Radar-based Noncontact Human Activity Classification Using Genetic Programming

Julio J. Valdés[1], Zachary Baird[2], Sreeraman Rajan[3], and Miodrag Bolic[4]

[1]Digital Technologies Research Centre, National Research Council Canada, Ottawa, Canada
email: julio.valdes@nrc-cnrc.gc.ca
[2]Department of Systems and Computer Engineering, Carleton University, Ottawa, Canada
email: zachbaird@cmail.carleton.ca
[3]Department of Systems and Computer Engineering, Carleton University, Ottawa, Canada
email: sreeraman.rajan@carleton.ca
[4]School of Electrical Engineering and Computer Science, University of Ottawa, Ottawa, Canada
email: miodrag.bolic@uottawa.ca

***Abstract*—This paper analyzes the structure of the feature space of radar data collected from real subjects either still or in motion and provides dimensionality reduction and modeling through genetic programming. Three movement classes: sedentary and still, sedentary with movements, and walking contained in the returns obtained from a single channel continuous wave phase-modulated radar are considered. Unsupervised methods are used for finding the intrinsic dimensionality of the space of the original features and nonlinear mappings are used to obtain lower dimensional representations. The classification results for the original and the reduced dimension data are similar, thus, indicating the redundancy of the eliminated features. The white-box models obtained through genetic programming is then compared with the conventional black-box models obtained through supervised classification using random trees, extreme learning machines and multilayer perceptron. For this problem, the explicit white-box models obtained with genetic programming produced equal or better classification accuracies than those obtained with black-box approaches. In addition to explainability, the genetic programming models found have the additional advantage of involving only a few relevant predictors, exhibiting good feature selection capabilities.**

I. Introduction

RADAR has long been exploited for contact-less sensing applications and recently for vital sign monitoring. Radar offers many benefits over other vital sign sensing techniques in that it is contact-less, it can operate in all ambient lighting conditions, it can sense through barriers, it uses little power and does not infringe upon the privacy of users.

Many studies in literature present signal processing algorithms for extraction of vital sign signals from raw radar returns. Majority of such studies deal only with data collected with stationary, or sedentary, subjects. As radar is capable of detecting even small movements, any movement by the subject will affect the extraction and estimation of the vital [1]. A preprocessing stage to identify the presence of movement was introduced in [2], [3], [19]. In the case that a sedentary subject is identified, signal processing algorithms are applied to estimate vital signs. When a moving subject is identified, either the processing is halted or a different kind of signal processing algorithms are utilized to estimate the vital signs. In [3], a supervised approach to classification of subject activity is presented where different machine learning models and feature subspace mappings were explored. In [19] the data structure was analyzed visually and the intrinsic dimensionality of the non-linear manifold was determined. In [20] a genetic algorithm approach was used to identify an optimal subset of features for classifying activities.

In this paper, the previously mentioned approaches are compared and a new approach to human activity classification using genetic programming (GP) is proposed. The proposed model is a white box model which is an alternative to the conventional black box model. Our proposed model produces an explicit (algebraic) representation of the dependencies between the predictor and target variables. Further, such models pave way for further analysis of the phenomenology of the subject motion in CW radar returns.

The rest of the paper is organized as follows: Section II describes the methodology of the work including data collection, preprocessing and feature extraction. Section III presents the concept of intrinsic dimensionality and selected estimation methods. Section IV describes the Gamma test (a residual variance analysis technique), used in combination with a genetic algorithm approach to feature selection. In Section V the classifier models used in this paper are introduced. Section VI describes the experimental settings associated to the different techniques used in the paper. Section VII presents the results from this work. Finally, Section VIII ends the paper with conclusions and final remarks.

II. Methodology and Materials

A. Radar

The radar used to collect data for this paper was a custom built (K&G Spectrum, Quebec) continuous wave (CW) radar. Separate transmit and receive antennas were used and mounted in very close proximity to one another, therefore the radar was

978-1-7281-6929-3/20/$31.00 ©2020 Crown

operating in pseudo-monostatic (independent transmitter and receiver placed at a distance much smaller than the distance to the target) configuration. The antennas had an aperture of 20^o horizontal by 70^o vertical. The radar operated with a single channel carrier frequency of 24.125 GHz. The sampling rate of the radar was 905 samples/second. Binary phase coding of the radar signal was used for non-ambiguous range resolution. The approximate range resolution of the radar returns was 0.75 m with some overlap. The number of range bins is configurable, for this paper 9 range bins were collected. The radar streamed baseband data to a nearby computer running proprietary software via Ethernet. Data was saved to comma separated variable files for later processing.

B. Data Collection

All experiments referred to in this paper were approved by Carleton University's Research Ethics Board and were part of a project to estimate breathing rate while the subjects were conducting levels of activities. Data collection was performed in a room at Carleton University (Ottawa, Canada) which measured 3.15 x 3.35 x 2.95 m which resembled a confined environment similar to a prison cell. The radar was wall mounted 2.70 m above floor level in one corner of the room.

In this paper, human activity levels were divided into the following three classes:

1) *Sedentary and Still (SS-class)*: The subject remains in place motionless. Postures include sitting, standing and lying. Breathing patterns are either normal or the subject is instructed to hold their breath momentarily.
2) *Sedentary and Moving (SM-class)*: The subject remains in place but is instructed to move their limbs or head in a random fashion. Postures include sitting, standing and lying. Breathing patterns are either normal or the subject is instructed to hold their breath momentarily.
3) *Walking (W-class)*: The subject walks at approximately 0.5 m/s towards and away from the radar continuously for the duration of the test. Breathing patterns are normal.

The data collection protocol was designed to include as much variation in subject behavior as was practical. This variation included distance and relative angle between subject and radar, posture of the subject, and breathing rate and pattern. Subjects were asked to perform the data collection protocol at three different locations within the room, which can be seen in Fig. 1.

