

Privacy-Preserving Gesture Recognition with Explainable Type-2 Fuzzy Logic Based Systems

Josip Rožman, Hani Hagrass, Javier Andreu-Perez
The Computational Intelligence Centre, School of Computer Science
and Electronic Engineering
University of Essex, Colchester, UK

Damien Clarke, Beate Müller, Steve Fitz
Data Exploitation Group, Plextek
Plextek Ltd., Great Chesterford, UK

Abstract— Smart homes are a growing market in need of privacy preserving sensors paired with explainable, interpretable and reliable control systems. The recent boom in Artificial Intelligence (AI) has seen an ever-growing persistence to incorporate it in all spheres of human life including the household. This growth in AI has been met with reciprocal concern for the privacy impacts and reluctance to introduce sensors, such as cameras, into homes. This concern has led to research of sensors not traditionally found in households, mainly short range radar. There has been also increasing awareness of AI transparency and explainability. Traditional AI black box models are not trusted, despite boasting high accuracy scores, due to the inability to understand what the decisions were based on. Interval Type-2 Fuzzy Logic offers a powerful alternative, achieving close to black box levels of performance while remaining completely interpretable. This paper presents a privacy preserving short range radar sensor coupled with an Explainable AI system employing a Big Bang Big Crunch (BB-BC) Interval Type-2 Fuzzy Logic System (FLS) to classify gestures performed in an indoor environment.

Keywords— Type-2 Fuzzy Logic; Big Bang - Big Crunch; Privacy Preserving Sensing; Explainable Artificial Intelligence (XAI)

I. INTRODUCTION

With the growing interest in smart home applications there is an increasing demand for privacy preserving sensors and technologies [1], [2]. Current household appliances use a touch based interface which has an inherent flaw; the requirement of the individual to be in proximity of the device. This can be an issue when designing devices useable by all individuals. Hence, alternative communication channels have been explored including gesture recognition in particular [3], [4], [5], [6].

The majority of research conducted in the field of indoor gesture recognition is based on RGB camera systems. One such system can be found in [7] where a public video sequence dataset was used to train a model composed of a Deep Belief Network (DBN) and a 3D Convolutional Neural Network (CNN). The dataset used for training contained a total of 940 video sequences containing 10 to 20 sequences of each of the 20 gestures used. The results were evaluated using the Jaccard Index and a value of 80.9% was reported. The wide use of such camera based systems, despite boasting high classification

performance, is limited due to privacy issues concerning RGB cameras. A large majority of people would not feel comfortable with constant camera surveillance required for such systems to work. This is especially true in privacy-sensitive rooms of the house such as bathrooms and bedrooms. This has led to a surge in research based on privacy preserving sensors for gesture recognition.

An implementation of an Ultrasonic based gesture classification system can be found in [8]. The system was trained to recognize 6 small finger gestures at a short range. An additional class was used to represent no fingers being present. In the training process 2700 sample sequences of gestures and an additional 2700 of the no finger class were used. The final reported accuracy of the system is 96.34%. A high resolution and low power gesture sensing technology based on a millimeter-wave radar named Soli was presented by [9]. Short range temporal variation was observed and supplied to a random forest classifier to predict one of 4 gestures. An accuracy of 92.1% was reported for 1000 test gestures. As this work focused on short range gestures it is important to note the noise floor was greatly decreased. The Soli sensor was used by [10] to explore a wider range of gestures. A total of 11 gestures were observed and classified using a Convolutional Neural Network architecture. An accuracy of 88.0% was reported for a non-personalized classifier and a 94.5% for a personalized classifier.

All above mentioned systems use black box classification models which lack interpretability, and inference cannot be augmented by expert knowledge or be assessed by third parties, based on the XAI's principles [11], [12].

This paper proposes the use of a Texas Instruments (TI) IWR6843ISK-ODS short range radar paired with a fully explainable BB-BC type-2 Fuzzy Logic classification system. The use of radar based technology has multiple advantages over cameras and similar sensors. Data gathered by radar are anonymous, containing only range, radial velocity and angle values. An additional advantage of radars is its insensitivity to lighting and obscurants as opposed to cameras. On the other hand fuzzy logic offers a powerful tool to deal with high noise to signal ratio and ambiguity, while offering an explainable alternative to black box models [13]. The proposed system performance will be compared to a Type-1 Fuzzy logic system, a Deep Neural Network and a Convolutional Neural Network.

Section II details the Data collection and processing steps. Section III presents a brief overview on Interval Type-2 Fuzzy Logic. Section IV presents the proposed BB-BC type-2 FLS classification system. The experiments and results are presented in Section V. Conclusions and future work are presented in Section VI.

II. DATA COLLECTION AND FEATURE EXTRACTION

A. Data collection

For the collection of data, a Texas Instruments (TI) IWR6843ISK-ODS board (depicted in Fig.1) was used. The board contains 3 millimeter-wave radar transmitters, 4 receivers and operates in a frequency range of 60.25 to 64 GHz.



