Exercise and Sedentary Activity Recognition Using Late Fusion: Building Adaptable Uncertain Models

Ezequiel Juarez Garcia[1], Victoria R. Rodrigues[1], Mehrdad Fazli[2], Laura E. Barnes[2], and Nicholas J. Napoli[1]
[1]Dept. of Electrical and Computer Engineering, University of Florida, Gainesville, FL 32611
[2]Dept. of Systems and Information Engineering, University of Virginia, Charlottesville, VA 22903
[1]{ejuarezgarcia,victoria.ribeiro,n.napoli}@ufl.edu, [2]{mf4yc,lb3dp}@virginia.edu

Abstract*—Wearable smart devices are capable of capturing a variety of information from their users using a multitude of noninvasive sensing modalities. Using features from the raw measurements of wearable devices, sensor fusion enables us to obtain a holistic picture of the users' context and monitor their activity state with increased accuracy. Human activity recognition using noninvasive sensors allows us to capture the natural behavior of users in their day-to-day lives. This in-the-wild activity recognition, however, poses several key challenges that must be addressed to create effective classification models. The main challenges are class imbalance, uncertainty in classifier decisions, and large feature spaces. To address them, this study further explores a probabilistic sensor fusion method called Naive Adaptive Probabilistic Sensor (NAPS) Fusion. In doing so, we establish the viability of NAPS Fusion for natural human activity recognition using noninvasive sensing modalities. NAPS Fusion handles dimensionality reduction by creating reduced feature sets and mitigates the class imbalance issue through the use of Synthetic Minority Oversampling Technique (SMOTE). Moreover, NAPS Fusion addresses uncertainty in the decisions of classifiers using a Dempster-Shafer theoretic late fusion framework. Our empirical evaluation demonstrates that NAPS Fusion has broad applications beyond its original design for cognitive state detection. It outperforms similar decision level sensor fusion methods (late fusion using averaging, LFA, and late fusion using learned weights, LFL) in the detection of exercise and sedentary activities such as walking, running, lying down, and sitting. We observe improvements of up to 56% in F1 score and up to 59% in precision with NAPS Fusion over the compared methods.

I. Introduction

Human activity recognition can make a positive impact on the quality of life of users through the precise capture of human context using smart devices. Health monitoring, assisted living, and security monitoring are among the areas of interest that can benefit from activity recognition. For example, activity recognition can remind patients to take their medications on time [1], monitor the psychophysiological state of aircraft pilots [2], [3], and detect threat activity in homes [4]. With ubiquitous computing, many more areas stand to benefit from advances in sensor-based activity recognition.

Although human activity recognition has been an intently researched topic for decades [5], recent technological advancements in sensor technology have brought increased attention to it. For instance, the shift to micro-electromechanical systems (MEMS) technology for inexpensive and minute sensors such as accelerometers, gyroscopes, and microphones [6] has increased the number of sensors that are housed inside smartphones and smartwatches. This has facilitated the collection of rich and varied data from personal daily activities. In tandem with the surging interest and adoption in wearable technology, especially in healthcare monitoring applications [7], the need for human activity recognition is even greater.

Prior Work. Within human activity recognition, the characteristics of the sensor measurements and features oftentimes dictate the type of recognition method used. For example, each wearable sensor used for activity recognition tends to contribute more than a single feature to the dataset. Moreover, these sensors tend to capture raw measurements at a rate much greater than a few hertz. Without establishing proper relationships between a large feature space and different class labels, it is difficult to classify activities properly [8]. Research in human activity recognition has addressed the high dimensionality of datasets through the use of principal component analysis (PCA) [9] and neural networks with several hidden layers [10]. Due to the complex decision boundaries involved in high dimensional datasets, deep learning is one of the most commonly used approaches in human activity recognition [11], [12]. However, despite its widespread use, deep learning does not directly address the dimensionality problem.

Not only are the features generated by the sensors important in human activity recognition, but also the sensors themselves. Many research studies have classified human context by leveraging noninvasive sensors commonly found in portable, smart devices. For example, work in [13] and [14] provides single-sensor methods using accelerometers for human activity recognition. Accelerometers are a popular choice among single-sensor methods due to their ability to capture various types of physical, and often repetitive, activities. Despite their use, single-sensor methods cannot capture the full gamut of human activities. In an effort to improve classifier performance, studies tend to use multiple sensors for activity recognition [15], [16], [17]. Generally in multi-sensor activity recognition, a sensor fusion framework is used to combine information from the different sensors.

There exist many sensor fusion methods used to process and combine measurements from different sensors. These methods can be grouped into three categories: early fusion (data level fusion), intermediate fusion (feature level fusion), and late fusion (decision level fusion) [18]. Early fusion (EF) combines all the raw, often redundant, sensor data directly. This fusion technique works best for multiple homogeneous sensors, where the raw sensor data is fused without performing

any feature extraction. Intermediate fusion (IF) combines features from various sensor modalities and selects suitable features from those combinations. This technique is typically used in conjunction with dimensionality reduction methods such as principal component analysis. The main application of IF is in classification using general pattern recognition methods, such as neural networks. Although, this type of fusion suffers from information and performance loss, it is capable of incorporating data from both homogeneous and heterogeneous sensors. Late fusion (LF) combines the preliminary decisions of individual classifiers to form a final decision and improve classification accuracy. The most commonly used LF methods use Dempster-Shafer theory, fuzzy logic, or Bayesian inference [19]. Similar to IF, this technique also works for both homogeneous and heterogeneous sensors.