In total, four subjects participated in the tests. The protocol was as follows: one minute of normal breathing followed by one minute of cessation of breathing (or for as long as comfortably possible) finished by one minute of normal breathing while randomly moving limbs. This was repeated at location A, B and C. In each location, the protocol was repeated for the standing posture and seated posture. At location C, there was a bed placed along the wall and so an additional posture was added to the protocol. The posture was to lay in the bed in the left lateral recumbent position facing the radar. An additional test was recorded in which subjects paced back and forth between locations A and C for three minutes. In total 27 minutes of data was recorded for each of the four subject.

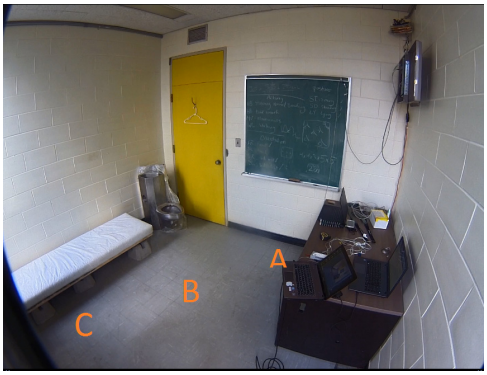


Fig. 1: Picture of the laboratory at the Carleton University with location markers. The radar is placed in the top right corner.

Data was labeled manually by segmenting the data every 10 seconds and devising a numeric code for activity class, posture, location and subject.

C. Preprocessing

Data segments were 10 seconds long with 9 range bins. This corresponded to data segment matrices of size 9050 x 9, the number of rows was defined by 10 seconds of data recorded at 905 samples/second and the number of columns corresponded to the number of preset range bins. From each data segment a single dimension time series vector was extracted. These time series vectors corresponded to the signal within the data that corresponded to reflections from the human target. To determine which of the 9 range bins, or columns the target was most likely present, the power of each column of data was computed. The column with the largest power was taken to be the column containing the target signal. The extracted time series vector for each data segment was therefore of size 9050 x 1. Of all of the data segments, 575 were kept. Some segments were corrupted with noise, or had been zero padded and were considered as erroneous segments. The elimination of erroneous segments was done through manual inspection (through graphical visualization of data).

A 2^{nd} order Butterworth bandpass filter with upper and lower frequency bounds of 0.08 Hz and 20 Hz respectively was applied to all single dimensional data segments (extracted time series vectors). This removed the DC bias resulting from environmental clutter as well as high frequency noise.

D. Feature Extraction

In total 43 features were extracted from the data. Features are shown in Table I. First eleven features are time domain features. In Table I, $x(n)$ represents the data segment of interest. The next 32 features are all categorized as frequency domain features. The frequency domain for each data segment was computed using a 2^{16} point Welch-Periodogram from the data segment $x(n)$. Here, $S[f[i]]$ represents the magnitude of

TABLE I: Extracted features

Index	Name	Explanation and Use
1	*Correlation*	Autocorrelation at lag l for the for the first 50 lags.
2	*Correlation_Breathing*	Autocorrelation of band pass filtered signal between the bounds of 0.2 Hz and 0.33 Hz.
3	*Correlation_Heart*	Autocorrelation of band pass filtered signal between the bounds of 0.883 Hz and 3 Hz.
4	*Root_Mean_Square*	$RMS = \sqrt{(E(x^2))}$, $E(\cdot)$ is the expectation function.
5	*Zero_Crossing*	Number of times $x(n)$ crosses zero divided by the length of $x(n)$.
6	*Turns_Count*	Zero crossing rate of the derivative of $x(n)$.
7	*Variance*	$E((x(n) - E(x(n)))^2)$
8	*Skewness*	Measure of how far the data strays from a normal distribution: $E((x(n) - E(x(n)))^3)$
9	*Kurtosis*	Measure of 'tailedness' of the distribution: $E((x(n) - E(x(n)))^4)$
10	*Mobility_Factor*	Measure of signal activity: $\frac{\sigma_{x'}}{\sigma_x}$
11	*Form_Factor*	$\frac{\sigma_{x''}/\sigma_{x'}}{\sigma_{x'}/\sigma_x}$
12	*Total_Power*	the sum of magnitudes the Welch-Periodogram spectrum divided by the number of spectrum points
13	*Mean_Frequency*	$f_{mean} = \sum_{i=0}^{N-1} S[f[i]]f[i] / \sum_{i=0}^{N-1} S[f[i]]$
14	*Median_Frequency*	The frequency that divides the integrable power in the spectrum into two equally summable parts.
15	*Second_Spectral_Moment*	$E((S-\mu)^2)$ where μ is the mean value of the spectrum.
16	*Third_Spectral_Moment*	$E((S-\mu)^3)$
17	*Frequency_Band_1*	$\sum_{i_1}^{i_2} S[f[i]]$ where $f(i_1) = 0.2Hz$, $f(i_2) = 0.667Hz$.
18	*Frequency_Band_2*	$\sum_{i_2}^{i_3} S[f[i]]$ where $f(i_3) = 3Hz$.
19	*Frequency_Band_3*	$\sum_{i_3}^{i_4} S[f[i]]$ where $f(i_4) = 5Hz$.
20	*Frequency_Band_4*	$\sum_{i_4}^{i_5} S[f[i]]$ where $f(i_5) = 8Hz$
21	*Frequency_Band_5*	$\sum_{i_5}^{i_6} S[f[i]]$ where $f(i_6) = 11Hz$.
22	*Frequency_Band_6*	$\sum_{i_6}^{N-1} S[f[i]]$.
23	*Frequency_First_Harm*	$\sum_{i_1}^{i_{11}} S[f[i]]$ where $f(i_{11}) = 0.33Hz$.
24	*Frequency_Sec_Harm*	$\sum_{i_{11}}^{i_2} S[f[i]]$.
25	*F_Ratio_1_2*	*Frequency_Band_1/Frequency_Band_2*
26	*F_Ratio_1_3*	*Frequency_Band_1/Frequency_Band_3*
27	*F_Ratio_1_4*	*Frequency_Band_1/Frequency_Band_4*
28	*F_Ratio_1_5*	*Frequency_Band_1/Frequency_Band_5*
29	*F_Ratio_1_6*	*Frequency_Band_1/Frequency_Band_6*
30	*F_Ratio_2_3*	*Frequency_Band_2/Frequency_Band_3*
31	*F_Ratio_2_4*	*Frequency_Band_2/Frequency_Band_4*
32	*F_Ratio_2_5*	*Frequency_Band_2/Frequency_Band_5*
33	*F_Ratio_2_6*	*Frequency_Band_2/Frequency_Band_6*
34	*F_Ratio_3_4*	*Frequency_Band_3/Frequency_Band_4*
35	*F_Ratio_3_5*	*Frequency_Band_3/Frequency_Band_5*
36	*F_Ratio_3_6*	*Frequency_Band_3/Frequency_Band_6*
37	*F_Ratio_4_5*	*Frequency_Band_4/Frequency_Band_5*
38	*F_Ratio_4_6*	*Frequency_Band_4/Frequency_Band_6*
39	*F_Ratio_5_6*	*Frequency_Band_5/Frequency_Band_6*
40	*FB1_FB2*	*Frequency_First_Harm/Frequency_Sec_Harm*
41	*Shannon_Entropy*	$H(S) = -\sum_{i=0}^{N-1} P(i)\log_2(P(i))$, where $P(i)$ is the spectrum at point i normalized by the sum of magnitudes of S.
42	*Shannon_Entropy_Breathing*	$H(S)$ for the band pass filtered signal in $[0.2Hz, 0.33Hz]$.
43	*Shannon_Entropy_Heart*	$H(S)$ for the band pass filtered signal in $[0.833Hz, 3Hz]$.