Fig. 1. TI IWR6843ISK-ODS board used for data collection.

A high velocity resolution of 0.15 km/h was used along with a preset maximum range of 5 m and a range resolution of 10 cm. The sampling frequency was set to 10 samples per second along with a video recording of the same scene with a sampling frequency of 30 frames per second.

The proposed radar is capable of measuring range and radial velocity along with azimuth and elevation angles of objects within the field of view [14]. At each time step the radar measures the amplitude for each combination of range and radial velocity for every transmitter and receiver antenna pair. A positive and negative value of radial velocity can be observed for objects moving away from and towards the sensor respectively. The radial velocity shall be referred to as the Doppler value. An example of a Range-Doppler Amplitude heatmap averaged across all antenna pairs for a bicycle like motion with hands is shown in Fig. 2.

A large amount of reflected signal can be observed at longer ranges. The closest range signal is caused by the direct reflection whereas the longer range signals are caused by multi-path reflections (e.g. from the ceiling and floor). To address this issue the highest peak in the zero Doppler bin was selected as it is assumed this represents the reflection from the static body of the subject. All range bins prior to the peak (range bins closer to the sensor) are retained while all range bins behind the peak value have been discarded.

All non-zero amplitude points in the heatmap are then transformed into a point cloud and the variation in antenna pair measurements for each Range-Doppler combination is used to infer 3D coordinates. It was noticed that spread along the elevation axis was inconsistent and the reason was found to be related to low Doppler return values and weak returns. The removal of the previously stated values was then applied to mitigate the issue. Fig. 3 shows a filtered and unfiltered windowed Range-Doppler heat map while Fig. 4 represents the resulting 3D point cloud.

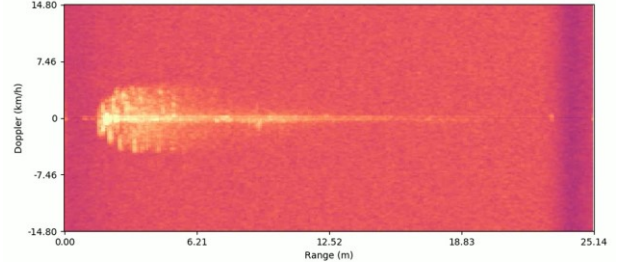


Fig. 2. Range-Doppler-Amplitude heatmap of the incoherent sum of antenna pairs.

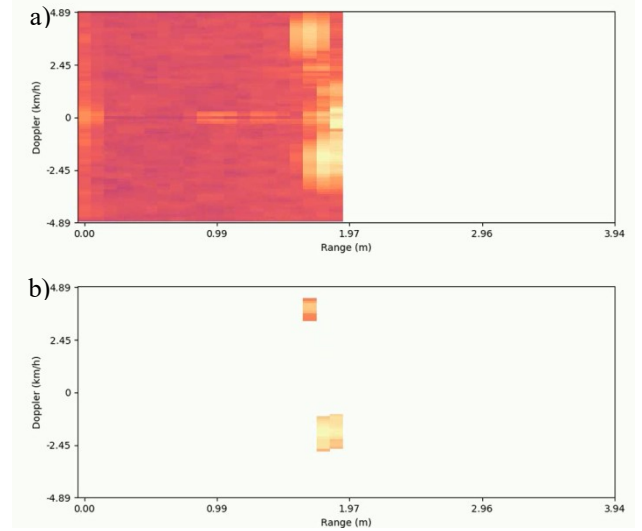


Fig. 3. a) unfiltered Range-Doppler Amplitude heatmap b) filtered Range-Doppler Amplitude heatmap

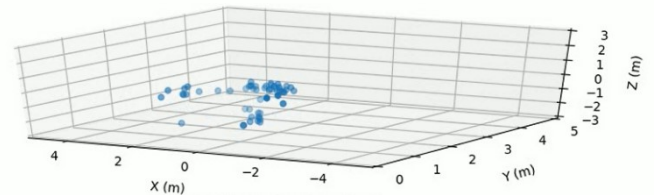


Fig. 4. Example of a 3D Point cloud representation of data in a single time instance.

B. Feature extraction

A total of 9 features were extracted from the point cloud data at each time step and time windowed for a total of one

second resulting in **90** features (inputs to the classification system). The features used were: change in x, y and z central coordinate from the previous frame, average radial velocity across all points, average acceleration across all points, volume of a cuboid fit to incorporate all points, average distance between points, squared average velocity across all points and the number of clusters.

The number of clusters was extracted using a DBSCAN algorithm with a 0.5 threshold, a minimum number of 10 samples and the use of the Euclidian distance as the metric.

