MIS Quarterly

RESEARCH ARTICLE

A DEEP LEARNING APPROACH FOR RECOGNIZING ACTIVITY OF DAILY LIVING (ADL) FOR SENIOR CARE: EXPLOITING INTERACTION DEPENDENCY AND TEMPORAL PATTERNS[1]

Hongyi Zhu
Department of Information Systems and Cyber Security, University of Texas at San Antonio,
San Antonio, TX, U.S.A. {hongyi.zhu@utsa.edu}

Sagar Samtani
Department of Operations and Decision Technologies, Indiana University,
Bloomington, IN, U.S.A. {ssamtani@iu.edu}

Randall A. Brown
Department of Pharmacy, University of Arizona,
Tucson, AZ, U.S.A. {randallb9@gmail.com}

Hsinchun Chen
Department of Management Information Systems, University of Arizona,
Tucson, AZ, U.S.A. {hchen@eller.arizona.edu}

Ensuring the health and safety of senior citizens who live alone is a growing societal concern. The Activity of Daily Living (ADL) approach is a common means to monitor disease progression and the ability of these individuals to care for themselves. However, the prevailing sensor-based ADL monitoring systems primarily rely on wearable motion sensors, capture insufficient information for accurate ADL recognition, and do not provide a comprehensive understanding of ADLs at different granularities. Current healthcare IS and mobile analytics research focuses on studying the system, device, and provided services, and is in need of an end-to-end solution to comprehensively recognize ADLs based on mobile sensor data. This study adopts the design science paradigm and employs advanced deep learning algorithms to develop a novel hierarchical, multiphase ADL recognition framework to model ADLs at different granularities. We propose a novel 2D interaction kernel for convolutional neural networks to leverage interactions between human and object motion sensors. We rigorously evaluate each proposed module and the entire framework against state-of-the-art benchmarks (e.g., support vector machines, DeepConvLSTM, hidden Markov models, and topic-modeling-based ADLR) on two real-life motion sensor datasets that consist of ADLs at varying granularities: Opportunity and INTER. Results and a case study demonstrate that our framework can recognize ADLs at different levels more accurately. We discuss how stakeholders can further benefit from our proposed framework. Beyond demonstrating practical utility, we discuss contributions to the IS knowledge base for future design science-based cybersecurity, healthcare, and mobile analytics applications.

Keywords: Activity of Daily Living recognition, deep learning, human-object interaction, 2D interaction kernel, convolutional neural networks, sequence-to-sequence model, design science

[1] Gedaminas Adomavicius was the accepting senior editor for this paper. Pedro Ferreira served as the associate editor. This research was funded by NSF awards IIP-1622788, IIP-1417181, CNS-1850362, and OAC-1917117.

DOI:10.25300/MISQ/2021/15574

Introduction

Medical advances and increased healthcare accessibility have enabled an increasing life expectancy (World Health Organization 2016). In 2016, 49.2 million U.S. citizens (15.2%) were over 65 years old (i.e., senior citizens) (Census Bureau 2017). The American Community Survey estimates indicate that 79.2% of U.S. senior citizens live independently (Census Bureau 2016). Chronic conditions, frailty, dementia, and other health concerns or diseases can impact the health, safety, and quality of life of senior citizens who live alone. Ensuring health and safety for these senior citizens is a growing societal concern.

Researchers and practitioners often use the Activity of Daily Living (ADL) approach to monitor the self-care ability, health status, and disease progression (Foti and Koketsu 2013) of senior citizens who live alone. Basic ADLs engage simple physical movements (e.g., self-feeding and toilet hygiene) (Katz 1983). Instrumental ADLs involve more cognitively complex tasks, such as preparing meals, taking prescribed medications, and shopping (Hardy 2014). Conventional approaches to monitoring ADL performance include clinical exams (Katz 1983; Singh et al. 2014), home care services (Whitehead et al. 2015), and self-reported activities at home (Chung et al. 2017). Clinical exams provide the most direct assessment. However, because of their infrequency (e.g., monthly), they are not entirely suitable for early intervention and preventive care. Home care is timelier but is not always affordable (Genworth 2017). Self-reported data (e.g., activity diary) is affordable but lacks reliability if cognitive function is deteriorating.