the spectrum S at frequency index $f[i]$ and N is the number of samples in the frequency spectrum S.

The data feature matrix consisted of 575 rows of 43 dimensional feature vectors. The feature values were converted to z-scores by subtracting the column means and dividing by column standard deviation. An additional column (column 44) was added to include numerical class labels.

III. Unsupervised Analysis

Data observations are usually described in terms of measured and/or computed attributes. Modern developments in sensor, communication and computing technologies have increased the amounts of data generation (objects and also features). While larger feature dimensionality can lead to better discrimination, it inevitably introduces many redundancies. Moreover, the non-redundant information in the features is contained on low dimension non-linear manifolds within the feature space and the dimension of these manifolds is the real (intrinsic) dimensionality of the data. Learning the mapping of the data onto the non-linear manifolds is useful to understand the true structure of the information, for mitigating the curse of dimensionality and also for lowering the computational load of machine learning procedures. Different techniques have been proposed for estimating intrinsic dimensionality (correlation dimension (CD), maximum likelihood (MLE), Takens, nearest neighbours, geodesic entropic graphs (GMST), U-statistic and others) [4], [8], [9], [14], [18], [17]. There are also many lower dimensional mappings techniques. The one used here is the t-Distributed Stochastic Neighbor Embedding (t-SNE) [22], which is an enhancement of the SNE [10] approach. The aim is to minimize the Kullback-Leibler divergence between the conditional probability distributions between data points of the original and the lower dimension space. The drawback of SNE related to the low cost when representing highly separated points is overcome by t-SNE by using a Student's t-distribution in the target space which has a heavier tail and addresses the crowding problem created by using a Gaussian representation. This approach has been successfully used for data visualization in a broad range of application domains.

IV. Feature Selection Methods - Gamma Test (Residual Variance) Analysis

The Gamma test is a nonparametric technique aimed at estimating the variance of the noise present in a dataset [5], [11], [16]. Noise is considered as any source of variation in the target variable that cannot be explained by a smooth function relating the target with the predictor variables. The gamma estimate indicates whether it is possible to explain the target variable by a smooth deterministic model based on the predictor variables.

Let $\mathcal{S}$ be a system described by a set of variables and with $y \in \mathbb{R}$ being a variable of interest, potentially related to a set of m variables $\overleftarrow{x} \in \mathbb{R}^m$ expressed as $y = f(\overleftarrow{x}) + r$, where f is a smooth unknown function representing the system, $\overleftarrow{x}$ is a set of predictor variables and r is a random variable representing noise or unexplained variation. Let M be the number of observations and p be the number of nearest neighbors considered. If $\overleftarrow{x}_{N\lfloor i,k \rfloor}$ is the k-th nearest neighbor of object $\overleftarrow{x}_i$, for every $k \in [1, p]$, a sequence of estimates of $\mathbf{E}\left(\frac{1}{2}(y' - y)^2\right)$ based on sample means is computed as $\gamma_M(k) = \frac{1}{2M}\sum_{i=1}^{M} |y_{N\lfloor i,k \rfloor} - y_i|^2$, $\delta_M(k) = \frac{1}{M}\sum_{i=1}^{M} |\overleftarrow{x}_{N\lfloor i,k \rfloor} - \overleftarrow{x}_i|^2$ where $\mathbf{E}$ denotes the mathematical expectation and $|.|$ Euclidean distance. The relationship between $\gamma_M(k)$ and $\delta_M(k)$ is assumed linear as $\delta_M(k) \to 0$ and an estimate for the variance of the noise $\Gamma = Var(r)$ is obtained by linear regression of $\delta_M(k)$ vs. $\gamma_M(k)$ as $\gamma_M(k) = \Gamma + G\,\delta_M(k)$. From it, the vRatio ($V_r$) is defined as a normalized Γ value with respect to the variance of the target variable ($V_r = \frac{\Gamma}{Var(y)}$). Since $V_r \in [0, 1]$, it allows comparisons across different datasets. Assessing the