III. A BRIEF OVERVIEW ON INTERVAL TYPE-2 FUZZY LOGIC

Fuzzy logic provides a powerful tool to deal with real world uncertainty and provides a mathematical model to handle this uncertainty [15], [16], [17], [18], [19], [20], [21], [22]. As described in [23] various types of uncertainty can be observed:

- Uncertainty about the meaning of the words that are used in the rules
- Uncertainty about the consequent that is used in a rule
- Uncertainty about the measurements that activate the fuzzy system
- Uncertainty about the data that are used to tune parameters of a fuzzy system

Type-2 Fuzzy Logic allows modeling of a higher degree of uncertainty, which is of importance when using noisy sensor data. A type-2 fuzzy set can be defined as [23]:

$$\tilde{A} = \{(x, u), \mu_{\tilde{A}}(x, u) \mid \forall x \in X, \forall u \in J_x \subseteq [0, 1]\} \quad (1)$$

$x \in X$ is a value within the universe of the primary variable, $u \in U \equiv [0, 1]$ is value within universe of the secondary variable, $J_x \subseteq [0, 1]$ represents the primary membership of x . $\mu_{\tilde{A}}(x, u)$ is a type-1 fuzzy set known as the secondary set. A type-2 fuzzy set \tilde{A} is completely determined by its footprint of uncertainty (FOU) (shown cross hatched in Fig. 5). The FOU can be properly defined by its lower and upper membership functions [23].

$$\bar{\mu}_{\tilde{A}}(x) = \sup\{u \mid u \in [0, 1], \mu_{\tilde{A}}(x, u) > 0\} \quad (2)$$

$$\underline{\mu}_{\tilde{A}}(x) = \inf\{u \mid u \in [0, 1], \mu_{\tilde{A}}(x, u) > 0\} \quad (3)$$

In this paper, we will employ the interval Type-2 Fuzzy system which assumes all secondary membership grades of the type-2 sets are set to 1, that is $\mu_{\tilde{A}}(x, u) = 1, \forall x \in J_x \subseteq [0, 1]$. An example of an Interval type-2 Membership Function is given in Fig. 5.

IV. THE PROPOSED BB-BC TYPE-2 FLS CLASSIFIER SYSTEM

Fig. 6 depicts the steps involved in the development of BB-BC type-2 FLS classifier system which starts by data collection followed by random rule generation and then followed by raw

rule reduction and rule length reduction while employ the BB-BC to maximise the model accuracy and interpretability (by generating small rule bases with short rules). The following subsections will discuss these steps in detail.

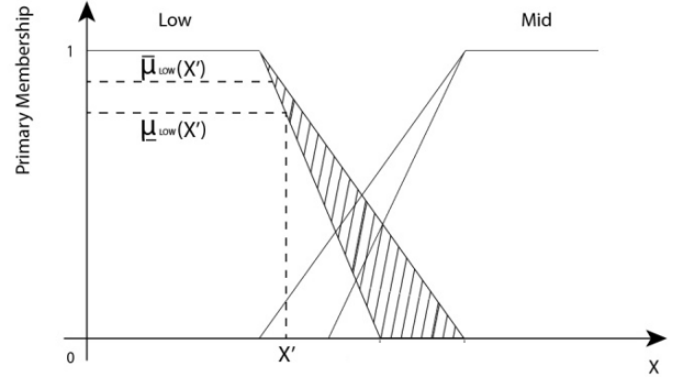


Fig. 5. Type-2 Interval Type-2 Membership Function.

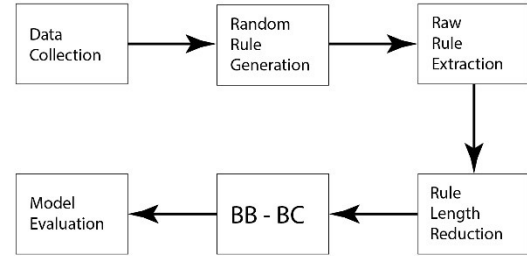


Fig. 6. Flow chart of proposed optimization steps for the type-2 Fuzzy Logic classifier

A. Initial rule base creation

Antecedent membership functions are generated by creating sets of 3 piecewise linear membership functions corresponding to Low, Medium and High for each input as shown in Fig. 7.

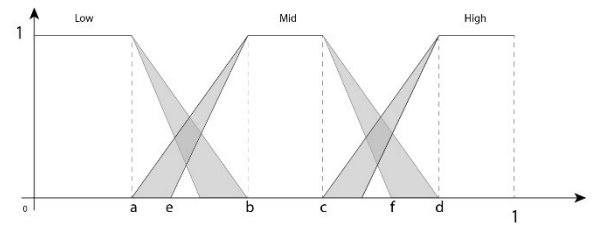


Fig. 7. Type-2 Membership Functions generated for each input.