The limitations of current approaches have motivated clinicians to leverage remote home ADL monitoring systems with cameras, environment sensors, and motion sensors, which use modern sensing technologies to objectively record activities in real time in order to implement timely interventions and treatments (Bravo et al. 2016; Silva et al. 2015). Researchers often prefer accelerometer- and gyroscope-based motion sensors because of their high levels of sensitivity, high data granularity, low set-up costs, and relative unobtrusiveness (Haghi et al. 2017). *Wearable* sensors are attached to one's chest, hip, and/or wrists. *Object* sensors are deployed on commonly used household items such as pillboxes, appliances, and doors. Each sensor generates 10 to 100 data points per second (i.e., 10-100 Hz). Deep learning models, such as convolutional neural networks (CNNs), are the prevailing analytical approach. By simultaneously analyzing the signals of both wearable and object sensors, scholars can conduct ADL recognition (ADLR) at a high level (HL-ADL), mid-level (ML-ADL), and interaction level. Each level of granularity enables stakeholders to monitor selected physical (e.g., Parkinson's progression), mental (e.g., dementia resulting from Alzheimer's), and other health (e.g., medication nonadherence) progressions (Bryant et al. 2015; Jekel et al. 2015).

Despite their benefits over conventional ADLR approaches, current motion sensor-based systems have three key drawbacks. First, most systems deploy wearable sensors only. Thus, they cannot identify human-environment interactions. However, extracting such detail is often critical for clinicians to make timely interventions and diagnoses. Second, past ADLR literature applies standard deep learning models directly onto raw sensor signals. However, sensor signals contain complex cross-sensor, temporal, and axial dependencies, especially when using wearable and object sensors simultaneously. Identifying human-environment interactions and maximizing ADLR performance requires carefully designing an approach to extract the most salient representation for subsequent deep learning processing (Goodfellow et al. 2016; Li et al. 2018). Finally, extant models perform specific recognition tasks on a selected ADL level, preventing level-by-level ADLR and, in turn, a thorough understanding of a patient's physical and mental health progression at varying granularities (i.e., ADL levels). While information systems (IS) scholars are uniquely equipped to tackle these challenges, the existing health information technology (HIT) literature focuses on health IS (HIS), medical data use, and health data analytics that use traditional data sources (e.g., electronic health records). Motion sensor-based health analytics remains a nascent yet promising domain for IS scholars seeking to make a unique and positive societal impact.

In this research, we adopt the computational design science paradigm (Rai 2017) to develop a novel hierarchical, multiphase deep learning-based framework for ADLR. This framework has three key novelties. First, we leverage wearable and object motion sensors simultaneously to model ADLs. Second, we design a novel 2D interaction kernel for CNNs to capture human-object interactions. Third, we carefully design a level-by-level and end-to-end ADLR framework with interpretable intermediate results to analyze ADL patterns with different granularities. We rigorously evaluate the proposed framework and its constituent components against state-of-the-art feature engineering and deep learning models using two complementary datasets with different ADL granularities.

Table 1. Overview of Three ADL Levels

ADL Level	Description	Examples	Duration of ADL	Selected Practical Health Value and Applications
Activity (Instrumental; HL-ADL)	High-level motion sequences where a human interacts with *multiple* objects to realize a general motive	• Food preparation • Taking medication	>1 minute	Identify activity patterns for caretakers
Gesture (Basic; ML-ADL)	Middle-level motion sequences in which a human interacts with a *single* object to accomplish a particular goal	• Open/close a fridge door • Pick up / put down a pillbox	10-15 seconds	Identifying human-pillbox relationships to detect medication nonadherence
Interaction	Basic motion primitives. Context irrelevant, and characterized by relative movements between a human and an object	• Push, pull, pick up, put down, slide left, slide right	<10 seconds	Identify physical deterioration (e.g., Parkinson's progression)

The first dataset, Opportunity, is a publicly available morning activity dataset containing labels for complex activities (e.g., cleaning, making a sandwich) (Roggen et al. 2010). We also utilize SilverLink, a National Science Foundation (NSF)-funded novel multi-accelerometer activity monitoring system developed by the Artificial Intelligence (AI) Lab at the University of Arizona (Maimoon et al. 2016), to collect a dataset with labels for basic human-object interactions (e.g., pick up, pull). Apart from contributing to ADLR, our use of multiple sensor types, our 2D interaction kernel design, and our overall hierarchical framework follows design principles that can guide future cybersecurity, health, and mobile analytics research.