relevance of the predictor variables is approached by searching for subsets with good Γ-statistics. The full search space is determined by the power set of the predictor variables and evolutionary computation methods provide an alternative to the prohibitive exhaustive search. In the present study, a genetic algorithm explores subsets of predictors represented as binary vectors $\overleftarrow{\vartheta} = \{0,1\}^m \in \mathbb{R}^m$. Each one codes a subset of predictors and the potential of each subset is given by the Γ-statistics, which could be specified in different ways. A single-objective cost function can be formulated as a linear combination of partial fitness coming from *i)* the MSE as associated to V_r (the I_f term), *ii)* 'model smoothness' as associated to G (the G_f term) and *iii)* 'model complexity' given by the relative number of predictors (the L_f term). $F(\overleftarrow{\vartheta}) = W_i * I_f(\overleftarrow{\vartheta}) + W_g * G_f(\overleftarrow{\vartheta}) + W_l * L_f(\overleftarrow{\vartheta})$, where $W_i = 0.8$, $W_g = 0.1$, $W_l = 0.1$ are the weights of the contributing fitness terms, the largest of which is given to I_f, directly related to the estimated MSE.

$$
\begin{aligned}
I_f(\overleftarrow{\vartheta}) &= \begin{cases} 1-(1-10*V_r(\overleftarrow{\vartheta}))^{-1} & \text{if } V_r(\overleftarrow{\vartheta}) < 0 \\ 2-2(1+V_r(\overleftarrow{\vartheta}))^{-1} & \text{otherwise} \end{cases} \\
G_f(\overleftarrow{\vartheta}) &= 1-(1+|G(\overleftarrow{\vartheta})|/range(y))^{-1} \\
L_f(\overleftarrow{\vartheta}) &= \sum \overleftarrow{\vartheta}/m
\end{aligned} \tag{1}
$$

The choice of the weights $\{W_i, W_g, W_l\}$ is a compromise between the importance given to the partial fitness components coming from the subset's V_r, the model complexity G and the model's cardinality (the smaller, the simpler, since it contains fewer predictors). The practical form of $\{I_f, G_f, L_f\}$ in (1) is a heuristic that has been successful in different real world domains.

V. Classification Techniques

For this paper, two distinct classification approaches were explored. First, black box models including Random Forests, Extreme Learning Machines and Multilayer Perceptrons are used. These models have been shown to work very well on similar classification problems and are presented in this paper as a baseline for performance. Secondly, a white box model is presented. This model uses Genetic Programming to uncover a function using the given feature set. The clear benefit of this approach is the ability to analyze underlying phenomenon in the structure of the data. Unlike the black box models, white box models allow the researcher to learn about the problem.

A. Genetic Programming

Genetic Programming (GP) is a kind of evolutionary algorithm introduced in [12], [13] as an extension of the Genetic Algorithm (GA) to specifically evolve a population of computer programs. There are several variants of GP and in particular, the one used here is Gene Expression Programming (GEP) [6], [7], where individuals are expression trees encoded as strings of fixed length [6]. GEP uses an unambiguous translation system to transfer the language of chromosomes into the language of expression trees and vice versa. In the present case, the task is to classify the observations into binary classes. The way in which this is achieved is by using GP (GEP) to learn a decision function of the form

```
IF          F(x⃖) ≥ T_h
THEN        class=1
OTHERWISE   class=0
```

where $\overleftarrow{x} \in \mathbb{R}^n$ is a vector composed of $n \in \mathbb{N}^+$ predictor variables, $\mathbf{F} : \mathbb{R}^n \rightarrow \mathbb{R}$ is a real-valued function of the elements of the predictor vector (possibly with some constant terms), $class$ represents the decision variable of interest (with only two categories) and $T_h \in \mathbb{R}$ is a real-valued constant (threshold) for transforming the continuous function values into a binary output ($\{0,1\}$). In GP, the **F** function is built by gradually assembling functional terms from the Function Set (user-defined) as well as constants that are considered as 0-arity functions. GEP chromosomes have a structure composed of a head (composed of functions and terminals) and a tail (composed only of terminal terms). The former encodes the functions chosen for the problem, and the later is a reservoir that ensures the formation of only valid structures.

There are specific parameters controlling the size of the functions that can be generated and optionally, an algebraic simplification step, introduced in [21], is applied at the post-processing stage. There are genetic operators shared by most EAs, but there are several specific to GEP which are described in [7].

VI. Experimental Settings

The data set of size 575 x 43 was partitioned into 75%/25% training and testing sets following a uniform random distribution. Three binary classifiers were produced (one for each class) in order to analyze the relationship between each feature and each class.

1) Feature Subset Extraction Settings: For determining the feature subsets of each class, a genetic algorithm with chromosomes of length $m = 43$ were used with a population of 100. The single point crossover rate and bit mutation rate were 0.5 and 0.05 respectively. The fitness function referred to earlier (Eq. 1) was used, and the Γ -statistics were obtained using 10-nearest neighbors as suggested by preliminary experiments. A Dell Precision workstation with an Intel(R) Xeon(R) dual core X5570 2.93 GHz ran the evolutionary process for a duration of 20 minutes. From each classification problem, the predictor variables emerging from the best individual were used for creating a newer version of the training and testing sets.

2) Classifier Settings: The settings used for the Random Forests were: $batchsize = 100$, $bagsize = 100\%$ and unlimited tree depth. To determine the best number of hidden layer sigmoid activation neurons in the Extreme Learning Machine (ELM), 10-fold cross validation was used with 5 random initializations. The MLP classifier had a hidden layer with 6 sigmoid neurons and a output layer with a single sigmoid neuron. The number of hidden neurons was determined based on the intrinsic dimensionality estimation [19] and