This is done by randomly generating sets of three numbers for each membership function. The first number represents the end of the rising edge (b); the second number represents the start of the falling edge (c), while the final number represents the relative size of the FOU (e and f). The start of the rising edge of each membership function is anchored at the start of the falling edge of the previous membership function. The same logic is applied to the end of the falling edge and the end

of the rising edge of the next membership function. This is done to ensure overlap of membership functions and prevent segments in which the membership grade of all membership function would amount to 0. The relative footprint of uncertainty value represents the displacement between the start of the rising edge of the lower membership function as opposed to the upper membership function. The same logic is applied to the falling edge, while the end of the rising edge and start of the falling edge are anchored in the same points as their upper membership function counterparts.

B. Raw rule extraction

To assign valid consequents for each rule a procedure described in [24] was implement. For each input vector in the dataset, the membership grades of the upper and lower membership functions were calculated and all combinations where the values of the membership grades were nonzero were extracted. For each of the given combinations a consequent membership function was assigned containing the true value for the given input. Scaled firing strengths are then calculated for each rule using the given equation :

$$\overline{f^{jt}}(x^t) = \prod_1^n \overline{\mu_{A_n^{ajt}}}(x_n) \quad (4)$$

$$\underline{f^{jt}}(x^t) = \prod_1^n \underline{\mu_{A_n^{ajt}}}(x_n) \quad (5)$$

Where $\overline{f^{jt}}(x^t)$ and $\underline{f^{jt}}(x^t)$ represent the upper and lower bounds of the firing strengths for each of the t data points.

The given procedure results in a large number of rules, some of which can have the same antecedent but a different consequent. In order to calculate the weighted scale dominance the firing strengths of all rules have to be normalized. This is done by dividing the firing strength of the given rule by the summation of all the firing strengths of rules sharing the same consequent [24].

$$\overline{f_s^{jt}} = \frac{\overline{f^{jt}}}{\sum_{j \in \text{Class } j} \overline{f^{jt}}} \quad (6)$$

$$\underline{f_s^{jt}} = \frac{\underline{f^{jt}}}{\sum_{j \in \text{Class } j} \underline{f^{jt}}} \quad (7)$$

Where $\overline{f_s^{jt}}$ and $\underline{f_s^{jt}}$ are the upper and lower scaled bounds of the firing strength of rules sharing the same consequent.

To resolve the conflicting rules containing the same antecedent and different consequent scaled confidence and scaled support are calculated as [25]:

The scaled confidence is computed as follows:

$$\overline{\text{conf}}(\tilde{A}_q \Rightarrow C_q) = \frac{\sum_{x_s \in C_q} \overline{f_s^{jt}}(x_s)}{\sum_{j=1}^m \overline{f_s^{jt}}(x_s)} \quad (8)$$

$$\underline{\text{conf}}(\tilde{A}_q \Rightarrow C_q) = \frac{\sum_{x_s \in C_q} \underline{f_s^{jt}}(x_s)}{\sum_{j=1}^m \underline{f_s^{jt}}(x_s)} \quad (9)$$

Where $\overline{\text{conf}}(\tilde{A}_q \Rightarrow C_q)$ and $\underline{\text{conf}}(\tilde{A}_q \Rightarrow C_q)$ are the upper and lower scale confidence bounds of the class C_q being the consequent class given the antecedents \tilde{A}_q for m conflicting rules.

The scaled support is computed as follows:

$$\overline{\text{supp}}(\tilde{A}_q \Rightarrow C_q) = \frac{\sum_{x_s \in C_q} \overline{f_s^{jt}}(x_s)}{m} \quad (10)$$

$$\underline{\text{supp}}(\tilde{A}_q \Rightarrow C_q) = \frac{\sum_{x_s \in C_q} \underline{f_s^{jt}}(x_s)}{m} \quad (11)$$

Where the scaled support of a rule having the consequent C_q given the antecedents \tilde{A}_q can be defined by its upper and lower bounds $\overline{\text{supp}}(\tilde{A}_q \Rightarrow C_q)$ and $\underline{\text{supp}}(\tilde{A}_q \Rightarrow C_q)$. m is the number of training patterns.

The scaled dominance is then defined as:

$$\overline{d}(\tilde{A}_q \Rightarrow C_q) = \overline{\text{conf}}(\tilde{A}_q \Rightarrow C_q) * \overline{\text{supp}}(\tilde{A}_q \Rightarrow C_q) \quad (12)$$

$$\underline{d}(\tilde{A}_q \Rightarrow C_q) = \underline{\text{conf}}(\tilde{A}_q \Rightarrow C_q) * \underline{\text{supp}}(\tilde{A}_q \Rightarrow C_q) \quad (13)$$

Where $\overline{d}(\tilde{A}_q \Rightarrow C_q)$ and $\underline{d}(\tilde{A}_q \Rightarrow C_q)$ represent the upper and lower bound of scaled dominance of a rule belonging to class C_q given the antecedents \tilde{A}_q .