The remainder of this paper is organized as follows. First, we review healthcare IS, design science, ADLR, and deep learning literature to identify research gaps and propose research questions to explore. Second, we detail the major components of our research design. Subsequently, we present our results, contributions we make to the IS knowledge base, and practical implications for selected stakeholders. Finally, we discuss our findings and suggest promising future research directions.

Research Background

We review three literature streams: (1) healthcare IS literature and computational design science guidelines to inform and guide the development of a novel ADLR IT artifact, (2) motion sensor-based ADLR to gain knowledge on sensor signal data characteristics and identify prevailing ADLR methods, and (3) the state-of-the-art deep learning architectures for sensor data pattern recognition and sequence modeling.

Healthcare IS Literature and Computational Design Science Guidelines

The successful dissemination of IT into the healthcare industry has enabled IS scholars to make remarkable advances in three broad areas of healthcare information technology: health information systems (HIS), medical data use, and health data analytics. Table 2 summarizes selected literature in each category.

HIS and medical data usage studies employ behavioral theories and econometrics to explore health system adoption and investments, data sharing, security, and privacy. Despite their important contributions, the methods used in these studies cannot handle the volume and velocity of sensor data. Past health data analytics studies have adopted the design science paradigm for detecting hospital readmission, adverse events, and similar patients in social media and EHR contexts. Data analytics in mobile health contexts remains an understudied but societally relevant topic. Given that mobile IS literature has focused on web design (Adipat et al. 2011), service innovation (Kankanhalli et al. 2015; Ye and Kankanhalli 2018), and online service addiction (Kwon et al. 2016) to mobile devices in nonhealth contexts, a novel IT artifact designed for comprehensive ADL monitoring is critically needed. Such an artifact would be aligned at the intersection of health and mobile data analytics, potentially spearheading a new and promising area of IS research inquiry. Developing an IT artifact for advanced ADLR requires a careful approach to analyzing sensor data. The design science paradigm offers guidelines to systematically develop novel IT artifacts (e.g., constructs, models, methods, and instantiations) capable of solving salient business issues (Hevner et al. 2004).

Table 2. Summary of Selected Healthcare IS Literature

Category	Topic	Year	Study	Focus	Paradigm
HIS	Adoption & learning	2011	Mukhopadhyay et al.	The learning curve for physician referral systems	Economic
		2016	Venkatesh et al.	Adoption of eHealth Kiosk in India	Behavioral
	Investment	2015	Salge et al.	Mechanisms affecting HIS investment decisions	Economic
Medical data use	Sharing	2011	Ozdemir et al.	Incentives and switching cost for adopting and sharing EHR	Economic
		2017	Ayabakan et al.	Cost reduction by avoiding duplicate tests	Behavioral
		2018	Adjerid et al.	Reduce organizational expenditure with Health Information Exchange system	Economic
	Security	2014	Kwon and Johnson	Value of proactive security investments versus reactive investments	Economic
		2017	Angst et al.	Factors affecting data security technology adoption regarding healthcare data breaches	Behavioral
	Privacy	2011	Anderson and Agarwal	Individual's privacy boundary in the health context	Behavioral
		2017	Li and Qin	EHR data anonymization	Design Science
	Integration	2011	Oborn et al.	Feasibility of EHR integration in multidisciplinary care	Behavioral
Health data analytics	Social network based	2015	Yan et al.	Similar patient identification	Design Science
	EHR-based	2015	Bardhan et al.	Hospital readmission prediction	Design Science
		2017	Lin et al.	Hospital adverse event prediction	Design Science

Note: EHR = electronic health record

The breadth of the IS discipline has enabled four genres of design science to emerge: computational, optimization, representation, and economics (Rai 2017). Computational design science provides IS scholars with three concrete guidelines to design novel algorithms, computational models, and systems for advanced data analytics applications (e.g., ADLR). First, the IT artifact's design should be inspired by key domain characteristics. Lin et al. (2017) offers a recent healthcare IS example, where key contextual cues from EHRs guided a novel Bayesian multitask learning approach for predicting adverse events in hospitals. Second, researchers should demonstrate the novelty of their design and its technical superiority over selected baseline approaches via quantitative metrics (e.g., accuracy, precision, recall, F_1, etc.). Finally, the artifact's design should contribute situated implementations (e.g., software), nascent design theory (e.g., design principles), and/or well-developed design theory to the IS knowledge base (Rai 2017; Gregor and Hevner 2013). Executing each guideline requires understanding the application (in this study, motion sensor-based ADLR) for which the artifact is being developed.

