HoloAuth: Hololens2 Sensor Streaming Application and User Identification

Chandrika Mukherjee
cmukherj@purdue.edu
PUID - 32808289
Dept. of Computer Science
Purdue University

***Abstract*—In the past few years, we have seen people becoming more interested in the various applications of AR-VR domain. Not only that, big internet companies have started investing more for AR-VR projects. User's physical movements and characteristics are continuously monitored in AR-VR systems to build an immersive experience for the user. The applications are not only limited to games and fitness, but the area has expanded to medical, military use, and even for simulating outer space environment. All these things are possible only because a variety of sensors are deployed in the AR-VR devices. HoloLens2, developed by Microsoft is one of the most popular AR/MR devices available in the market. In this paper, I present HoloAuth - a sensor streaming application and user identification system in HoloLens2 and show how some sensor data can be utilized to eventually uniquely identify a person by only based on their head movements. Gyroscope based SVM model achieves around 94% accuracy in identifying a user and also in identifying a specific head gesture made by a particular user. Using head origin data, HoloAuth achieves respectively 89% and 78% accuracy for the cases mentioned above.**

I. Introduction

For the past few years, big internet companies like Meta, Apple, Google inclined their focus towards not only developing Virtual Reality(VR) and Mixed Reality(MR) applications on smartphone, but also in developing Headsets. These headsets can also come up with hand-held controllers to provide more accurate immersive experience. Virtual Reality(VR) headsets can provide a user with immersive feel of virtual world. Augmented Reality(AR) and Mixed Reality(MR) terms are sometimes used interchangeably as both of the systems overlay digital information over physical world. However, there is a slight difference between AR and MR, where MR enables user to interact with surroundings by tracking user's movement, thereby creating more immersive experience. For an example - a user uses Amazon app feature to place a virtual furniture in living room before buying is an example of AR application whereas Microsoft HoloLens2 is a MR device, where user can interact with digital objects by moving his hand, eye or head.

Microsoft released HoloLens2 in 2019 which was a major upgrade on their previous release HoloLens-1st gen (released in 2016). To name a few improvements of HoloLens2 over its previous version - provides bigger field of view, introduced eye tracking, fully articulated hand tracking, improved hologram stability and user will be able to wear the head mounted display (HMD) for longer time.

HoloLens2 comes up with various sensors to provide mixed reality experience - 4 visible light cameras for head tracking, 2 infrared cameras for eye tracking, 1-MP time-of-flight depth sensor, inertial measurement unit (IMU) conists of accelerometer, gyroscope, magnetometer and camera of 8-MP stills, 1080p30 video. Research Mode was introduced in HoloLens (1st gen) to give access to some sensors solely for research purpose and not for deployment in market. Using Research Mode in HoloLens2, we can get access to IMU sensors, depth camera, visible light environment tracking cameras and two versions of the IR-reflectivity stream.

Till the date, the most common authentication system in AR/VR devices is based on password/PIN. It is highly cumbersome to input password and PIN using virtual keyboard, and therefore, users choose weaker passwords. In a public environment, slow speed in entering password can leave the user vulnerable to external observer attacks like shoulder surfing [1]. Over the years, there have been multiple research studies that have tried to defend against shoulder surfing attacks [2], [3]. According to [4], continuous biometric authentication/movement recognition system should be deployed in VR, however, system should also provide provable privacy guarantee to the sensor data, otherwise, malicious apps can utilize user tracking information to profile a user.

GlassGesture, an authentication system developed for Google Glass, [5] proposed to use combination of security questions and head gesture to authenticate a user. Head movement is used as it is hand-free, easy to perform and accelerometer, gyroscope can detect head movements due to their high electromagnetic sensitivity.

In this work, I implemented a sensor streaming application on TCP to obtain data in real-time while user wears the HoloLens2 and does any activity. I have utilized the Research Mode in HoloLens2 to particularly capture data from IMU sensors along with head tracking and eye tracking information. The HoloAuth system recognizes user based on their head movements. Particularly, I collected 2 sets of data from 2 different users and 1 set of data from another user. Each set contains 2-4 repetitions of simple head gestures (circle, triangle, w-shape, square). The paper describes the effectiveness of HoloAuth in recognizing user based on their head movements.

II. Related Work

Garrido et al. [4] in the SoK mentions that few minutes of user specific data tracking can give attackers enough information to profile a person in VR systems. This can essentially leak user's sensitive information and therefore privacy is a concern in this domain. They also suggested researchers to explore biometric movement authentication.