To calculate the weighted scaled dominance the average dominance over rules with same antecedents and different consequents is subtracted from the dominance values calculated in the previous step [24].

$$\overline{wd}(\tilde{A}_q \Rightarrow C_q) = \overline{d}(\tilde{A}_q \Rightarrow C_q) - \overline{d_{ave}} \quad (14)$$

$$\underline{wd}(\tilde{A}_q \Rightarrow C_q) = \underline{d}(\tilde{A}_q \Rightarrow C_q) - \underline{d_{ave}} \quad (15)$$

Where $\overline{wd}(\tilde{A}_q \Rightarrow C_q)$ and $\underline{wd}(\tilde{A}_q \Rightarrow C_q)$ are the upper and lower bounds of the weighted scaled dominance of a rule belonging to class C_q given the antecedents \tilde{A}_q . Where $\overline{d_{ave}}$ and $\underline{d_{ave}}$ are upper and lower bounds of the average dominance over rules with the same antecedents \tilde{A}_q , and a conflicting consequent class.

The number of rules generated by this procedure is large so for each outcome only **30 rules** with the highest weighted dominance are retained. By performing this step, some combination of inputs will not be represented in the rule base. Classification is performed in one of two ways dependent on if the current combination of inputs is represented in any rule. If the input matches one of the existing rules the final class can be obtained by taking the maximum value of the $Z\text{Class}_h$ [24].

$$\overline{Z\text{Class}}_h(x^t) = \frac{\sum_{j \in h} \overline{f^{jt}}(x^t) * \overline{wd}(\tilde{A}_q \Rightarrow C_q)}{\max_{j \in h} (\overline{f^{jt}}(x^t) * \overline{wd}(\tilde{A}_q \Rightarrow C_q))} \quad (16)$$

$$\underline{ZClass}_h(x^t) = \frac{\sum_{j \in h} f^j x^t * \overline{wd}(\bar{A}_q \Rightarrow C_q)}{\max_{j \in h} (f^j(x^t) * \overline{wd}(\bar{A}_q \Rightarrow C_q))} \quad (17)$$

$$ZClass_h = \frac{\overline{ZClass}_h(x^t) + \underline{ZClass}_h(x^t)}{2} \quad (18)$$

Where $ZClass_h$ is the total vote strength for a class given an input vector $x^{(p)}$ calculated as the mean of its upper and lower bounds \overline{ZClass}_h and \underline{ZClass}_h .

In the event the input does not match any of the rules in the rule base a similarity measure is calculated using the distances between the linguistic labels of the input fired and those in each rule of the rule base. The most similar rules dominance and the firing strength of the current combination of inputs shall be used in the previously stated equations to determine the output [24].

A subset of 30 rules representing each of the possible outputs was kept by selecting only the rules containing the highest weighted dominance values.

C. Rule length reduction

In order to increase the readability and interpretability of the final output a basic genetic algorithm without crossover was implemented to generate sets of 4 indices representing the inputs to use for each of the rules. The fitness of each chromosome was determined by evaluating the F1 score on the training set. The sample with the highest fitness score in the initial population is then selected and mutated to produce the next generation. This procedure is performed for a set number of generations.

The F1 score was chosen as the metric of choice as it incorporates both precision and recall and is defined as:

$$F1 = \frac{2 * precision * recall}{precision + recall} \quad (19)$$

D. Big Bang-Big Crunch antecedent optimisation

Big Bang-Big Crunch is an algorithm inspired by the creation of the universe [26, 27]. The algorithm is based on two stages, the first being the Big Bang stage which introduces disorder while in the Big Crunch stage randomly distributed particles are pulled into order [27]. Initial points are distributed randomly and then shrunk into a single point being the center of mass. At each Big Bang stage a smaller segment around the centre of mass is used to generate new points [28, 29].

$$x_{new} = x_c + l r / k \quad (20)$$

Where x_c represent the current centre of mass, l the upper limit, r a random value uniformly distributed over the interval $[0, 1)$ and k the current iteration index.

The chromosomes generated by the Big Bang-Big Crunch algorithm consisted of three parameters (b, c, p). The fitness function maximized by the algorithm was the F1 score achieved on the training dataset. It is important to note that p is the relative size of the footprint of uncertainty and is used to encode both the e and f parameters. The values of e and f can be calculated as:

$$e = a + p * (b - a) \quad (21)$$

$$f = d - p * (d - c) \quad (22)$$

V. EXPERIMENTS AND RESULTS

A. Dataset preparation

The dataset used consists of multiple samples of 6 individuals performing 5 different motions as shown in Fig 8. The motions observed were:

- Single hand swipe movement (left or right)
- Up down movement
- A bicycle like motion towards the radar with both arms (opposite circular)
- Circular movement with both arms tangential to the radar
- Linear forward and backward movement with both hands

Motions used can easily be mapped to a traditional control system in a domestic space where:

- Circular motion – Scroll over potential devices
- Hand swipe – Scroll operation on device
- Opposite circular - run current device
- Up down – stop current device
- Forward back – activate currently selected feature of device (button press)

