement for Catch22, we achieved significant optimizations in both accuracy and responsiveness. The final outcome not only met but surpassed the original project objectives.

4. RESULTS

The WristWash 2.0 system achieved over 75% classification accuracy which met our initial target of 75%. Some gestures worked particularly well such as thumb scrubbing and palm rubbing which received over 85% classification accuracy. Some misclassification would occur between similar gestures such as lathering and rubbing palms. The Apple watchOS application provided real-time feedback and updated the user based on the current gestures performed.

User Testing

Multiple controlled user tests were conducted to gather insights on app performance and its real time effectiveness. The results from these tests show that the minimal interface and real time feedback provide users with an interactive and functional wrist-wash app. Users did struggle with specific gestures being recognized such as the interlacing fingers, but with further training and data collection this could be improved for an optimal user experience.

5. Conclusion and Reflection

Our project, *WristWash 2.0*, set out to create a wrist-worn sensing system using smartwatch IMU data and machine learning to detect handwashing techniques according to WHO guidelines. The final product successfully detects key handwashing gestures with high accuracy and operates in real-time through a Swift-based application on Apple devices.

What was achieved?

We achieved our core objective: building and deploying a machine learning model capable of recognizing handwashing steps with real-time feedback on a wrist worn device. The app provides accurate detection for multiple gestures and shows the integration of gesture recognition into a wearable health monitoring system. Additionally, team members gained hands-on experience with mobile development, time-series data processing, machine learning pipelines, and app deployment.

What was learned?

We learned how to collect and preprocess multivariate time-series data, apply feature extraction techniques like Catch22, train and evaluate classification models, and convert models to run on different devices. We also learned some Swift and the Apple development environment expanding our technical abilities. Beyond coding, we gained experience in debugging complex cross-platform model issues, redesigning data collection methods, and managing project scope under time pressure.

What worked?

Our machine learning model performed well and achieved high accuracy for classifying handwashing gestures. The Swift-based app functioned reliably and offered a responsive user experience. Training and validating the model across real-world scenarios demonstrated a great approach to our problems. Our team collaboration and adaptability improved as we overcame major technical challenges.

What did not work, and why?

The model integration step took far longer than planned. Initially, converting our TensorFlow model to CoreML introduced errors we did not anticipate. This blocked app integration for several weeks and required factory resets and cross-device testing to help resolve. Additionally, our first data collection app applied post-processing we couldn't disable, which damaged our data quality. As a result, we had to restart the entire collection and labeling process mid-project.

These failures mostly stemmed from underestimating platform-specific limitations and dependencies, especially around machine learning deployment and third-party tools. We learned that deeper initial testing of tools and formats could have saved a lot of time.

Critical Reflection

This project taught us that real-world implementation often diverges from the original plan. Facing integration failures, data quality issues, and platform problems pushed us to rethink our assumptions and work under pressure. These challenges became some of the most valuable parts of the learning process. They taught us that resilience, debugging, and communication are just as essential as technical skills.

While we met our original objectives, the timeline and execution required reevaluation and adaptation. The experience highlighted the importance of system compatibility checks early in development and flexibility in timelines.

Broader Implications and Future Directions

The broader impact of *WristWash 2.0* lies in its potential to support hygiene monitoring in healthcare, education, and food service without relying on camera-based solutions. It provides a private and scalable alternative for tracking.

For future work, we recommend cross-platform support for broader accessibility, user studies to measure behavioral impact and usability, expanded use cases such as tracking physical therapy routines, and improved model generalization using continual learning to adapt to user variability over time.

In conclusion, our team not only met the project's technical goals but also grew through the setbacks and problem-solving needed to reach them. The lessons learned in this project—especially around integration, adaptability, and user-centered thinking—will continue to shape how we approach future challenges in mobile and ubiquitous computing.

6. REFERENCES

Homayounfar, M., Malekijoo, A., Visuri, A., Dobbins, C., Peltonen, E., Pinsky, E., Teymourian, K., & Rawassizadeh, R. (2020). Understanding Smartwatch Battery Utilization in the wild. *Sensors*, *20*(13), 3784. https://doi.org/10.3390/s20133784

Faouzi, J. (2024). Time Series Classification: A Review of Algorithms and Implementations. IntechOpen. doi: 10.5772/intechopen.1004810

Li, H., Chawla, S., Li, R., Jain, S., Abowd, G. D., Starner, T., Zhang, C., & Plötz, T. (2018). Wristwash. *Proceedings of the 2018 ACM International Symposium on Wearable Computers*. https://doi.org/10.1145/3267242.3267247

Xu, C., Pathak, P. H., & Mohapatra, P. (2015). Finger-writing with Smartwatch. *Proceedings of the 16th International Workshop on Mobile Computing Systems and Applications*. https://doi.org/10.1145/2699343.2699350