Stephenson et al. in their SoK [1] mentions that password based authentication in AR-VR is highly undesirable. Password based authentication is vulnerable to shoulder surfing attack, also it is difficult to input stronger password faster in the AR-VR environment as user needs to interact with virtual keyboard. They suggested researchers to explore federated login for AR-VR domain. Along with that, they also motivated researchers in exploring user authentication system based on their biometrics. Developer's survey in paper [1] shows that developers of AR-VR system choose password based authentication because of ease-of-implementation.

GlassGesture paper [5] demonstrates a system that can recognize user's head gesture and authenticate a user using combination of security questions and user head movement in Google Glass. They report high accuracy in identifying user by training one-class SVM classifier using gyroscope and accelerometer data based on user's head movement. This paper primarily motivated me in exploring the research mode sensors data to identify users based on their head movements in HoloLens2. Till the date, HoloLens2 uses password/PIN based authentication mechanism which is highly inefficient.

III. HoloAuth System Design

In this section, I present the system design (fig-1) of HoloAuth. There are total three sections in the system - Data Collector, Head Gesture Recognition, and Head Gesture Based Authentication.

A. *Data Collector -*

To analyze user's head movement, first I had to implement a data collection app in HoloLens2. To access the data from IMU sensors, Research Mode needed to be enabled in the HoloLens2 device. I implemented [6] a Unity 2019 mixed reality app for HoloLens2 (fig-3), and a server application to run on laptop. The app utilizes Research Mode API [7] by Microsoft to enable raw data streaming from Research Mode sensors. Based on choice, user can stream data from above

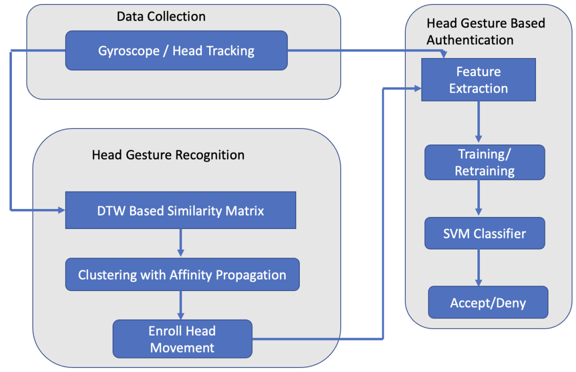

Fig. 1. System Architecture

mentioned sensors, and server will be able to capture those data. A sample of log data can be found in fig-2. The details of enabling eye tracking, research mode and windows device portal are described in detail in the appendix-A of this paper.

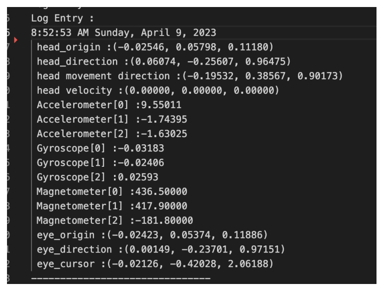

Fig. 2. Sample Log

Fig. 3. Screen Capture From HL2 App

B. *Head Gesture Recognition -*

1) ***Gesture Selection :*** GlassGesture [5] mentions that head gestures are different from traditional hand gestures as head gestures mainly consists of rotational movements. People use many AR-VR applications in shared environment, therefore,

head gestures must be carefully chosen - otherwise people may not be able to comfortably perform the gestures. For this experiments, I have selected very less time consuming and intuitive head gestures (circle, triangle, square and w-shape). To complete a gesture, it only takes 5-6 seconds. On the contrary, in case of password based authentication system, interacting with virtual keyboard is much more time consuming and very less approachable for people with disabilities [1].

*2) **Gesture Recognition**:* Gesture Recognition utilizes two algorithms to enroll head gestures.

- **Dynamic Time Warping -**
  Head gestures are time series data. To compare a head gesture with other, this module uses Dynamic Time Warping (DTW) algorithm. DTW is used to measure similarity between two temporal sequences which may vary in length. DTW similarity score between two time sequences X and Y is computed as Euclidean distance between optimally aligned X and Y, where π is the optimal aligned path.

$$DTW(X,Y) = \sqrt{\sum_{(i,j)\in\pi} \|X_i - Y_j\|^2}$$

Fig. 4. DTW Similarity Score

- **Affinity Propagation -**
  Affinity Propagation (AP) [8] is a clustering algorithm which takes a similarity measure of data points as input. Real valued messages are exchanged between the data points until a high quality set of exemplars (representative of each cluster) and corresponding clusters emerge. AP does not require the number of clusters to be pre-determined before running the algorithm unlike k-means or k-medoids.