TABLE II: Experimental Settings for the GEP experiments

Parameter	Value
Number of Generations	10000
Population Size	30
Fitness Function	ACC, CE, $CE_{10}\_FNR_{90}$, $CE_{20}\_FNR_{80}$, $CE_{30}\_FNR_{70}$, $CE_{40}\_FNR_{60}$, $CE_{50}\_FNR_{50}$, $CE_{60}\_FNR_{40}$, $CE_{70}\_FNR_{30}$, $CE_{80}\_FNR_{20}$, $CE_{90}\_FNR_{10}$, FNR, SS, PPV, NPV, $SSPN$
Classification Threshold	1, 2, 3, 4, 5, 6, 7, 8, 9, 10
Number of Genes	6, 8
Number of Constants per Gene	2
Constants lower/upper Bounds	[-10, 10]
Head Size	6, 8, 10
Use Constants	Yes
Random Numbers Generation Seeds	5 machine-generated random seeds
Function Set	$+, -, *, abs, x^2, x^3, sin, cos, atan$
Linking Function	$+$

the network was trained with the Broydon-Fletcher-Goldfarb-Shanno (BFGS) algorithm [15]. For GEP, the following genetic operators were used: Inversion Probability = 0.1, Mutation Probability = 0.044, Is-transposition Probability = 0.1, Ris-transposition Probability = 0.1 , One Point Recombination Probability = 0.3, Two Point Recombination Probability = 0.3, Gene Recombination Probability = 0.1, Gene Transposition Probability = 0.1, Use Constants = TRUE, Number of Constants per Gene = 2, Integer Constants = FALSE, Constants Lower Limit = -10, Constants Upper Limit = 10, RNC Mutation Probability = 0.01, DC Mutation Probability = 0.044, DC Inversion Probability = 0.1, and DC Is-transposition Probability = 0.1. The following fitness functions were used in the binary classification experiments:

- Classification Accuracy (ACC): $ACC = (TP+TN)/N$
- Classification Error (CE): $CE = 1 - ACC$
- False Negative Ratio (FNR): $FNR = FN/(FN+TP)$
- Sensitivity (SE): $SE = TP/(TP + FN)$
- Specificity (SP): $SP = TN/(TN + FP)$
- Sensitivity-Specificity (SS): $SS = SE * SP$
- Positive Predictive Value (PPV): $PPV = TP/(TP + FP)$
- Negative Predictive Value (NPV): $NPV = TN/(TN + FN)$
- Positive Predictive Value - Negative Predictive Value ($PPVNPV$): $PPVNPV = PPV * NPV$
- Sensitivity-Specificity-$PPVNPV$ ($SSPN$): $SSPN = SE * SP * PPV * NPV$
- Mixed Classification Error and False Negative Ratio ($CE_{\alpha}\_FNR_{\beta}$): $CE_{\alpha}\_FNR_{\beta} = (\alpha * CE + \beta * FNR)/100$, with $\alpha \in \{10, 20, 30, 40, 50, 60, 70, 80, 90\}$, $\beta = 100 - \alpha$. This is a convex combination tradeoff between overall classification error and false negatives.

TP: true positives, FP: false positives, TN: true negatives, FN: false negatives. The rest of the GEP experimental settings are shown on Table II.

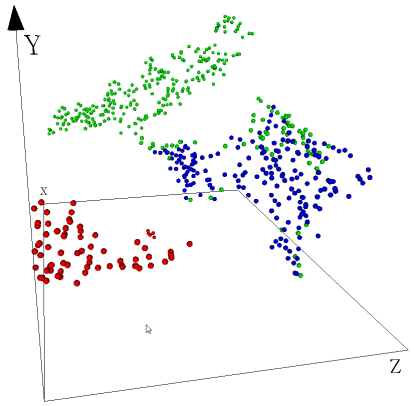


Fig. 2: 3-D Mapping of the radar data obtained with t-SNE.

VII. Results

A. Intrinsic Dimensionality, 3D Mapping and Visualization

The intrinsic dimensionality estimates of the data set obtained from the methods discussed in Section III were: PCA = 7 (with 97.5% cumulated variance), MLE = 5.988, Correlation Dimension = 1.346, GMST = 6.175, Takens = 3.571 and U-statistic = 4.000. Although the estimates have high variability, they are all considerably lower than the dimensionality of the original feature space (43). This indicates that there is a lot of redundancy contained in the original data space, and dimensionality reduction can be performed without the loss of much information. As the mean and median intrinsic dimensionality estimates are near 3, unsupervised mappings of the data to 3D can be constructed (not using class information) to visualize an approximation to the internal structure. The class distribution is overlaid *a posteriori* for comparison A snapshot of the $\mathbb{R}^{43} \rightarrow \mathbb{R}^3$ mapping obtained with t-SNE is shown on Fig. 2. for visualization purposes as color (Green: SS-class, Blue: SM-class, Red: W-class). In the t-TSN 3D space, the W-class appears completely separable from the others, whilst classes SS and SM exhibits some overlap. The SS-class appears as bi-modal, with a low density cluster intersecting the SM-class, more distinguishable from the higher density mode of the SS-class distribution. Interestingly, the SM-class lies between the SS-class and the W-class. This indicates a clear trajectory within the subspace corresponding to level of movement. Also, the intersection with the SS-class happens with the lower density elements of the SM-class, which could represent subjects that had accidentally moved a part of the body when they were told to be still.

B. Classification Using the Original Feature Set

Classification was performed with two different sets of features: the entire 43 dimensional feature set and the feature set which emerged from the genetic algorithm guided $\Gamma-$Test. They were used to train and validate classifiers for each of the three classes. The results for the reference models using the 43 dimensional feature set are shown in Table III and provide a baseline for comparison with the results using the genetic

TABLE III: Classification results using 43 features

SS-Class (43 Features)						
Classifier	Accuracy	TPR	FPR	TNR	FNR	V_r
R. Forest	0.972	0.975	0.031	0.969	0.025	0.134
ELM	0.979	0.975	0.015	0.985	0.025	0.134
MLP	0.958	0.962	0.046	0.954	0.038	0.134
SM-Class (43 Features)						
Classifier	Accuracy	TPR	FPR	TNR	FNR	V_r
R. Forest	0.972	0.957	0.020	0.980	0.043	0.149
ELM	0.951	0.913	0.031	0.969	0.087	0.149
MLP	0.965	0.913	0.010	0.990	0.087	0.149
W-Class (43 Features)						
Classifier	Accuracy	TPR	FPR	TNR	FNR	V_r
R. Forest	1	1	0	1	0	0.014
ELM	1	1	0	1	0	0.014
MLP	1	1	0	1	0	0.014