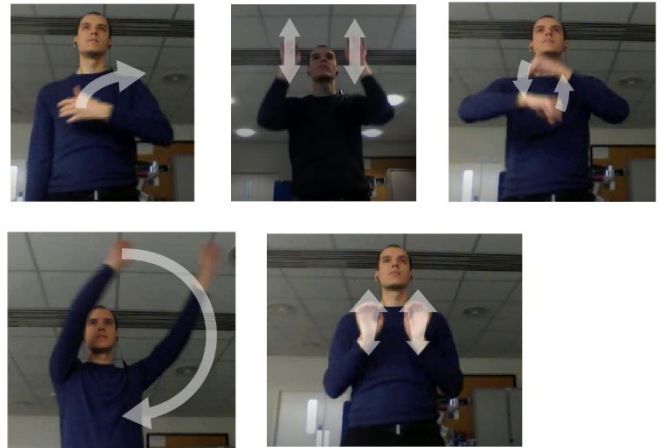


Fig. 8. Motions used starting from the top left a) Hand swipe, b) Up down motion, c) Opposite circular motion, d) Circular motion, e) Forward backward motion

Training was performed on 50 files (10 of each motion) containing continuous repetition of the given motion for 17 to 26 seconds of data. Upon performing the previously described feature extraction steps the resulting dataset contains 8821 samples with 90 features. A separate validation set was provided containing a total of 15 files (3 for each motion)

resulting in 2995 samples and the models were evaluated on 5 files (one of each motion type).

A Type-1 Fuzzy Logic classifier, Type-2 Fuzzy Logic classifier, a Deep Neural Network (DNN), and a Convolutional Neural Network based classifier were trained and evaluated.

The Deep Neural Network used consists of 5 fully connected layers using a Rectified Linear Unit (ReLU) activation function in all hidden layers, while the output layer uses a softmax activation function. The proposed CNN consists of two blocks of two 1D convolutional layers followed by batch normalization and max pooling. Another two 1D convolutional layers are then applied and followed by a batch normalization layer and Global Average Pooling layer, which feeds into a fully connected layer using a softmax activation function to produce the final outputs.

B. Results

Offline evaluation of the system was performed to produce a confusion matrix as well as visual evaluation by tagging a video with the predictions. The final outputs were produced by performing a classification for 5 consecutive time steps and then choosing the class which was predicted the most times in that interval. In the event more than a single class had the maximum number of predictions, the class with the highest absolute weighted scale dominance was chosen. The confusion matrix for the proposed Type-2 Fuzzy Logic System can be found below (Fig. 9) followed by a confusion matrix for the proposed CNN (Fig. 10).

		PREDICTED				
		Circular	Forward_back	Swipe	Opposite_Circular	Up_Down
T	Circular	27	2	10	2	12
R	Forward_back	0	35	2	17	0
U	Swipe	10	11	27	1	5
E	Opposite_Circular	0	0	2	50	0
	Up_down	11	0	0	6	36

Fig. 9. Confusion matrix for the proposed Type-2 Fuzzy Logic classifier on unseen data

		PREDICTED				
		Circular	Forward_back	Swipe	Opposite_Circular	Up_Down
T	Circular	45	0	1	1	6
R	Forward_back	0	53	1	0	0
U	Swipe	3	0	42	9	0
E	Opposite_Circular	0	0	0	52	0
	Up_down	11	0	0	0	42

Fig. 10. Confusion matrix for proposed CNN classifier on unseen data

Table I shows on the testing unseen data, the performances achieved by a the proposed Type 2 Fuzzy Logic System, a Convolutional Neural Network, a stacked autoencoder Deep Neural Network, and a Type-1 Fuzzy Logic System.

TABLE I. CLASSIFIER PERFORMANCE ON UNSEEN DATA

	Precision	Recall	F1-Score	Accuracy
T2 FLS	65.96%	65.75%	65.86%	65.79%
T1 FLS	31.33%	40.85%	35.46%	31.58%
CNN	88.02%	88.62%	88.32%	87.97%
DNN	73.82%	74.15%	73.98%	73.78%

The results show the black box Deep Neural Network model achieves only 8.3% higher average recall as opposed to the proposed system. Both systems have fallen short to a more complex and specialized Convolutional Neural Network. The poor performance of the type-1 Fuzzy Logic system could be attributed to the high noise and uncertainty in the dataset for which the proposed system is better suited.

Fig. 11 shows the video used to visually evaluate the results. In the top left the predicted label is shown, while the lower part of the image contains a bar chart which displays the percentage of the motion belonging to each class.