- **Enrollment of Head Movements -**
  Head gestures are compared against each-other and DTW similarity score is obtained for each pair of gestures, therefore, forming a similarity matrix. This matrix is fed as input to AP algorithm to find clusters. Similar head gestures are clustered together and according to [9], AP is proposed as an effective method.

  Manual labelling is prone to error. Say, a user is instructed to perform square head gesture, but the head movement done for a particular instance is similar to the user's circular head movement. In that case, if we would have labelled them manually, we could have marked the movement as square which is not correct. But as we use Affinity Propagation, because of similarity, that head movement will be clustered with the circular head movements of the user. In this way, we can take care of smaller details of each head movement and create true label for each gesture.

  Affinity Propagation cluster centers are stored as a gesture template in the HoloAuth System to be used for recognition later.

C. Head Gesture Based Authentication -

*1) **Gyroscope and Head Origin Data** :* As previously mentioned in the Gesture Selection section above that head gestures mainly consists of rotational movements. GlassGesture [5] has provided evidences that head gestures can be detected through gyroscope readings. This paper also mentions that distinguishable gyroscope readings can be captured while user makes any head movement compared to relatively noisy accelerometer readings. Therefore, in this paper, Gyroscope reading is selected for head movement recognition. This work also uses head tracking origin data to separately create gesture templates and evaluate the performance against gyroscope based template to recognize user.

*2) **Feature Selection** :* For each Gyroscope or Head Origin time series data, some statistical features are extracted. Each of this type of data point consists of X, Y and Z values. For each axis, 23 statistical features, so, total 69 features are extracted for single gesture. Selected features are -

- kurtosis
- mean
- mean_abs_change
- mean_change
- mean_second_derivative_central
- median
- number_cwt_peaks
- number_peaks
- abs_energy
- absolute_sum_of_changes
- longest_strike_above_mean
- longest_strike_below_mean
- ratio_value_number_to_time_series_length
- root_mean_square
- sample_entropy
- skewness
- standard_deviation
- sum_of_reoccurring_data_points
- sum_values
- variance
- variance_larger_than_standard_deviation
- variation_coefficient
- median_abs_deviation

Above mentioned features are extracted using tsfresh [10] python package.

*3) **Training and Classification**:* The training phase happens offline using Support Vector Machine (SVM) classifier. For biometric-based authentication, SVM has been widely used. Grid search has been applied to find out optimal parameters for SVM classifier (SVC). RBF kernel with 5-fold cross-validation is used. The trained model is used to classify the test-samples. Samples are labelled in two ways - 1) based on user id and 2) cluster id based on affinity propagation. HoloAuth compares the true label of the head gesture with predicted label to evaluate performance of the model.

IV. Results and Analysis:

A. *Different Sensor Readings*

Figure 5, 6 demonstrate different sensor readings for a circular head gesture of a user. Figure 7 and 8 show sensor readings for triangular head gesture, made by the same user. Circular gesture was 5s long and triangular gesture was around 4 seconds long. We can observe that same sensor readings for different gestures are very different.

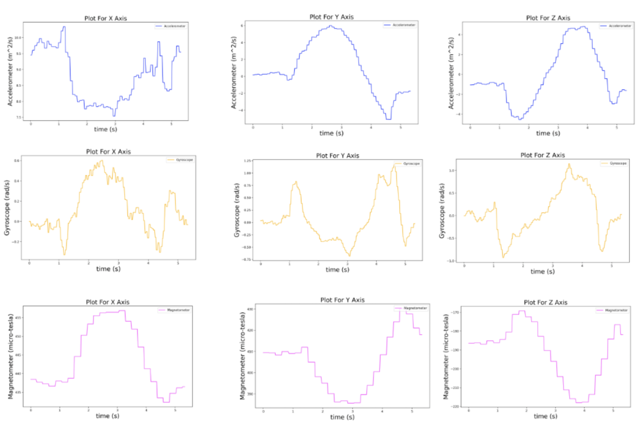

Fig. 5. Accelerometer, Gyroscope, Magnetometer Reading (Circular Gesture)

B. *Affinity Propagation Analysis*

For this experiment, First day, 4 repetitions of each circular, triangular, w-shaped and 2 repetitions of square head gestures were taken from user1. DTW was applied on gyroscope signal data of these 14 gestures to get similarity matrix, then Affinity Propagation was executed on that similarity matrix. The dataset was divided into 3 classes, even if user1 performed 4 different gestures. Figure 9 shows that all circular, triangular and w-shaped signals formed different clusters 0, 1 and 2 respectively. However, the square signals didn't form separate cluster, but those gestures were assigned to the cluster 1 (cluster of triangular signal).