Motion Sensor-based ADLR

ADLR uses sensors placed on humans and/or objects to identify ADL events at three levels of granularity: interaction, gesture, and activity (Roggen et al. 2010). *Interaction recognition* extracts information about reciprocal, physical motion primitives between an activity performer and an unspecified object (e.g., pull). Object information (e.g., "sensor is attached to the fridge") is then added to the interaction (e.g., pull) to complete each *gesture's* semantics (e.g., open the fridge). *Activity recognition* uses gesture patterns to identify complex and often interwoven operations that consist of temporally dependent sequences of gestures. Activity (i.e., HL-ADL) examples include drinking coffee and eating a sandwich and can span several minutes. Activities can share the same gestures but have different gesture patterns (e.g., drinking from a cup for both a coffee break and to take medication). Figure 1 illustrates how each level is decomposed and relates to others.

Most motion sensors used for ADLR (e.g., accelerometers, gyroscopes) have multiple axes that generate time-series data with temporal single-axial, cross-axial, and cross-sensor-axial patterns. Single-axial dependency denotes the temporal patterns within the same axis. Cross-axial dependency denotes patterns such as sensor rotation. Cross-sensor-axial dependency is the local dependency between axes of different sensors, denoting sensor interactions. Figure 2 further illustrates each dependency.

Multiple human sensors can be attached to different body locations to collect comprehensive human motion data, while one motion sensor is often sufficient for an object (Roggen et al. 2010). Irrespective of sensor type, continuous values (e.g., acceleration) are sampled along each axis simultaneously. When sampled at 10 Hz (i.e., 10 data samples per second), the sensor can generate 864,000 data samples per axis per day, a rate beyond a human's cognitive processing capability. This velocity and volume have motivated scholars to employ computational approaches for ADLR applications.

Computational Models for ADLR

Conventional computational ADLR models include classical machine learning algorithms such as discriminative models (e.g., support vector machine [SVM] in Reyes-Ortiz et al. 2016; k Nearest Neighbors [kNN] in Cao et al. 2012) and generative models (e.g., Hidden Markov Model [HMM] in Safi et al. 2016). These algorithms rely on manually engineered features such as activity duration, location, acceleration, rotation, signal amplitude, and motion frequency (Cao et al. 2012; Emi and Stankovic 2015; Safi et al. 2016). However, feature engineering is often labor intensive, ad hoc, and may not extract all salient cues. These issues have motivated numerous scholars to use deep learning for ADLR (Wang et al. 2019). Deep learning is a class of machine learning algorithms that use multiple layers of feed-forward artificial neural networks (ANNs) with nonlinear activation functions, error correction, and backpropagation to automatically learn the most salient features from data (LeCun et al. 2015). Table 3 summarizes selected recent studies applying deep learning on motion sensor signal data for ADLR.

While the ADL hierarchy suggests using both human and object sensors to maximize information for ADLR, most studies use human sensors only (Avilés-Cruz et al. 2019; Sun et al. 2018). Such systems require deploying human sensor networks with over 19 motion sensors on a user's wrists, chest, and other body parts (Murad and Pyun 2016; Ordóñez and Roggen 2016; Hammerla et al. 2016; Yang et al. 2015). These configurations are often obtrusive and do not mirror real-life home monitoring scenarios. Moreover, the lack of object information cannot pinpoint the semantics of performer-object interaction and limits HL-ADL detection. Most extant deep learning-based ADLR models only focus on a selected ADL level (e.g., gesture recognition) and do not capture all ADL levels. This results in models not generalizable for all ADL levels and prevents a comprehensive, end-to-end understanding of a subject's daily living patterns. Irrespective of configuration, CNNs are the prevailing deep learning model for analyzing sensor signal data. Figure 3 illustrates a common ADLR CNN architecture.