Leading work in human activity recognition at the University of California, San Diego (UCSD), has highlighted the difficulties related to detailed activity recognition and how sensor fusion can address these issues. A study by Vaizman et al. explored two distinct methods for late fusion. These LF methods were Late Fusion using Averaging (LFA) and Late Fusion with Learned weights (LFL) [20]. LFA averages the output prediction probabilities of single-sensor logistic regression classifiers to obtain a final prediction. This fusing method places equal weight to each sensor and avoids retraining after the initial learning of the single-sensor classifiers. Unlike LFA, LFL uses a logistic regression model as a second stage to obtain a final prediction given the probability outputs of the single-sensor classifiers as inputs. LFL exploits the fact that some sensors can perform better for different class labels (e.g., GPS is better for running than lying down) and assigns different weights to them. The use of logistic regression in LFL, however, may not provide the most effective method for adapting sensors to certain features. This can lead to the creation of fixed decision boundaries that are unable to adapt as new information or sensors are presented.

Challenges. Human activity comprises multiple complex body movements and environmental tasks performed in a specific order to achieve a desired state. Understanding these complex body and environmental interactions requires the use multiple sensors, with each sensor capturing a multitude of raw measurements and generating several features. The sheer number of measurements and features presents a high dimensionality challenge. This problem often crops up in pattern recognition research and, in particular, multi-sensor human activity recognition. Techniques such as principal component analysis (PCA) and correlation-based feature selection (CFS) [21] can reduce the number of features and speed up classification by decreasing the cost of computation and storage [22]. However, they can also degrade the performance of activity recognition by mapping the original feature space to a less interpretable one. Moreover, salient features in the original dataset can be lost. These features may only be prominent in the minority classes of a class-imbalanced dataset. The loss of these features hinders the performance of the trained classifier ensemble.

In addition to high dimensionality, the complexity of human activity recognition often results in uncertainty in the decision boundaries of machine learning models (i.e., classifiers). Without an adaptive and modular approach to model selection, the performance of a late fusion method using an ensemble of classifiers is highly dependent on the complexity of the classifier. Adaptive fusion approaches can make use of simpler classifiers and, instead, weigh the classifiers that perform better on certain class labels higher than the others. The modularity of a fusion method helps classify new classes with ease by only adding classifiers trained on the new data to the ensemble, without having to retrain existing models. Uncertainty in the decision of a classifier can stem from missing sensor data, sporadic measurements, or low confidence values. This can lead to high variability in model prediction performance. Properly managing uncertainty in the collected data and models can avoid unnecessary retraining of classifiers. To make an activity or context inference using uncertain measurements or features, various probabilistic assignments can be used. For example, Bayesian networks trained on real data assign probabilistic values to different nodes, which are later used to infer new contexts [23]. Fuzzy logic and Dempster-Shafer theory (DST) can also be used to handle uncertainty. While these theoretic frameworks are capable of handling epistemic model uncertainty, methods built using them tend not to be adaptive and modular [24], [25].

The handling of imbalanced, or skewed, data poses another challenge in activity recognition, especially in the model creation stage. This can result in further uncertainty in a classifier's decision boundary. The problem of data skewness arises when there is a considerable amount of imbalance among the classes. This issue can worsen if positive examples are inherently of more importance. Human activity recognition datasets (e.g., [26]) are usually imbalanced when capturing in-the-wild human behavior. Using datasets with unequal samples between classes in multi-class classification problems introduces class bias in the classifier training stage.

Insights. To overcome the aforementioned challenges present in human activity recognition, we turn to a different adaptive late fusion framework called Naive Adaptive Probabilistic (NAPS) Fusion [27]. NAPS Fusion circumvents the disadvantages of traditional dimensionality reduction techniques by creating a multitude of reduced feature datasets that span the original feature space. This addresses the problem of large feature spaces without relying on other dimensionality reduction techniques. To handle model uncertainty, we leverage NAPS Fusion's sensor fusion approach, which relies on Dempster-Shafer's theory (DST) of evidence [28], [29]. Uncertainty reduction methods using DST have been shown to perform well when combining information from multiple sensors [30], [31]. To complement the DST-based fusion, NAPS Fusion uses bootstrap aggregation in the model creation stage to reduce classifier variance. Additionally, NAPS Fusion uses Synthetic Minority Over-sampling Technique (SMOTE) [32] to equalize minority and majority class imbalance and reduce classifier bias.

Contributions. Through the use NAPS Fusion, we provide a new approach to activity recognition that addresses some of the key challenges present in the field. The following list summarizes the contributions of this work:

- We demonstrate the successful translation of NAPS Fusion, a late fusion method with adaptive model selection, to a 4-class in-the-wild human activity recognition problem, extending its initial range of applications.
- We show that synthetic oversampling and simple, binary classifiers trained on reduced feature sets can obtain comparable or improved classification performance compared to the LFL and LFA benchmark late fusion methods.
- The Dempster-Shafer theoretic fusion framework in NAPS Fusion is capable of handling epistemic uncertainty in a classifier's activity classification, providing an overall improvement in classification performance.

Our methodology was validated on the UCSD ExtraSensory dataset [20]. This dataset provides a suitable testbed for activity recognition in-the-wild using noninvasive sensors in wearable devices. We compare the results of NAPS Fusion against LFL and LFA fusion to demonstrate the improvements of NAPS Fusion over similar in-the-wild fusion methods.