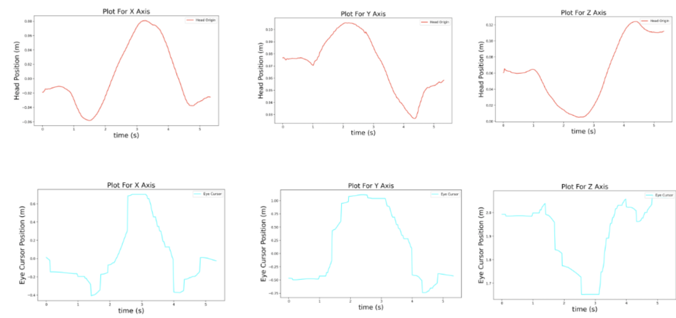

Fig. 6. Head Origin, Eye Cursor Reading (Circular Gesture)

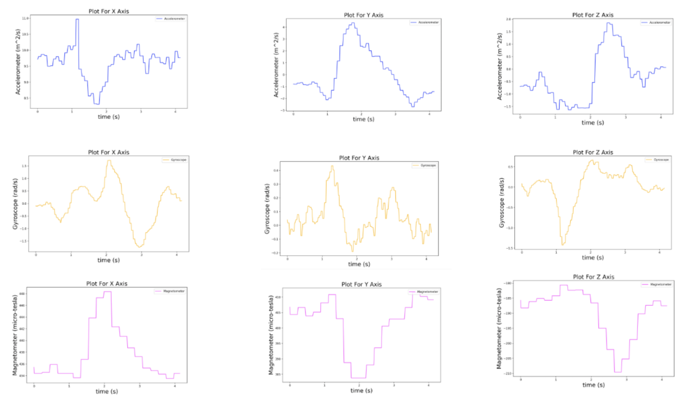

Fig. 7. Accelerometer, Gyroscope, Magnetometer Reading (Triangular Gesture)

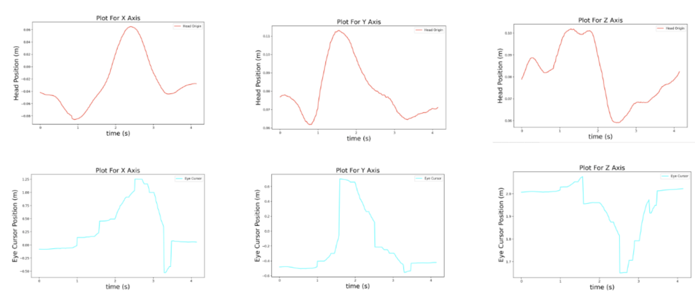

Fig. 8. Head Origin, Eye Cursor Reading (Triangular Gesture)

Next day, 4 repetitions of each circular, triangular, w-shaped, and square - total of 16 new head gestures were taken from user1. Then the new head gestures were tested against previously defined cluster centers based on older data of user1. Figure 10 shows that new circular, triangular and w-shaped data were assigned to its corresponding classes. Again, the square head gestures were assigned to triangular cluster. With this observation, we can say that a user has a unique way to perform a head gesture. Corresponding Results with head origin will be available in appendix-A section.

To understand performance of affinity propagation a little more, another test was performed. Here, similar type of head gestures were selected from separate user (user2) and the dataset of user2 was tested against the clusters defined in figure 9. Figure 11 shows that even if user2 performed circular gesture, it was not assigned to the circular gesture cluster of user1, rather it was assigned to triangular cluster. Even most of the gestures were recognized as triangle.

	True Label	Assigned Cluster	Assigned User	Assigned Shape
0	cir1_user1	class 0	user1	circle
1	cir3_user1	class 0	user1	circle
2	cir2_user1	class 0	user1	circle
3	cir4_user1	class 0	user1	circle
4	tri1_user1	class 1	user1	triangle
5	tri3_user1	class 1	user1	triangle
6	tri4_user1	class 1	user1	triangle
7	tri2_user1	class 1	user1	triangle
8	w2_user1	class 2	user1	w-shape
9	w4_user1	class 2	user1	w-shape
10	w3_user1	class 2	user1	w-shape
11	w1_user1	class 2	user1	w-shape
12	square2_user1	class 1	user1	triangle
13	square1_user1	class 1	user1	triangle

Fig. 9. AP Clustering on User1 [Gyroscope Data]