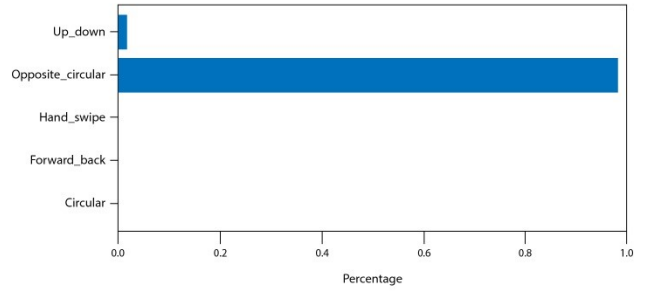


Fig. 11. Frame of a video used for visual evaluation and tagging of motion on a video sequence

While the other black box models lack explainability an additional emphasis has been placed on making the type-2 Fuzzy Logic model explainable. To this end the rule length has been limited to a maximum of 4 antecedents per rule (while the system has 90 inputs to the FLS) while retaining only 30 rules per class. Table II contains an example of a rule used for up down movement classification.

TABLE II. EXAMPLE OF A RULE GENERATED BY THE UP DOWN FUZZY LOGIC CLASSIFIER

Ante. 1	Ante. 2	Ante. 3	Ante. 4	Output
Med_7	Med_1	Low_8	Med_3	Circular Motion

With prior knowledge of the feature space this rule can be read as follows: IF the change of z coordinate in time step 7 is

Moderate, and *change of z coordinate in time step 1* is Moderate, and *average distance between points in time step 8* is Low, and *change of y coordinate in time step 3* is Moderate, then predict the output class as **UP-Down Motion**. It can be observed as the movement is vertical the classifier retained features which are related to a change in the elevation of the target.

An added benefit of the explainability is the ease of integrating expert knowledge into the system in terms of custom rules to augment the performance of the system. An example of such rule for forward-backward motion can be found in Table III.

TABLE III. EXAMPLE OF AN EXPERT RULE

Ante. 1	Ante. 2	Ante. 3	Ante. 4	Output
Low_1	High_2	High_3	High_4	Forward Backward

The interpretation of this rule would be: If *the change in the y coordinate in time step 1* is Low, and *y coordinate change in time steps 2, 3, 4* is High, the motion is **Forward Backward**. This could be interpreted as the arms have been stationary and have moved rapidly in consecutive measurements towards the sensor.

It is also important to note that the time complexity of training the Fuzzy Logic classifier model is greater than the time complexity of training the ANN models. However, this does not influence the inference time of the final model.

VI. CONCLUSIONS AND FUTURE WORK

In this paper we have presented a privacy preserving short range radar sensor coupled with an Explainable AI system employing a Big Bang Big Crunch (BB-BC) Interval Type-2 Fuzzy Logic System (FLS) to classify gestures performed in an indoor environment. Data has been collected using a TI IWR6843ISK-ODS board and used to optimize a type-2 Fuzzy Logic system for gesture recognition. The result show that the type-2 FLS results are comparable to black box CNN and stacked autoencoder deep neural networks while producing an Explainable AI (XAI) systems which could be easily read, analysed and augmented by the engineering team (to allow the verification of the control system) and the end user. The system allows for easy embedding of expert knowledge by allowing the user to modify or add custom rules. A significant performance increase compared to a type-1 Fuzzy Logic system can be observed. This is most likely due to the type-2 Fuzzy Logic systems enhanced performance in noisy environments. In future work improvements shall be made to the classification performance of the system in more challenging domestic and outdoor spaces.