TABLE IV: Classification results using Γ-test features

SS-Class (24 Features)						
Classifier	Accuracy	TPR	FPR	TNR	FNR	V_r
R. Forest	0.965	0.962	0.031	0.969	0.038	0.0219
ELM	0.965	0.962	0.031	0.969	0.038	0.0219
MLP	0.944	0.937	0.046	0.954	0.063	0.0219
SM-Class (22 Features)						
Classifier	Accuracy	TPR	FPR	TNR	FNR	V_r
R. Forest	0.979	0.978	0.020	0.980	0.022	0.0379
ELM	0.965	0.957	0.031	0.969	0.043	0.0379
MLP	0.924	0.891	0.061	0.939	0.109	0.0379
W-Class (24 Features)						
Classifier	Accuracy	TPR	FPR	TNR	FNR	V_r
R. Forest	1	1	0	1	0	$2 \cdot 10^{-8}$
ELM	1	1	0	1	0	$2 \cdot 10^{-8}$
MLP	1	1	0	1	0	$2 \cdot 10^{-8}$

algorithm feature set results and the GEP. The table includes the V_r values corresponding to the training set. The lower the values, the higher the determinism between the predictors and the class variable. The V_r values for each class in Table III are low, indicating that classifier models can be found which are able to discriminate the classes with high accuracy. This is corroborated with the classification performance measures. The value of V_r for the W-class is an order of magnitude lower than for the other classes, explaining why that class is fully discriminated. As anticipated from the visualization of the data in Fig. 2 and the V_r values, the classifiers trained on the 43 dimensional feature set performed well. The SS-class and SM-class were classified with high accuracy (over 95%), despite of their overlapping. For all models, the FNR for the SS-class is lower than for the SM-class. In particular, the distinction between SS-class and SM-class is important for health care applications. In cases where senior citizens are being monitored, long uninterrupted periods of SS-class may be an indication of a problem and can automatically trigger an alert to nearby health care professionals.

C. Classification Using Feature Set Derived From the Γ-test

The results for the classifiers tested on the reduced feature space resulting from the Γ-test based genetic algorithm method applied to each class are shown in Table IV. The number of features extracted using the genetic algorithm method is in the range of [22,24] which is nearly half of the number of original features. Moreover, V_r became one order of magnitude smaller, indicating that the reduced sets of features have more explanatory power than the originals.

The results of the classifiers on the reduced dimensionality feature set are nearly the same as those obtained with the classifiers using the original features. The accuracy of all three classifier types for SS-class are only slightly reduced for the reduced dimensionality feature set, while the accuracy of two of the three classifiers for SM-class increased slightly. It is interesting to note that the SM-class classifiers, which experienced an increase in accuracy, used the smallest number of relevant features of the three classes (22 of the original 43 features). The accuracy of the W-class classifiers remain unchanged at 100% for the Γ-test feature set. These results show that there is a lot of redundancy in the original feature space, and that redundancy can effectively be reduced using the Γ-test genetic algorithm without sacrificing accuracy.

The reference classifiers had comparative performance and Random Forests and ELM were the top ones. MLP, while performing with the least accuracy, was the simplest classifier of all three. The MLP had a single hidden layer of 6 neurons while the ELM required 148, 196 and 153 hidden layers for SS-class, SM-class and W-class respectively (with 43-D feature set). With the reduced dimensional feature set, ELM still required 105, 81 and 148 hidden layers for the three respective classes. While ELM is complex in size, MLP is complex in training (ELM involves no iterations through the network and trains with linear regression, while MLP is trained with BFGS). Random Forest is complex as well, using 100 decision trees for each classifier.

Table V presents the 43 original features with a column for each class indicating (with 1) if the feature was selected as relevant by the Γ-test based genetic algorithm. There were 6 features completely irrelevant for all classes: *Total_Power*, *Freq_Band_2*, *Freq_Band_5*, *Freq_Band_6*, *Freq_First_Harm* and *Freq_Sec_Harm*. Importantly, 7 features were found relevant for all classes: *Correlation*, *Root_Mean_Square*, *Zero_Crossing*, *Kurtosis*, *Mobility_Factor*, *F_Ratio_2_4* and *F_Ratio_2_6*. Interestingly, there are features which are relevant for one class but not for another. They contain information pertinent to a given type of human activity, particularly important for the differentiation of the SS and the SM classes.

D. Genetic Programming Model Extraction

The reference classifiers provided good performance, but are black-boxes. GP are white box models which explicit the relationships between the features and the classes. For the SS-class, 383 GEP models were found with $ACC \geq 0.965$ and $FNR \leq 0.03$. Ordered by accuracy, false negative ratio (both on the testing set) and arity, the top one model had $ACC =$

TABLE V: Γ-test relevant features for each class.