	True Label	Assigned Cluster	Assigned User	Assigned Shape
0	tri4_user1_2	class 1	user1	triangle
1	w1_user1_2	class 2	user1	w-shape
2	cir4_user1_2	class 0	user1	circle
3	tri1_user1_2	class 1	user1	triangle
4	cir1_user1_2	class 0	user1	circle
5	w4_user1_2	class 2	user1	w-shape
6	square2_user1_2	class 1	user1	triangle
7	square1_user1_2	class 1	user1	triangle
8	cir2_user1_2	class 0	user1	circle
9	tri2_user1_2	class 1	user1	triangle
10	w2_user1_2	class 2	user1	w-shape
11	square4_user1_2	class 1	user1	triangle
12	square3_user1_2	class 1	user1	triangle
13	w3_user1_2	class 2	user1	w-shape
14	tri3_user1_2	class 1	user1	triangle
15	cir3_user1_2	class 0	user1	circle

Fig. 10. AP Clustering Test Result on User1 with Different Dataset [Gyroscope Data]

	True Label	Assigned Cluster	Assigned User	Assigned Shape
0	w2_user2	class 2	user1	w-shape
1	cir1_user2	class 1	user1	triangle
2	tri1_user2	class 1	user1	triangle
3	square2_user2	class 2	user1	w-shape
4	square1_user2	class 1	user1	triangle
5	cir2_user2	class 1	user1	triangle
6	w1_user2	class 1	user1	triangle
7	tri2_user2	class 1	user1	triangle

Fig. 11. AP Clustering Test Result (User2 Data with User1 Clusters) [Gyroscope Data]

C. Affinity Propagation By Mixing Data from Different Users

Head gestures were captured from three different users. 2 users provided 2 different sets of head movements and another user provided 1 set of head movements. 1 set of all the gestures from each user were mixed, and based on affinity propagation, 8 different clusters were created - and a very interesting observation can be made that a particular cluster only contains repetitions from a single user (fig-12). Also, except for three instances (1^{st}, 19^{th} and 26^{th}row), rest of the times, correct user label was assigned. On the other hand, in the experiment with head origin data, it correctly assigned user labels for all the instances (fig-20).

D. Analysis of Same Head Gestures of Different Users

Figure 13 shows that corresponding circular and triangular movement of different users are very different. One evident difference between user1 and user2 based on this observation is that user1 completes the gesture faster than user2.

E. Analysis of Different Head Gestures of Same User

Figure 14 shows that different gestures from same user are also very different.

F. Analysis of Same Head Gestures of Same User

Figure 15 shows that similar head gestures of a user follow similar pattern. In this figure different repetitions of circular head movement of user1 clearly shows the similarity with each other.

G. SVM Classifier Accuracy

The true labels of all the data were obtained based of affinity propagation clustering. The whole data set obtained from three different users were splitted into training and testing samples. Model was trained using SVC classifier and rbf kernel with 5-fold cross validation. Output label of test samples were predicted as in what cluster id can be assigned to a test sample.

	True Label	Assigned Cluster	Assigned User	Assigned Shape
0	cir1_user1	class 0	user1	circle
1	cir1_user3	class 2	user2	circle
2	cir1_user2	class 2	user2	circle
3	cir3_user1	class 0	user1	circle
4	cir3_user3	class 1	user3	circle
5	cir2_user1	class 0	user1	circle
6	cir2_user2	class 2	user2	circle
7	cir2_user3	class 1	user3	circle
8	cir4_user1	class 0	user1	circle
9	tri1_user3	class 3	user3	triangle
10	tri1_user2	class 2	user2	circle
11	tri1_user1	class 4	user1	triangle
12	tri3_user3	class 3	user3	triangle
13	tri3_user1	class 4	user1	triangle
14	tri4_user1	class 4	user1	triangle
15	tri2_user2	class 2	user2	circle
16	tri2_user3	class 3	user3	triangle
17	tri2_user1	class 4	user1	triangle
18	w2_user1	class 5	user1	w-shape
19	w2_user2	class 6	user3	w-shape
20	w2_user3	class 6	user3	w-shape
21	w4_user1	class 5	user1	w-shape
22	w3_user1	class 5	user1	w-shape
23	w3_user3	class 6	user3	w-shape
24	w1_user1	class 5	user1	w-shape
25	w1_user3	class 6	user3	w-shape
26	w1_user2	class 6	user3	w-shape
27	square3_user3	class 7	user3	square
28	square2_user1	class 4	user1	triangle
29	square2_user2	class 2	user2	circle
30	square2_user3	class 7	user3	square
31	square1_user1	class 4	user1	triangle
32	square1_user3	class 7	user3	square
33	square1_user2	class 2	user2	circle

Fig. 12. AP Clustering on Data from Multiple Users [Gyroscope Data]