VII. REFERENCES

- [1] A. ElShafee and K. A. Hamed, "Design and implementation of a WIFI based home automation system.", World academy of science, engineering and technology, vol. 68, pp. 2177-2180, 2012.
- [2] E. Horvitz and D. Mulligan, "Data, privacy, and the greater good", Science, vol. 349, no. 6245, pp. 253-255, 2015.
- [3] S. Mitra and T. Acharya, "Gesture Recognition: A Survey", IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews), vol. 37, no. 3, pp. 311-324, 2007.
- [4] R. Vatavu, "User-defined gestures for free-hand TV control", Proceedings of the 10th European conference on Interactive tv and video - EuroITV '12, 2012.
- [5] J. Daugman, "Face and gesture recognition: overview", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 19, no. 7, pp. 675-676, 1997.
- [6] S. Yang, P. Premaratne and P. Vial, "Hand gesture recognition: An overview", 2013 5th IEEE International Conference on Broadband Network & Multimedia Technology, 2013.
- [7] D. Wu et al., "Deep Dynamic Neural Networks for Multimodal Gesture Segmentation and Recognition", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 38, no. 8, pp. 1583-1597, 2016.
- [8] Y. Sang, L. Shi and Y. Liu, "Micro Hand Gesture Recognition System Using Ultrasonic Active Sensing", IEEE Access, vol. 6, pp. 49339-49347, 2018.
- [9] J. Lien et al., "Soli", ACM Transactions on Graphics, vol. 35, no. 4, pp. 1-19, 2016.
- [10] S. Wang, J. Song, J. Lien, I. Poupyrev and O. Hilliges, "Interacting with Soli", Proceedings of the 29th Annual Symposium on User Interface Software and Technology - UIST '16, 2016.
- [11] L. Gilpin, D. Bau, B. Yuan, A. Bajwa, M. Specter and L. Kagal, "Explaining Explanations: An Overview of Interpretability of Machine Learning", 2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA), 2018.
- [12] A. Preece, "Asking 'Why' in AI: Explainability of intelligent systems - perspectives and challenges", Intelligent Systems in Accounting, Finance and Management, vol. 25, no. 2, pp. 63-72, 2018.
- [13] W. Wei and J. Mendel, "A fuzzy logic method for modulation classification in nonideal environments", IEEE Transactions on Fuzzy Systems, vol. 7, no. 3, pp. 333-344, 1999.
- [14] M. Richards, Fundamentals of radar signal processing. New York [etc.]: McGraw-Hill Education, 2014.
- [15] A. Starkey, H. Hagaras, S. Shakya, G. Owusu, "A Multi-Objective Genetic Type-2 Fuzzy Logic Based System for Mobile Field Workforce Area Optimization" Information Sciences, Vol. 333, pp. 390-411, September 2016.
- [16] H. Hagaras, C. Wagner, "Introduction to Interval Type-2 Fuzzy Logic Controllers - Towards Better Uncertainty Handling in Real World Applications", The IEEE Systems, an and Cybernetics eNewsletter, Issue 27, June 2009.
- [17] H. Hagaras, M. Colley, V. Callaghan, M. Carr-West, "Online Learning and Adaptation of Autonomous Mobile Robots for Sustainable Agriculture", Autonomous Robots, Vol. 13, pp. 37-52, July 2002.
- [18] A. Antonelli, D. Bernardo, H. Hagaras, F. Marcelloni, "Multi-Objective Evolutionary Optimization of Type-2 Fuzzy Rule-based Systems for Financial Data Classification" IEEE Transactions on Fuzzy Systems, Vol.25, No.4, pp. 993-1005, August 2017
- [19] A. Sakalli, T. Kumbasar, E. Yesil, H. Hagaras, "Analysis of the performances of type-1, self-tuning type-1 and interval type-2 fuzzy PID controllers on the Magnetic Levitation system". Proceedings of the 2014 IEEE International Conference on Fuzzy Systems, Beijing, July 2014
- [20] J. Andreu-Perez, F.Cao, H. Hagaras, G. Yang, "A self-adaptive online brain-machine interface of a humanoid robot through a general type-2 fuzzy inference system", IEEE Transactions on Fuzzy Systems, Vol. 26, No.1, pp. 101-116, February 2018
- [21] C. Lynch, H. Hagaras, V. Callaghan, "Embedded Interval Type-2 Neuro-Fuzzy Speed Controller for Marine Diesel Engines", Proceedings of the International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU 2006), pp. 1340-1347, Paris, France, July 2006.
- [22] H. Hagaras, V. Callaghan, M. Colley, M. Carr-West "A Fuzzy-Genetic Based Embedded-Agent Approach to Learning and Control in Agricultural Autonomous Vehicles", Proceedings of the 1999 IEEE

- International Conference on Robotics and Automation, pp. 1005-1010, Detroit, U.S.A, May 1999
- [23] J. Mendel, Uncertain Rule-Based Fuzzy Logic Systems. Springer International Publishing AG, 2017.
 - [24] D. Bernardo, H. Hagraş and E. Tsang, "A genetic type-2 fuzzy logic based system for the generation of summarised linguistic predictive models for financial applications", *Soft Computing*, vol. 17, no. 12, pp. 2185-2201, 2013.
 - [25] H. Ishibuchi and T. Yamamoto, "Fuzzy rule selection by multi-objective genetic local search algorithms and rule evaluation measures in data mining", *Fuzzy Sets and Systems*, vol. 141, no. 1, pp. 59-88, 2004.
 - [26] B. Yao, H. Hagraş, D. Alghazzawi and M. Alhaddad, "A Big Bang–Big Crunch Type-2 Fuzzy Logic System for Machine-Vision-Based Event Detection and Summarization in Real-World Ambient-Assisted Living", *IEEE Transactions on Fuzzy Systems*, vol. 24, no. 6, pp. 1307-1319, 2016.
 - [27] O. Erol and I. Eksin, "A new optimization method: Big Bang–Big Crunch", *Advances in Engineering Software*, vol. 37, no. 2, pp. 106-111, 2006.
 - [28] R. Chimatapu, H. Hagraş, A. Starkey and G. Owusu, "Stacked Auto Encoder Based Hybrid Genetic Algorithm for Workforce Optimization", 2018 10th Computer Science and Electronic Engineering (CEECE), 2018.
 - [29] R. Chimatapu, H. Hagraş, A. Starkey and G. Owusu, "A Big-Bang Big-Crunch Type-2 Fuzzy Logic System for Generating Interpretable Models in Workforce Optimization", 2018 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), 2018.