Index	Name	SS $Class$	SM $Class$	W $Class$
1	*Correlation*	1	1	1
2	*Correlation_Breathing*		1	1
3	*Correlation_Heart*	1	1	
4	*Root_Mean_Square*	1	1	1
5	*Zero_Crossing*	1	1	1
6	*Turns_Count*	1	1	
7	*Variance*			1
8	*Skewness*		1	1
9	*Kurtosis*	1	1	1
10	*Mobility_Factor*	1	1	1
11	*Form_Factor*		1	
12	*Total_Power*			
13	*Mean_Frequency*		1	1
14	*Median_Frequency*	1	1	
15	*Second_Spectral_Moment*			1
16	*Third_Spectral_Moment*	1		
17	*Frequency_Band_1*			1
18	*Frequency_Band_2*			
19	*Frequency_Band_3*			1
20	*Frequency_Band_4*			1
21	*Frequency_Band_5*			
22	*Frequency_Band_6*			
23	*Frequency_First_Harm*			
24	*Frequency_Sec_Harm*			
25	*F_Ratio_1_2*	1		1
26	*F_Ratio_1_3*		1	1
27	*F_Ratio_1_4*	1	1	
28	*F_Ratio_1_5*	1		
29	*F_Ratio_1_6*			1
30	*F_Ratio_2_3*		1	1
31	*F_Ratio_2_4*	1	1	1
32	*F_Ratio_2_5*	1		1
33	*F_Ratio_2_6*	1	1	1
34	*F_Ratio_3_4*	1		
35	*F_Ratio_3_5*	1		
36	*F_Ratio_3_6*	1		1
37	*F_Ratio_4_5*	1		1
38	*F_Ratio_4_6*		1	1
39	*F_Ratio_5_6*	1	1	
40	*FB1_FB2*	1	1	
41	*Shannon_Entropy*	1	1	
42	*Shannon_Entropy_Breathing*	1		
43	*Shannon_Entropy_Heart*	1	1	

0.986, $FNR = 0.0127$ and $Arity = 10$:

$$\begin{aligned}&\text{IF } (((((((F\_Ratio\_1\_3 - atan(Frequency\_First\_Harm))\\&+ cos((((F\_Ratio\_2\_6^2) + sin((Frequency\_Band\_2^2)))\\&- cos(abs((Variance^3)))))) + cos(Median\_Frequency))\\&+ ((-4.029297606999815)Zero\_Crossing))\\&+ sin(abs(F\_Ratio\_1\_3))) + abs(abs(F\_Ratio5\_6)))\\&+ (F\_Ratio\_2\_5 - F\_Ratio\_5\_6))\\&+ abs(((atan(Turns\_Count)^3)^2))) \geq 3\\&\text{THEN SS-Class is TRUE} \qquad (2)\end{aligned}$$

The following equation, with $Arity = 14$, $Th = 7$ (threshold), predicts SS-class with $ACC = 0.979$, has $FNR = 0$ on the testing set:

$$\begin{aligned}&\text{IF } (((((((cos(F\_Ratio\_2\_4) + (5.543789157637942\\&- Shannon\_Entropy\_Heart))\\&+ (Form\_Factor((Frequency\_Sec\_Harm\\&+ (Zero\_Crossing - sin((F\_Ratio\_1\_4\\&* Zero\_Crossing))))^3))) + abs(cos(((Variance\\&+ Frequency\_Band\_1)^3)))) + ((sin(cos(Correlation))\\&(abs(cos(F\_Ratio\_5\_6))^3))abs(F\_Ratio\_4\_6)))\\&+ atan(F\_Ratio\_2\_5)) + atan((Form\_Factor^2)))\\&+ (atan(abs((((F\_Ratio\_2\_5^2)(Mean\_Frequency\\&- Turns\_Count))cos(atan(Mean\_Frequency)))))^3)) \geq 7\\&\text{THEN SS-Class is TRUE} \qquad (3)\end{aligned}$$

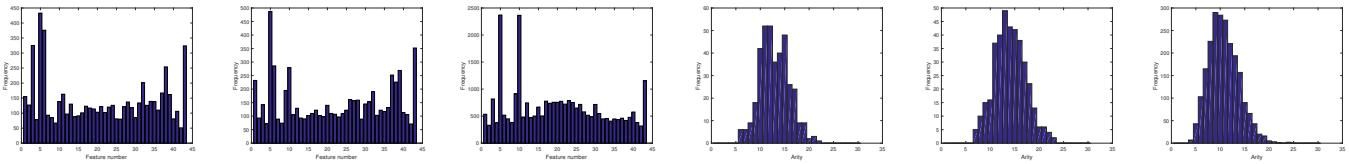
For the SM-class, 378 models were found, using the same criteria as for SS-class. The top one , with $Arity = 8$, $Th = 3$ (threshold), predicts SM-class with $ACC = 0.986$ and has $FNR = 0$ on the testing set:

$$\begin{aligned}&\text{IF } ((((((atan(cos(((F\_Ratio\_4\_6^3) - Turns\_Count)))\\&+ sin((atan(Shannon\_Entropy\_Heart)^3)))\\&+ (Frequency\_Band\_4 - atan(F\_Ratio\_2\_4)))\\&+ cos(Zero\_Crossing)) + cos(Zero\_Crossing))\\&+ atan(abs(sin(Shannon\_Entropy\_Heart))))\\&+ (F\_Ratio\_1\_3 * Correlation)) \geq 3\\&\text{THEN SM-Class is TRUE} \qquad (4)\end{aligned}$$

For the $Wclass$, 1977 models were found, all having $ACC = 1$ and has $FNR = 0$ on the testing set. The top one was:

$$\begin{aligned}&\text{IF } (((((atan((Variance^3)) + Zero\_Crossing)\\&+ cos(Frequency\_Band\_6)) + (Mobility\_Factor^3))\\&+ atan(Frequency\_Band\_6)) + Mobility\_Factor) \geq (4)\\&\text{THEN SM-Class is TRUE} \qquad (5)\end{aligned}$$

It is interesting to examine the composition of the best models from the point of view of the frequency with which the different predictor variables appear (the most important variables would tend to appear more often in the models that perform well for the given class). Fig. 3 shows the distributions of feature occurrence and arity for the top performing GEP models. The feature distributions highlight a few, very important variables for each class. Many features appear with similar frequencies, which likely indicates interdependencies among them. Arity distributions are centered for the SS and SM classes, but skewed for W indicating that models for that class tend to require fewer predictors. The findings corroborate that, the key to high performance is not necessarily having many predictor features, but having the right ones. In the present case, GEP was able to find the right features, as well as the right functional relationships between them, producing highly accurate classification models.



(a) Feature for SS (b) Feature for SM (c) Feature for W (d) Arity for SS (e) Arity for SM (f) Arity for W

Fig. 3: Feature and model arity distributions in top performing genetic programming models for all classes.