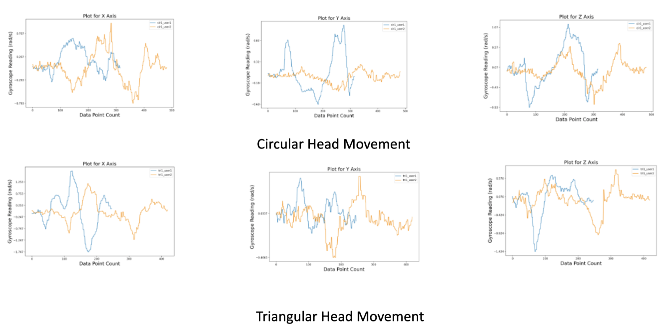

Fig. 13. Same Gesture, Different User (Gesture Comparison)

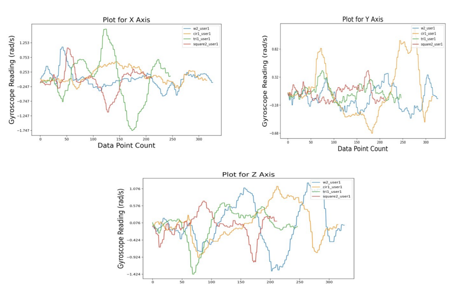

Fig. 14. Different Gestures, Same User (Gesture Comparison)

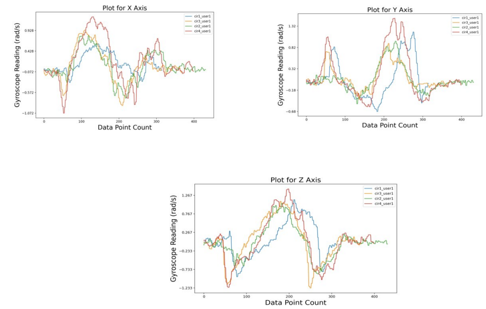

Fig. 15. Same Gestures, Same User (Gesture Comparison)

This model achieves **94%** accuracy (fig-16). Another model was trained to identify only the user id, not any particular cluster. With the similar method mentioned above, the model achieved similar accuracy of **94%** (fig-17). On the other hand, head origin based trained model achieved accuracy of **89%** and **78%** respectively to identify a user and identify cluster id. Details of this result can be found in appendix-A.

```
############ classification_report ####################
              precision    recall  f1-score   support

     class 0       1.00      1.00      1.00         2
     class 1       1.00      0.75      0.86         4
     class 2       0.67      1.00      0.80         2
     class 3       1.00      1.00      1.00         1
     class 4       1.00      1.00      1.00         1
     class 5       1.00      1.00      1.00         1
     class 6       1.00      1.00      1.00         3
     class 7       1.00      1.00      1.00         2
     class 8       1.00      1.00      1.00         1
     class 9       1.00      1.00      1.00         1

    accuracy                           0.94        18
   macro avg       0.97      0.97      0.97        18
weighted avg       0.96      0.94      0.95        18
```

Fig. 16. SVM Accuracy Report on AP Cluster Label [Gyroscope Data]

```
############ classification_report ####################
              precision    recall  f1-score   support

       user1       1.00      1.00      1.00        11
       user2       0.80      1.00      0.89         4
       user3       1.00      0.67      0.80         3

    accuracy                           0.94        18
   macro avg       0.93      0.89      0.90        18
weighted avg       0.96      0.94      0.94        18
```

Fig. 17. SVM Accuracy Report on User [Gyroscope Data]

V. Conclusion

This paper presents HoloAuth, a real-time sensor streaming application and user recognition system based on head movement for HoloLens2 device. Above analysis shows that head movement based gyroscope data alone can provide enough information to identify a user and also a particular head gesture performed by the user. However, a broad user study is required to confirm the accuracy. The paper demonstrated the efficacy of affinity propagation algorithm to effectively cluster user's head movement. One interesting observation was that when applying AP on multiple user's data, a particular cluster had data from only a single user. Using SVM model trained with gyroscope data, HoloAuth could recognize user and also particular gesture with **94%** accuracy. For future work, it will be great to explore association of eye tracking data with head tracking to improve the authentication system. It takes only 4-5 seconds to create useful head motion data using HoloAuth. The number of head gesture data used in the experiments is less, therefore, it will be interesting to find out the time requirement by SVM model for training and testing on large scale data. As a future work, it will be great to explore - if an attacker can evade this type of authentication systems, and if so, how can we build better defense.

REFERENCES

[1] S. Stephenson, B. Pal, S. Fan, E. Fernandes, Y. Zhao, and R. Chatterjee, "Sok: Authentication in augmented and virtual reality," in *2022 IEEE Symposium on Security and Privacy (SP)*, pp. 267–284, 2022.