II. Methods

The NAPS Fusion method proposed in [27] for fusing information from a variety of sources of information, sensors in our case, is based on Dempster-Shafer theory (DST) [29]. Dempster-Shafer theory (DST) is regarded as a generalization of probability theory that takes into consideration possibly conflicting sources of information. Uncertainty is the central concept in DST that quantifies the degree of certainty or ignorance in our decisions. Belief and plausibility are other concepts in DST that measure the minimum and maximum certitude about a decision. DST provides a framework for combining multiple, possibly conflicting, *bodies of evidence* to make a decision. With DST, problems such as specifying priors can be avoided.

An overview of the entire NAPS Fusion framework and its components is shown in Fig. 1. In this section, we break down the most important components in the framework to explain how NAPS Fusion addresses the following challenges in activity recognition: 1) high dimensionality, 2) multi-class classification, 3) class imbalance, and 4) model uncertainty. Although a more in-depth review of DST, tailored to sensor fusion, is given after the description of the Adaptive Probabilistic Sensor shown in Fig. 1, some DST concepts cannot be avoided and are introduced beforehand to properly understand NAPS Fusion. At the end of this section, we describe our application of NAPS Fusion on the ExtraSensory dataset.

A. Naive Adaptive Probabilistic Sensor

1) High dimensionality: Given an original feature space from a dataset

$$\mathcal{S} = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1L} & C_1 \\ x_{21} & x_{22} & \dots & x_{2L} & C_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ x_{N1} & x_{N2} & \dots & x_{NL} & C_N \end{bmatrix} \tag{1}$$

with L physiological or physical features, N samples, and corresponding activity/class labels C_n, NAPS Fusion randomly selects a reduced number of features from $\mathcal{S}$ to reduce the dimensionality of the feature space. More precisely, a data structure D_k is defined within the framework with a reduced number of features $L' << L$. These features are quasi-randomly drawn, meaning that each sensor is guaranteed a certain number of features to be drawn from it. This selection process is repeated K times to ensure that the reduced feature sets $\{D_1, D_2, \dots, D_K\}$ randomly span the entire feature space. It is important to note that the features are taken from M sensors, where each i-th sensor provides a total of l_i features, such that $L = \sum_{i=1}^{M} l_i$.

2) Multi-class classification: NAPS Fusion is able to handle multi-class classification problems by treating each class label as a proposition ω_i within the *frame of discernment (FoD)* Ω of the DST framework. DST requires that the propositions in Ω be mutually exclusive and exhaustive. The power set $\mathcal{P}(\Omega)$ consists of all subsets of Ω. An element in $\mathcal{P}(\Omega)$ is called an augmented class or also a proposition. Support can be assigned to each augmented class in $\mathcal{P}(\Omega)$. Excluding $\emptyset$ and Ω, different augmented classes can be selected from $\mathcal{P}(\Omega)$ to train classifiers. As an example of the augmented classes that can be generated, given a 4-class problem with $\Omega = \{\omega_1, \omega_2, \omega_3, \omega_4\}$, Table I shows the proposition combinations $\mathbf{C}_p$ for 4-class and 2-class (binary) classifiers. In total, there are 14 valid proposition combinations in a 4-class problem.

TABLE I
Augmented Classes Created in a 4-Class Problem

Proposition Combination	Augmented Classes in Combination	Number of Classes
$\mathbf{C}_1$	$\{\omega_1\}$, $\{\omega_2\}$, $\{\omega_3\}$, $\{\omega_4\}$	4
$\mathbf{C}_2$	$\{\omega_1\}$, $\{\omega_2, \omega_3, \omega_4\}$	2
$\mathbf{C}_3$	$\{\omega_2\}$, $\{\omega_1, \omega_3, \omega_4\}$	2
$\mathbf{C}_4$	$\{\omega_3\}$, $\{\omega_1, \omega_2, \omega_4\}$	2
$\mathbf{C}_5$	$\{\omega_4\}$, $\{\omega_2, \omega_3, \omega_4\}$	2
$\mathbf{C}_6$	$\{\omega_1, \omega_2\}$, $\{\omega_3, \omega_4\}$	2
$\mathbf{C}_7$	$\{\omega_1, \omega_3\}$, $\{\omega_2, \omega_4\}$	2
$\mathbf{C}_8$	$\{\omega_1, \omega_4\}$, $\{\omega_2, \omega_3\}$	2

Given a recorded response (i.e., class label) $C_i \in \Omega$ for a sample in $\mathcal{S}$, the positive class label $\{\cdot\}^+$ is defined as an augmented class $A \in \mathbf{C}_p$ such that $C_i \in A$. In other words, the positive class for a sample i is the augmented class in $\mathbf{C}_p$ containing the original sample label C_i. The set of all other classes in $\mathbf{C}_p$, the negative classes, is denoted as $\{\cdot\}^-$. Thus, $\mathbf{C}_p \stackrel{\text{def}}{=} \{\{\cdot\}^+, \{\cdot\}^-\}$. Using the classes in $\mathbf{C}_p$, the augmentation of the dataset D_k with the positive class results in the creation of the augmented response data structure D_{pk}, which forms part of the *Augmented Class and ML Model* subcomponent in Fig. 1. The class labels in $\mathbf{C}_p$ become the