A CNN stacks convolutional and pooling layers to automatically learn features from raw data in a matrix form (Goodfellow et al. 2016). The convolution layer applies a grid kernel K on the data V to stride across the raw data and extract information from local regions in the input data. Most ADLR studies design a CNN with one dimension (1D) kernels to recognize single-axial temporal patterns from a single sensor type:

$$c_{i,j} = K_{1,[b]} \odot V_{i,[(j-1)\times s_c,b]},$$

where $c_{i,j}$ denotes the output value on the convolution layer at the position (i,j), and $\odot$ is the elementwise multiplication. $K_{1,[b]}$ only has one row with length b. $V_{i,[k:l]}$ denotes the l-length data segment from column k on the ith row. s_c is the stride lengths on the column (temporal) dimension, controlling the kernel's moving distance. Since these approaches extract local temporal patterns along different axes separately, they cannot guarantee that these patterns are temporally aligned (i.e., that motions along different axes occur together). Chen and Xue (2015) address this issue by altering the kernel shape to 2D.

$$c_{i,j} = \Sigma_{n=1}^{a}\left(K_{n,[b]} \odot V_{i+n-1,[(j-1)\times s_c,b]}\right).$$

However, this 2D kernel can only be applied to adjacent b rows in the input.

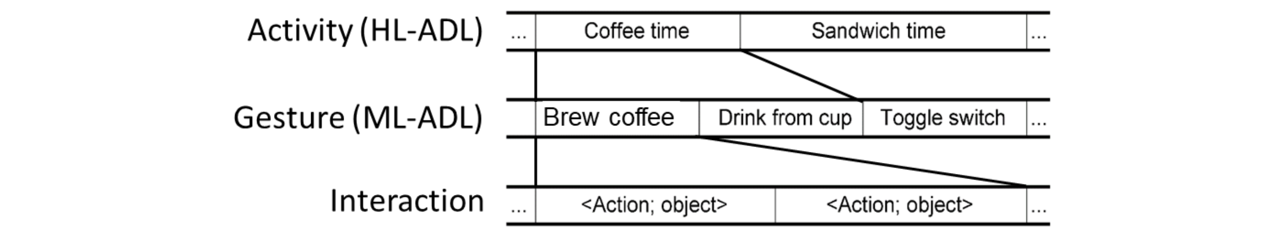


Figure 1. Example of ADL Decomposition (adapted from Roggen et al. 2010)

Note: ADLs are decomposed hierarchically into three levels: activity (HL-ADL), gesture (ML-ADL), and interaction. Elements on each level follow time sequences.

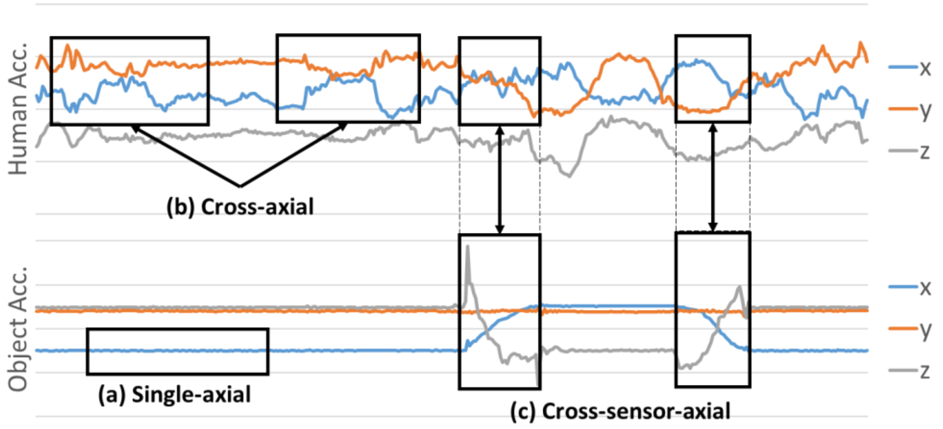


Figure 2. Different Dependencies in Motion Sensor Data: (a) Single-Axial, (b) Cross-Axial (x- and y-axis correlate negatively), and (c) Cross-Sensor Axial (human sensor's y-axis positively correlates with object sensor's z-axis).