[2] R. Düzgün, P. Mayer, and M. Volkamer, "Shoulder-surfing resistant authentication for augmented reality," in *Nordic Human-Computer Interaction Conference*, NordiCHI '22, (New York, NY, USA), Association for Computing Machinery, 2022.

[3] R. Zhang, N. Zhang, C. Du, W. Lou, Y. T. Hou, and Y. Kawamoto, "Augauth: Shoulder-surfing resistant authentication for augmented reality," *2017 IEEE International Conference on Communications (ICC)*, pp. 1–6, 2017.

[4] G. M. Garrido, V. Nair, and D. Song, "Sok: Data privacy in virtual reality," 2023.

[5] S. Yi, Z. Qin, E. Novak, Y. Yin, and Q. Li, "Glassgesture: Exploring head gesture interface of smart glasses," in *IEEE INFOCOM 2016 - The 35th Annual IEEE International Conference on Computer Communications*, pp. 1–9, 2016.

[6] C. Mukherjee, "HoloAuth Data Collector (https://github.com/chandms/HoloLens2-ResearchMode-Unity)," 4 2023.

[7] D. Ungureanu, F. Bogo, S. Galliani, P. Sama, X. Duan, C. Meekhof, J. St{"uhmer, T. J. Cashman, B. Tekin, J. L. Sch{"onberger, B. Tekin, P. Olszta, and M. Pollefeys, "HoloLens 2 Research Mode as a Tool for Computer Vision Research," *arXiv:2008.11239*, 2020.

[8] B. J. Frey and D. Dueck, "Clustering by passing messages between data points," *Science*, vol. 315, no. 5814, pp. 972–976, 2007.

[9] A. Akl and S. Valaee, "Accelerometer-based gesture recognition via dynamic-time warping, affinity propagation, compressive sensing," in *2010 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 2270–2273, 2010.

[10] M. Christ, N. Braun, J. Neuffer, and A. W. Kempa-Liehr, "Time series feature extraction on basis of scalable hypothesis tests (tsfresh – a python package)," *Neurocomputing*, vol. 307, pp. 72–77, 2018.

APPENDIX

For source code, please refer to this GitHub Repo. Some important steps that had to be followed to develop the HoloLens2 sensor streaming application are mentioned below.

Enable Windows Device Portal in HoloLens2 - Using Windows Device Portal (WDP), one can not only configure and manage the device, also, deploy and configure apps remotely. To enable WDP, follow the official document.

Enable Research Mode in HoloLens2 - By default, HoloLens2 does not allow access to raw data from Research Mode sensors to third party apps and built solely for research purpose. To enable Research Mode in HoloLens2, we need to complete the following steps -

- Open Windows Device Portal in your device(laptop), need to be within same network as of HoloLens2 device.
- Go to System > Research Mode
- Restart the device from power menu item at the top of the page.

These above steps can also be found in HoloLens2 website. To enable the app to capture data from restricted sensors, the developer needs to do the following -

- Build the project in unity
- Open *Package* APPXMANIFEST file, located under the <App Name> folder.
- Add *restrictedcapabilities* following the link.
- Add *perceptionSensorsExperimental* capability between *uap2:Capability* and *DeviceCapability*.
- If the app accesses data from IMU sensors, also add *backgroundSpatialPerception* to device capability.

Enable Eye Tracking in HoloLens2 - Official Document gives very detailed idea of enabling gaze input in HoloLens2 apps.

Affinity Propagation Result Based on Head Origin - Figure 18 shows similar clustering result as of based on gyroscope data. However, figure 19 shows that testing with different data set of same user sometimes get clustered with different type of head data, for example - 2^{nd}, 5^{th}, and 13^{th} rows demonstrate this mismatch. Figure 20 also shows similar kind of result as of figure 12. Each cluster only contains data from a particular user. In this particular experiment, head origin based data outperformed gyroscope based data in identifying correct user. However, gyroscope data could differentiate between shapes better than head origin data.

SVM Classification Accuracy Based on Head Origin - Output label as the AP cluster identifier, similar SVM model trained using head origin data provides accuracy of **78%** (fig - 21) which is way lesser than gyroscope based accuracy mentioned before. However, using head origin data for training, SVM model can still predict correct user **89%** (fig - 22) of the time. Clearly, the model outperformed when trained with gyroscope data instead of head origin data.