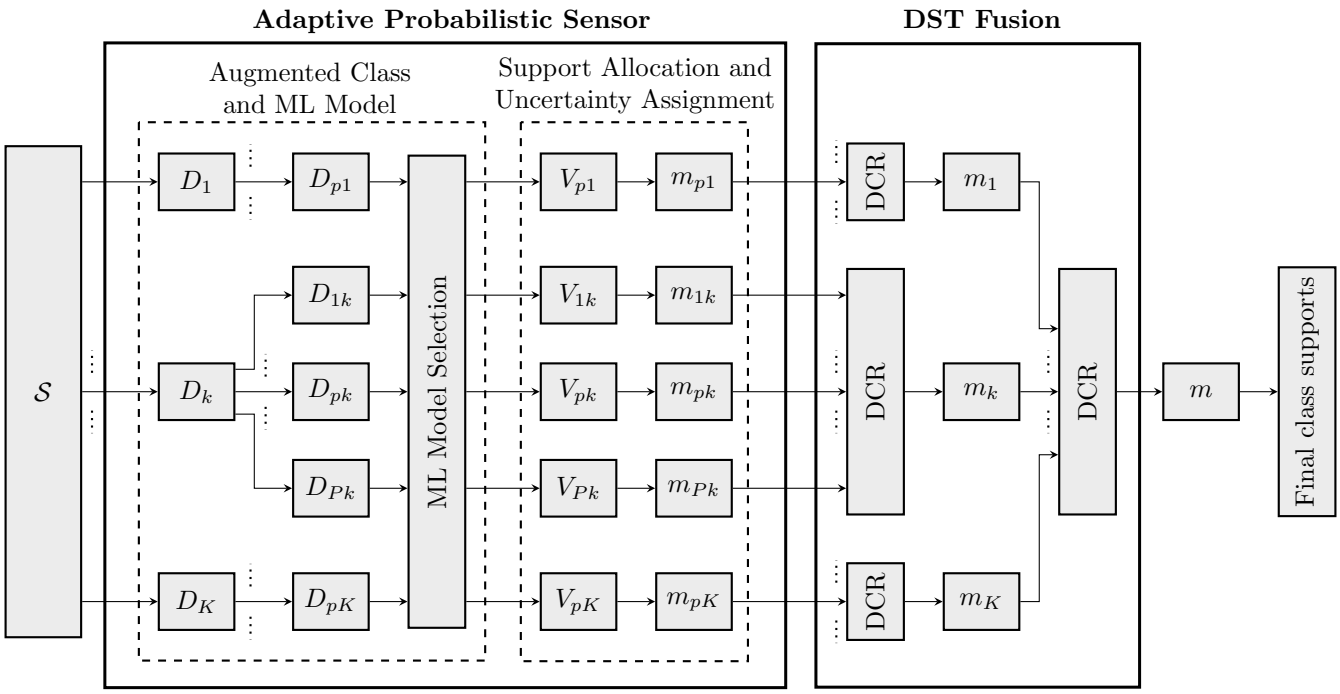

Fig. 1. Overview of the NAPS Fusion framework. The framework is broken down into two major components: the *naive adaptive probabilistic sensor* component, which structures uncertain sensor information prior to fusion, and the *DST fusion* component, which relies on DST to fuse bodies of evidence.

new class labels that the machine learning (ML) models are trained on. Due to this augmentation of classes, NAPS Fusion can simplify an intricate, multi-class classification problem by using classifiers trained on fewer classes. Moreover, NAPS Fusion's ability to handle combinations of singleton propositions (e.g., $\{\omega_1, \omega_2\} \in \mathbf{C}_6$) allow it to potentially form less complex decision boundaries compared to, for example, one vs. all classifiers. Uncertainty in the exact class is resolved in the fusion stage. Comparison fusion methods implemented in this paper strictly use binary classifiers trained on one vs. all proposition combinations $\mathbf{C}_2$–$\mathbf{C}_5$, whereas NAPS Fusion uses $\mathbf{C}_2$–$\mathbf{C}_8$.

3) Class imbalance: Using the augmented response D_{pk}, an ML model M_{pk} is created. The fusion stage, discussed later on, is amenable to ML models that provide a probability, distance metric, or vote as an output. Thus, we chose to work with logistic regression classifiers for the models. To improve the performance of M_{pk} and reduce the variance, bootstrap aggregating (bagging) is used on each model to create a multitude of bags or micro-models. More importantly, Synthetic Minority Over-sampling Technique (SMOTE) [32] is used to reduce the class imbalance that is further exacerbated by the introduction of the augmented class labels.

4) Model uncertainty: In addition to the class imbalance reduction that SMOTE provides, SMOTE reduces model uncertainty introduced by lack of density in the input feature space by clustering new feature samples using a k-nearest neighbors approach. Moreover, NAPS performs model selection on the $P \times K$ ML models to find the models with the lowest uncertainty (i.e., the strongest predictors) per proposition combination. This process of model selection results in the adaptive aspect of NAPS, as samples from the original dataset are routed through different ML models prior to fusion. It is motivated by research demonstrating that different sensors, and feature subspaces by implication, are more capable of predicting certain activities [33].

After the model selection process, additional model uncertainty is encapsulated in the support assigned to an augmented class through the use of a voting system. For each augmented class, the number of bags (micro-models) that "voted" for the augmented class is normalized by the total number of bags. These votes are placed in the support data structure V_{pk}, where zero support is given to augmented classes not in the positive class $\{\cdot\}^+$ or negative class $\{\cdot\}^-$. Further model uncertainty is handled by DST. The augmented classes in the support structure V_{pk} are assigned a mass through the use of a mass assignment function $m_{pk}(\cdot)$ that forms part of DST. Through the calculation of these masses, the *DST Fusion* component of NAPS Fusion generates the final supports for a data sample. The definition of the mass assignment function and other DST concepts are provided in the following subsection.