VIII. Conclusions

Human activity classification using returns from a CW Doppler radar were investigated with unsupervised and supervised computational intelligence techniques using data described by 43 time and frequency domain features. The feature space has a low intrinsic dimension, indicating redundancies in the original features. The t-SNE low dimensional mapping showed a clear separation between the W-class and the other two classes. The SS-class and the SM-class showed an overlap, suggesting that classification results for them should be expected to be lower compared those for the fully separable W-class, as was found.

Modeling used the original feature set and a smaller subset found with the Γ-test, using random forest, extreme learning machines and multilayer perceptron (black-box methods) and with gene expression programming (white-box). For both feature sets, all classifiers perfectly retrieved W-class. The SS and the SM classes, while overlapping in 3D space, were classified with accuracies of over 95% for the 43D classifiers and over 92% for the reduced feature subset. Classifier performance experienced very little degradation even when the feature space was drastically reduced.

The explainability associated to the GP models provided valuable insight and the statistics derived from the top performing models identified the strongest features and the redundant ones. Interestingly, the intrinsic dimensionality estimates showed that models with arities close to the intrinsic dimension performed as well as the ones with the highest arity. Overall, it was possible to drastically reduce the size of the predictor space without sacrificing classification performance. The application of this approach to the problem of human activity classification using CW radar revealed the possibilities of genetic programming techniques in producing highly accurate and explainable models. They were able to produce accurate and explainable models with similar and sometimes better performances than well established black-box methods. Such models also uncovered subsets of relevant predictor features.

Acknowledgment

This project was funded through NSERC Discovery and OGS grants.

References

[1] Fadel Adib, Hongzi Mao, Zachary Kabelac, Dina Katabi, and Robert C. Miller. Smart homes that monitor breathing and heart rate. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, CHI '15, pages 837–846, New York, NY, USA, 2015. ACM.

[2] Z. Baird. Human activity and posture classification using single non-contact radar sensor. Master's thesis, Carleton University, 2017.

[3] Z. Baird, J.J. Valdés, S. Rajan, and M. Bolic. Classification of human activity level using single channel cw doppler radar. In *International Conference on Pattern Recognition and Artificial Intelligence ICPRAI 2018*, CENPARMI, Montreal, 2018.

[4] J.A. Costa and A.O. Hero. Geodesic entropic graphs for dimension and entropy estimation in manifold learning. *IEEE Transactions on Signal Processing*, 52(8):2210–2221, 2004.

[5] D. Evans and A.J. Jones. A proof of the gamma test. *Proc. Roy. Soc. Lond. A*, 458:1–41, 2002.

[6] C. Ferreira. Gene expression programming: A new adaptive algorithm for problem solving. *Journal of Complex Systems*, 13(2):87–129, 2001.

[7] C. Ferreira. *Gene Expression Programming: Mathematical Modeling by an Artificial Intelligence*. Springer Verlag, 2006.

[8] P. Grassberger and I. Procaccia. Measuring the strangeness of strange attractors. *Physica*, D(9):189–208, 1983.

[9] M. Hein and J.Y. Audibert. Intrinsic dimensionality estimation of submanifolds in euclidean space. In L. de Raedt and S. Wrobel, editors, *Proceedings of the 22nd International Conference on Machine Learning (ICML)*, pages 289–296, 2005.

[10] G.E. Hinton and S.T. Roweis. Stochastic neighbor embedding. *In Advances in Neural Information Processing Systems*, 15:833–840, 2002.

[11] A.J. Jones, D. Evans, S. Margetts, and P. Durrant. *The Gamma Test. In Heuristic and Optimization for Knowledge Discovery*. Sarker, Abbass and Newton eds. Idea group Publishing, San Francisco, CA, 2002.

[12] J. Koza. Hierarchical genetic algorithms operating on populations of computer programs. In *Proceedings of the 11th International Joint Conference on Artificial Intelligence. San Mateo, CA*, pages 768–774. Morgan Kaufmann, 1989.

[13] John Koza. *Genetic programming: On the programming of computers by means of natural selection.* MIT Press, 1992.

[14] E. Levina and P.J. Bickel. Maximum likelihood estimation of intrinsic dimension. *Advances in Neural Information Processing Systems*, 17:777–784, 2005.

[15] J. Nocedal and S.J. Wright. *Numerical Optimization (2nd ed.)*. Springer-Verlag, Berlin, New York, 2nd edition, 2006.

[16] A. Stefánsson, N. Končar, and A.J. Jones. A note on the gamma test. *Neural Computing and Applications*, 5:131–133, 1997.

[17] F. Takens. On the numerical determination of the dimension of an attractor. *Dynamical systems and bifurcations*, pages 99–106, 1985.

[18] G. V. Trunk. Statistical estimation of the intrinsic dimensionality of a noisy signal collection. *IEEE Transaction on Computers*, 25:165—-171, 1976.

[19] J.J. Valdés, Z. Baird, S. Rajan, and M. Bolic. Pattern structure of human motion using single channel cw doppler radar: An unsupervised perspective. In *International Conference on Pattern Recognition and Artificial Intelligence ICPRAI 2018*, CENPARMI, Montreal, 2018.

[20] J.J. Valdés, Z. Baird, S. Rajan, and M. Bolic. Single channel cw doppler radar for differentiating types of human activity. In *Proc. IEEE International Joint Conference on Neural Networks, IJCNN 2018*, Rio de Janeiro, 2018.

[21] J.J. Valdés, A.J. Barton, and R. Orchard. Virtual reality high dimensional objective spaces for multi-objective optimization: An improved representation. In *Proceedings IEEE 2007 Congress on Evolutionary Computation.*, pages 4191–4198, Singapore, Sept 25-28 2007.

[22] L.J.P. van der Maaten and G. Hinton. Visualizing high-dimensional data using t-sne. *Journal of Machine Learning Research*, 2008.