	True Label	Assigned Cluster	Assigned User	Assigned Shape
0	cir1_user1	class 0	user1	circle
1	cir3_user1	class 0	user1	circle
2	cir2_user1	class 0	user1	circle
3	cir4_user1	class 0	user1	circle
4	tri1_user1	class 1	user1	triangle
5	tri3_user1	class 1	user1	triangle
6	tri4_user1	class 1	user1	triangle
7	tri2_user1	class 1	user1	triangle
8	w2_user1	class 2	user1	w-shape
9	w4_user1	class 2	user1	w-shape
10	w3_user1	class 2	user1	w-shape
11	w1_user1	class 2	user1	w-shape
12	square2_user1	class 1	user1	triangle
13	square1_user1	class 1	user1	triangle

Fig. 18. AP Clustering on User1 [Head Origin Data]

	True Label	Assigned Cluster	Assigned User	Assigned Shape
0	tri4_user1_2	class 1	user1	triangle
1	w1_user1_2	class 2	user1	w-shape
2	cir4_user1_2	class 1	user1	triangle
3	tri1_user1_2	class 1	user1	triangle
4	cir1_user1_2	class 0	user1	circle
5	w4_user1_2	class 1	user1	triangle
6	square2_user1_2	class 1	user1	triangle
7	square1_user1_2	class 1	user1	triangle
8	cir2_user1_2	class 0	user1	circle
9	tri2_user1_2	class 1	user1	triangle
10	w2_user1_2	class 2	user1	w-shape
11	square4_user1_2	class 1	user1	triangle
12	square3_user1_2	class 1	user1	triangle
13	w3_user1_2	class 1	user1	triangle
14	tri3_user1_2	class 1	user1	triangle
15	cir3_user1_2	class 0	user1	circle

Fig. 19. AP Clustering Test Result on User1 with Different Dataset [Head Origin Data]

	True Label	Assigned Cluster	Assigned User	Assigned Shape
0	cir1_user1	class 0	user1	circle
1	cir1_user3	class 3	user3	triangle
2	cir1_user2	class 5	user2	w-shape
3	cir3_user1	class 0	user1	circle
4	cir3_user3	class 3	user3	triangle
5	cir2_user1	class 0	user1	circle
6	cir2_user2	class 1	user2	circle
7	cir2_user3	class 3	user3	triangle
8	cir4_user1	class 0	user1	circle
9	tri1_user3	class 3	user3	triangle
10	tri1_user2	class 1	user2	circle
11	tri1_user1	class 2	user1	triangle
12	tri3_user3	class 3	user3	triangle
13	tri3_user1	class 2	user1	triangle
14	tri4_user1	class 2	user1	triangle
15	tri2_user2	class 1	user2	circle
16	tri2_user3	class 3	user3	triangle
17	tri2_user1	class 2	user1	triangle
18	w2_user1	class 4	user1	w-shape
19	w2_user2	class 5	user2	w-shape
20	w2_user3	class 3	user3	triangle
21	w4_user1	class 4	user1	w-shape
22	w3_user1	class 4	user1	w-shape
23	w3_user3	class 3	user3	triangle
24	w1_user1	class 4	user1	w-shape
25	w1_user3	class 3	user3	triangle
26	w1_user2	class 5	user2	w-shape
27	square3_user3	class 3	user3	triangle
28	square2_user1	class 2	user1	triangle
29	square2_user2	class 1	user2	circle
30	square2_user3	class 3	user3	triangle
31	square1_user1	class 2	user1	triangle
32	square1_user3	class 3	user3	triangle
33	square1_user2	class 1	user2	circle

Fig. 20. AP Clustering on Data from Multiple Users [Head Origin Data]

```
############ classification_report ####################
              precision    recall  f1-score   support

     class 0       0.60      1.00      0.75         3
     class 1       0.00      0.00      0.00         3
     class 2       1.00      1.00      1.00         1
     class 3       1.00      1.00      1.00         2
     class 4       0.75      1.00      0.86         3
     class 5       1.00      1.00      1.00         1
     class 6       1.00      1.00      1.00         2
     class 7       1.00      0.67      0.80         3
     class 8       0.00      0.00      0.00         0

    accuracy                           0.78        18
   macro avg       0.71      0.74      0.71        18
weighted avg       0.73      0.78      0.73        18
```

Fig. 21. SVM Accuracy Report on AP Cluster Label [Head Origin Data]

```
############ classification_report ####################
              precision    recall  f1-score   support

       user1       0.91      0.91      0.91        11
       user2       1.00      0.75      0.86         4
       user3       0.75      1.00      0.86         3

    accuracy                           0.89        18
   macro avg       0.89      0.89      0.87        18
weighted avg       0.90      0.89      0.89        18
```

Fig. 22. SVM Accuracy Report on User [Head Origin Data]