B. DST Fusion

Each classification model derived from a proposition combination in Table I contributes to the decision-making process in NAPS Fusion. The goal of DST fusion is to bring together the information provided by the "sensors" (i.e., ML models) that form part of the *Adaptive Probablistic Sensor*. This is

done by fusing the support information provided by the ML models trained on different feature subspaces. At the end of fusion process, the supports given to the original class labels are used to assign a final class label to a data sample. We provide a gentle exposition into Dempster-Shafer theory to explain the *DST Fusion* component in Fig. 1.

DST starts by assuming a frame of discernment (FoD) Ω, a concept previously discussed. A mass is assigned to each element in $\mathcal{P}(\Omega)$ with a function $m : \mathcal{P}(\Omega) \longmapsto [0, 1]$. This function is known as the *basic probability assignment*, or simply the *mass assignment*. If the subset is assigned a non-zero mass, then it is called a *focal element*. The mass assignment function satisfies the following properties:

$$m(\emptyset) = 0; \quad \sum_{A \subseteq \mathcal{P}(\Omega)} m(A) = 1$$

The mass assignment function used prior to the DST fusion stage is the following:

$$m_{pk}(A) = \begin{cases} \dfrac{1 - e^{-\Theta/D}}{1 - e^{-1/D}}, & \text{for } A = \Omega, \\ 0, & \text{for } A = \emptyset, \\ \dfrac{T_i}{T_{tot}}(1 - m_{pk}(\Omega)), & \text{otherwise,} \end{cases} \tag{2}$$

where $D = (e - 1)^e$ and Θ is the uncertainty associated with the classifier voting strategy. It is calculated as

$$\Theta = 1 - \frac{\sqrt{\left(\dfrac{T_1}{T_{tot}} - \dfrac{1}{C_{tot}}\right)^2 + \cdots + \left(\dfrac{T_{C_{tot}}}{T_{tot}} - \dfrac{1}{C_{tot}}\right)^2}}{\sqrt{\dfrac{C_{tot} - 1}{C_{tot}}}} \tag{3}$$

Here, C_{tot} represents total number of augmented classes in the proposition combination, T_{tot} is the total number of votes (bags), and T_i is the number of votes for augmented class i. Complete ignorance (i.e., maximum uncertainty) of the true augmented class occurs when for all i, $T_i = T_{tot}/C_{tot}$. This results in an uncertainty value of $\Theta = 1$ and a corresponding mass assignment of $m_{pk}(\Omega) = 1$. Minimum uncertainty (i.e., $\Theta = 0, m_{pk}(\Omega) = 0$) occurs when for some i, $T_i = T_{tot}$.

Dempster-Shafer theory allows the combination of mass assignments with the use of Dempster's Combination Rule (DCR) [29].

$$m(A) = \frac{\sum\limits_{B \cap C = A \neq \emptyset} m_1(B)\, m_2(C)}{1 - \sum\limits_{B \cap C = \emptyset} m_1(B)\, m_2(C)} \tag{4}$$

Here, $(A, B, C) \subseteq \Omega$ and m_1, m_2 are any m_{pk} and $m_{p'k'}$, where $p \neq p', k \neq k'$. The denominator and the numerator represent the conflict between the sets and the cumulative confirmation from the sets to support proposition A respectively. DCR allows the support information provided by all ML models to be fused together, two models at a time.

C. NAPS Fusion for Activity Recognition

We applied NAPS Fusion on the benchmark UCSD ExtraSensory dataset [20]. The ExtraSensory dataset is a feature rich and publicly available dataset with 51 diverse context labels and 225 different features across 10 sensing modalities. There are 300,000 activity samples from 60 participants collected over the course of a week. The dataset contains data from multiple smartphone sensing modalities such as accelerometer, gyroscope, magnetometer, audio, location, and phone-state, as well as accelerometer data from an additional smartwatch. Rather than providing raw sensor measurements, the dataset instead provides features derived from the raw measurements at 1-second intervals. On average, each sensor provides about 23 features.

The ExtraSensory dataset has a total of six mutually exclusive activities: lying down, walking, sitting, running, bicycling and standing. From these six activities, only the first four activities are explored in this study, that is, lying down, walking, sitting, and running. Our focus on only four activities helps reduce the exponential computational complexity penalty that is incurred in the fusion stage due to DCR [34]. The following sections break down important components of the experimental design.

1) Developing feature sets: The features selected from 6 out of the 10 sensors in the ExtraSensory dataset are shown in Table II. These same sensors are used in the study that introduces the LFA and LFL late fusion methods [20]—the methods this study compares against. The abbreviated labels for the six sensors are as follows: Acc = phone accelerometer, Gyro = phone gyroscope, Aud = phone audio, Loc = phone location, PS = phone state, and WAcc = watch accelerometer. Alone, these six sensors capture nearly 80% of the total number of features. Examples of features found in the dataset include mean and standard deviation of the accelerometer readings, phone state (active or inactive), and minimum and maximum speed of the user obtained from location measurements. More information on the features in the dataset can be found in [20] and at the dataset's website[1].

TABLE II
UCSD EXTRASENSORY DATASET FEATURES

Acc	Gyro	Aud	Loc	PS	WAcc	Total
26	26	26	17	34	46	175

The LFA and LFL fusion methods use all of a sensor's available features during model training (e.g., all 26 features in Acc). For NAPS Fusion, we randomly sampled 10% of features from each sensor, rounding down to the nearest integer value where necessary. This resulted in a total of 14 sampled features. These subsets of the original feature space $\mathcal{S}$ form the reduced feature sets D_k shown in Fig. 1. For the ExtraSensory dataset, $K = 200$ reduced feature sets were created to randomly span $\mathcal{S}$. The comparison late fusion methods, LFA

[1]ExtraSensory Dataset website: http://extrasensory.ucsd.edu/

and LFL, create a single logistic regression model per sensor in Table II, trained on all sensor features. NAPS Fusion instead uses a logistic regression bagging approach per reduced feature set D_k.

2) Augmenting response variables: Each of the original activity labels forms a proposition $\omega_i \in \Omega$. Prior to classifier training, the $P = 7$ two-class proposition combinations $\mathbf{C}_2$ through $\mathbf{C}_8$ in Table I are selected. The positive and negative augmented classes in these proposition combinations result in the creation of the augmented class datasets D_{pk}. Thus, prior to model training, $200 \times 7 = 1400$ augmented class datasets are created from the $K = 200$ reduced feature sets D_k and $P = 7$ two-class proposition combinations.

3) Model training using SMOTE and bagging: The augmentation of the class variables can exacerbate the class imbalance problem already present in the original dataset. To alleviate this, SMOTE creates synthetic instances of minority classes in order to balance the majority and minority classes. In order to reduce the variability in the model prediction, bagging was used to create $T_{tot} = 200$ bags by sampling 60% of the training data with replacement.

4) Model uncertainty calculation: With a multitude of logistic regression classifiers created, the strongest predictors are kept and used in the ML model selection stage. Before choosing the predictors, a normalized bagging voting approach is used to calculate the proportion of votes given to augmented classes by each classifier. Eq. (3) is used to calculate the uncertainty in the classifier votes by taking into account the amount of votes for each augmented class.

5) Model selection and fusion: With an uncertainty value associated to each model, we selected the top performing models in the ML Model Selection stage, shown in Fig. 1, and used them to perform the classification task. The top performing models for each combination in $\mathbf{C}_2$ through $\mathbf{C}_8$ (7 total combinations) were the 6 with the lowest uncertainty value Θ. This resulted in the selection $7 \times 6 = 42$ classifiers out of the original $1,400$. After the initial uncertainty is assigned to a test data sample using the mass assignment function m_{pk}, DST fusion combines all other mass assignments using DCR and produces the final supports for the singleton augmented classes.

III. Results & Discussion

To compare against NAPS Fusion, we implemented the late fusion methods LFA and LFL, and evaluated a 5-fold cross validation performance. We selected and measured the following performance metrics: precision, recall, specificity, F1 score, and balanced accuracy. The four classified activities were lying down, sitting, walking, and running. The results of the performance metrics are summarized in Table III. The "% Change NAPS" column in the table shows the percent gain or drop between the highest value of either of the compared late fusion methods and NAPS Fusion. Note that the values for the compared late fusion methods were from our implementation of the published methods. These implementations on our 4-class problem demonstrated similar performance metrics values with those reported in [20].

TABLE III
Performance Metric Results of the Tested Fusion Methods (Best Results are Bolded)

Performance Metric	Class	Sensor Fusion Method			% Change NAPS
		LFA	LFL	NAPS	
Precision	Lying down	0.70	0.77	**0.89**	**16%**
	Sitting	0.65	0.64	**0.83**	**28%**
	Walking	0.28	0.26	**0.58**	**110%**
	Running	0.02	0.03	**0.61**	**1900%**
Recall	Lying down	**0.91**	0.88	0.88	-3.3%
	Sitting	**0.86**	0.80	0.77	-10%
	Walking	**0.90**	0.81	0.77	-14%
	Running	**0.83**	0.73	0.58	-30%
Specificity	Lying down	0.80	0.86	**0.91**	**5.8%**
	Sitting	0.70	0.71	**0.82**	**15%**
	Walking	0.74	0.74	**0.79**	**6.7%**
	Running	0.51	**0.74**	**0.74**	0%
Balanced Accuracy	Lying down	0.85	0.87	**0.90**	**2.9%**
	Sitting	0.78	0.75	**0.79**	**1.9%**
	Walking	**0.82**	0.77	0.78	-4.9%
	Running	0.67	**0.74**	0.66	-10%
F1 Score	Lying down	0.79	0.82	**0.88**	**8%**
	Sitting	0.74	0.71	**0.80**	**8%**
	Walking	0.43	0.39	**0.66**	**55%**
	Running	0.03	0.06	**0.59**	**930%**

In all the four classified activities, NAPS Fusion resulted in higher precision values than the UCSD late fusion methods. Similar results are present in the other performance metrics, except for recall. We hypothesize that the lower recall of NAPS fusion, especially in walking and running, is due dataset balancing resulting from the synthetic oversampling of SMOTE. The use of synthetic oversampling may have introduced synthetic samples close to false negative samples that were detrimental to the performance of NAPS Fusion and LFL. The simplicity in the weighting approach of LFA, simple averaging, may have given it an advantage over the more complex weighting approach of NAPS Fusion and, to an extent, LFL.

In spite of the recall results, with NAPS Fusion we obtained higher F1 scores, shown in Table III, striking a good balance between precision and recall to decrease the miss-classification rate. Here we note the severe underperformance of LFA and LFL in the F1 scores of walking and running, where NAPS Fusion outperforms LFA in running by 56%. These results are consistent with those presented in [20], where running, in particular, is a difficult activity to classify. In the F1 score of running, we also observe the biggest percent change increase, 930%, between NAPS Fusion and LFL.

Additionally, the high specificity of NAPS Fusion highlights its ability to correctly identify the majority of the negative samples across all labels, thereby decreasing the occurrence of false positives. In balanced accuracy, a metric that takes into account both specificity and recall, we observed comparable performance between NAPS and LFL and LFA, with the biggest percent drop in performance, -10%, being in running.

The overall comparable performance demonstrates that despite the lower recall values of NAPS Fusion on all activities, NAPS Fusion is able to maintain comparable or better performance across the various performance metrics.

IV. Conclusion

Human activity recognition has the potential to greatly improve our daily lives through applications like automated caregiving (e.g., home-based rehabilitation) and enhanced health and fitness activity tracking. However, imbalanced datasets, large feature spaces, multi-class problems, and uncertainty in the decision boundaries of models, make in-the-wild human activity recognition a challenging task. Our evaluation of the NAPS Fusion framework shows that the use of Dempster-Shafer theory-based sensor fusion approach provides a significant step in the right direction to solve these challenges. The method outperformed related decision level fusion techniques in nearly all the performance metrics we tested. Furthermore, we managed strike a balance between precision, recall, and specificity through higher F1 score and balanced accuracy values across all activities. We are the first to show that the NAPS Fusion framework is able to provide an improvement in precision in multi-class activity recognition problems through its class balancing, model creation on reduced feature sets and augmented classes, and use of Dempster-Shafer theory to resolve model (e.g., binary classifier) uncertainty.

Future work will address the computational complexity of Dempster-Shafer theory, which can pose problems in classification tasks with more than five to six classes. To remedy this, we intend to explore other combination rules for the model fusion stage and use distributed computing approaches. Additionally, the potential applications of NAPS Fusion extend beyond human activity recognition. Future applications include pilot physiological monitoring to improve our work on cognitive state detection [35], [36]. These applications provide a suitable testbed for context recognition in highly demanding psychophysiological workload environments.

References

[1] K. Stawarz, A. L. Cox, and A. Blandford, "Don't forget your pill!: Designing effective medication reminder apps that support users' daily routines," in *Proc. 32nd annual ACM conference on Human factors in computing systems*. ACM, 2014, pp. 2269–2278.

[2] N. Napoli, A. Harrivel, and A. Raz, "Improving physiological monitoring sensor systems for pilots," *Aerospace America*, vol. 12, 2020.

[3] C. Stephens, K. Kennedy, N. Napoli, M. Demas, L. Barnes, B. Crook, R. Williams, M. C. Last, and P. Schutte, "Effects on task performance and psychophysiological measures of performance during normobaric hypoxia exposure," in *19th International Symposium on Aviation Psychology*, 2017, p. 202.

[4] J. Dahmen, B. L. Thomas, D. J. Cook, and X. Wang, "Activity learning as a foundation for security monitoring in smart homes," *Sensors*, vol. 17, no. 4, p. 737, 2017.

[5] N. H. Goddard, "Human activity recognition," in *Motion-Based Recognition*. Springer, 1997, pp. 147–170.

[6] J. Zhu, X. Liu, Q. Shi, T. He, Z. Sun, X. Guo, W. Liu, O. B. Sulaiman, B. Dong, and C. Lee, "Development trends and perspectives of future sensors and MEMS/NEMS," *Micromachines*, vol. 11, no. 1, p. 7, 2020.

[7] M. L. Cheung, K. Y. Chau, M. H. S. Lam, G. Tse, K. Y. Ho, S. W. Flint, D. R. Broom, E. K. H. Tso, and K. Y. Lee, "Examining consumers' adoption of wearable healthcare technology: The role of health attributes," *International journal of environmental research and public health*, vol. 16, no. 13, p. 2257, 2019.

[8] I. El Moudden, S. ElBernoussi, and B. Benyacoub, "Modeling human activity recognition by dimensionality reduction approach," in *Proc. 27th International Business Information Management Association Conference–Innovation Management and Education Excellence Vision*, vol. 2020, 2016, pp. 1800–1805.

[9] K. H. Walse, R. V. Dharaskar, and V. M. Thakare, "PCA based optimal ANN classifiers for human activity recognition using mobile sensors data," in *Proc. First International Conference on Information and Communication Technology for Intelligent Systems*, 2016, pp. 429–436.

[10] Y. Bengio, "Deep learning of representations: Looking forward," in *Proc. International conference on statistical language and speech processing*, 2013, pp. 1–37.

[11] J. Wang, Y. Chen, S. Hao, X. Peng, and L. Hu, "Deep learning for sensor-based activity recognition: A survey," *Pattern Recognition Letters*, vol. 119, pp. 3–11, 2019.

[12] M. Z. Uddin and A. Soylu, "Human activity recognition using wearable sensors, discriminant analysis, and long short-term memory-based neural structured learning," *Scientific Reports*, vol. 11, no. 1, p. 16455, 2021.

[13] Y.-S. Lee and S.-B. Cho, "Activity recognition using hierarchical hidden markov models on a smartphone with 3D accelerometer," in *Proc. International conference on hybrid artificial intelligence systems*. Springer, 2011, pp. 460–467.

[14] L. Chen, J. Hoey, C. D. Nugent, D. J. Cook, and Z. Yu, "Sensor-based activity recognition," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, no. 6, pp. 790–808, 2012.

[15] J. Yin, Q. Yang, and J. J. Pan, "Sensor-based abnormal human-activity detection," *IEEE Transactions on Knowledge and Data Engineering*, vol. 20, no. 8, pp. 1082–1090, 2008.

[16] B. J. Mortazavi, M. Pourhomayoun, G. Alsheikh, N. Alshurafa, S. I. Lee, and M. Sarrafzadeh, "Determining the single best axis for exercise repetition recognition and counting on smartwatches," in *Proc. 2014 11th International Conference on Wearable and Implantable Body Sensor Networks*. IEEE, 2014, pp. 33–38.

[17] S. Aziz, M. U. Khan, A. Zahoor, and S. Z. H. Naqvi, "Intelligent system for human context recognition," in *Proc. 2020 International Conference on Computing and Information Technology (ICCIT-1441)*, 2020, pp. 1–5.

[18] N. Gaw, S. Yousefi, and M. R. Gahrooei, "Multimodal data fusion for systems improvement: A review," *IISE Transactions*, vol. 54, no. 11, pp. 1098–1116, Oct. 2021.

[19] G.-Z. Yang and G. Yang, *Body sensor networks*. Springer, 2006, vol. 1.

[20] Y. Vaizman, K. Ellis, and G. Lanckriet, "Recognizing detailed human context in the wild from smartphones and smartwatches," *IEEE Pervasive Computing*, vol. 16, no. 4, pp. 62–74, 2017.

[21] R. Damaševičius, M. Vasiljevas, J. Šalkevičius, and M. Woźniak, "Human activity recognition in aal environments using random projections," *Computational and mathematical methods in medicine*, vol. 2016, 2016.

[22] A. N. Gorban, B. Kégl, D. C. Wunsch, A. Y. Zinovyev *et al.*, *Principal manifolds for data visualization and dimension reduction*. Springer, 2008, vol. 58.

[23] T. Gu, H. K. Pung, D. Q. Zhang, H. K. Pung, and D. Q. Zhang, *A Bayesian approach for dealing with uncertain contexts*, 2004.

[24] M. A. Lopez Medina, M. Espinilla, C. Paggeti, and J. Medina Quero, "Activity recognition for iot devices using fuzzy spatio-temporal features as environmental sensor fusion," *Sensors*, vol. 19, no. 16, p. 3512, 2019.

[25] F. Sebbak, F. Benhammadi, A. Chibani, Y. Amirat, and A. Mokhtari, "Dempster–shafer theory-based human activity recognition in smart home environments," *annals of telecommunications-annales des télécommunications*, vol. 69, pp. 171–184, 2014.

[26] D. Anguita, A. Ghio, L. Oneto, X. Parra, and J. L. Reyes-Ortiz, "A public domain dataset for human activity recognition using smartphones." in *Proc. European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN)*, 2013.

[27] N. J. Napoli, C. L. Stephens, K. D. Kennedy, L. E. Barnes, E. Juarez Garcia, and A. R. Harrivel, "NAPS Fusion: A framework to overcome experimental data limitations to predict human performance and cognitive task outcomes," *Information Fusion*, vol. 91, pp. 15–30, 2023.

[28] A. P. Dempster, "Upper and lower probabilities induced by a multivalued mapping," in *Classic Works of the Dempster-Shafer Theory of Belief Functions*. Springer, 2008, pp. 57–72.
[29] G. Shafer, *A mathematical theory of evidence*. Princeton University Press, 1976, vol. 42.
[30] N. Nesa and I. Banerjee, "IoT-based sensor data fusion for occupancy sensing using Dempster–Shafer evidence theory for smart buildings," *IEEE Internet of Things Journal*, vol. 4, no. 5, pp. 1563–1570, 2017.
[31] N. J. Napoli and L. E. Barnes, "A Dempster-Shafer approach for corrupted electrocardiograms signals," *Proc. 29th International Florida Artificial Intelligence Research Society Conference*, 2016.
[32] H. He, Y. Bai, E. A. Garcia, and S. Li, "ADASYN: Adaptive synthetic sampling approach for imbalanced learning," in *Proc. 2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, 2008, pp. 1322–1328.
[33] G. M. Weiss, J. L. Timko, C. M. Gallagher, K. Yoneda, and A. J. Schreiber, "Smartwatch-based activity recognition: A machine learning approach," in *Proc. 2016 IEEE-EMBS International Conference on Biomedical and Health Informatics (BHI)*. IEEE, 2016, pp. 426–429.
[34] M. Benalla, B. Achchab, and H. Hrimech, "On the computational complexity of dempster's rule of combination, a parallel computing approach," *Journal of Computational Science*, vol. 50, p. 101283, 2021.
[35] N. Napoli, S. Adams, A. R. Harrivel, C. Stephens, K. Kennedy, M. Paliwal, and W. Scherer, "Exploring cognitive states: Temporal methods for detecting and characterizing physiological fingerprints," in *AIAA Scitech 2020 Forum*, 2020, p. 1193.
[36] N. J. Napoli, M. Demas, C. L. Stephens, K. D. Kennedy, A. R. Harrivel, L. E. Barnes, and A. T. Pope, "Activation complexity: A cognitive impairment tool for characterizing neuro-isolation," *Scientific Reports*, vol. 10, no. 1, pp. 1–20, 2020.