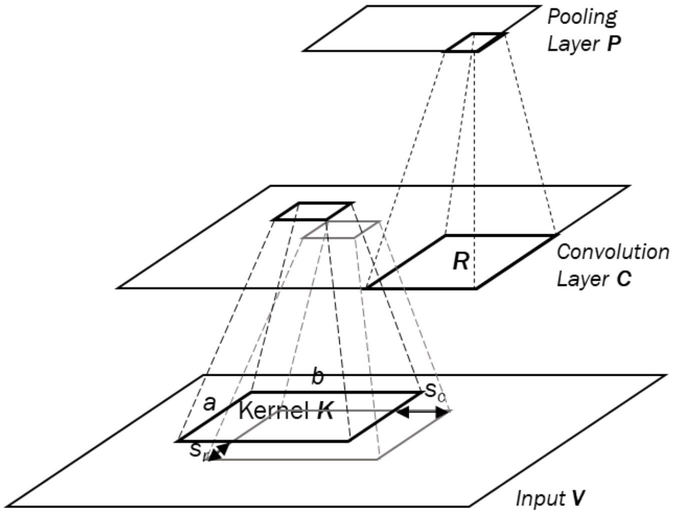


Figure 3. Convolution and Pooling Layers in a CNN

Table 3. Selected Recent Studies Applying Deep Learning on Motion Sensor Signal Data for ADLR

Year	Authors	# of object sensors	# of human sensors	Datasets	Models**	Task Level
2019	Avilés-Cruz et al.	0	2	Smartphone	Coarse-Fine CNN	Locomotion
		0	2	WISDM		
2018	Zhu et al.	8	1	Opportunity[+]	DeepConvLSTM + GRU-based Seq2Seq	HL-ADL
2018	Sun et al.	0	19	Opportunity	DeepConvLSTM	ML-ADL
2018	Liu et al.	0	1	Smartphone	SVM, CNN	Locomotion
2018	Pires et al.	0	1	Smartphone	ANN, DNN	Locomotion
2018	Kim et al.	0	2	10 subjects, 9 gesture patterns	CNN + GRU	ML-ADL
2018	Hassan et al.	0	2	Smartphone	PCA + DBN	Locomotion
2018	Li et al.	17	19	Opportunity	Hybrid, AE	ML-ADL, HL-ADL
2017	Almaslukh et al. 2017	0	1	Smartphone	SAE + SVM	Locomotion
2017	Murad and Pyun	0	19	Opportunity	LSTM	Locomotion, ML-ADL
		0	20	Skoda checkpoint		
2016	Ordóñez and Roggen	0	19	Opportunity,	DeepConvLSTM	Locomotion, ML-ADL
		0	20	Skoda checkpoint		
2016	Hammerla et al.	0	19	Opportunity	DNN, CNN, LSTM	ML-ADL
		0	3	PAMAP2		
2015	Chen and Xue	0	1	100 subjects, 8 functional movements	CNN	Locomotion
2015	Yang et al.	33	19	Opportunity	CNN	ML-ADL
		0	2	Hand Gesture		
2014	Zeng et al.	0	1	Opportunity	CNN	ML-ADL
		0	1	Skoda checkpoint		

Note: ** GRU = Gated Recurrent Unit; Seq2Seq = Sequence-to-Sequence Model; PCA = Principle Component Analysis; DBN = Deep Belief Network; AE = AutoEncoder; SAE = Stacked AutoEncoder, LSTM = Long Short-Term Memory, DNN = Deep Feed-Forward Neural Network. + "Opportunity" is an ADL dataset collected by a European Commission grant project.

Since CNNs were not designed to analyze temporal data, ADLR studies stack a recurrent neural network (RNN) to capture sequential human-object interactions for locomotion and gesture (ML-ADLs) recognition (Hammerla et al. 2016; Murad and Pyun 2017; Ordóñez and Roggen 2016).

RNNs are deep learning algorithms that include edges connecting adjacent time steps to capture temporal dependencies from sequential data (Lipton et al. 2015). Conventionally, RNN and its variants (e.g., LSTM, GRU) output one label for an entire sequence or generate one output label after processing a new input in the sequence. However, producing one label for an entire sequence prevents fine-grained, accurate, and ongoing HL-ADL recognition, which is critical for timely senior citizen care. Activities consist of interwoven and temporally dependent gesture sequences. Additionally, different activities can share similar gesture and locomotion patterns. A common approach to providing a series of labels for a sequence of activities uses a GRU-based Seq2Seq model (Zhu et al. 2018). Figure 4 depicts such a model for detecting HL-ADL label sequences from inputted gesture sequences.

The GRU-based Seq2Seq model "translates" raw sensor signals into human interpretable labels while maintaining the activity semantics using varied-length encoder and decoder recurrent networks (Cho et al. 2014; Sutskever et al. 2014). The encoder processes the input sequence $X = \{x_t, t = 1,2, \dots, n\}$ to capture local and global temporal patterns and represents them in a vector $\boldsymbol{s}$. The decoder then generates output sequence $Y = \{y_t, t = 1,2, \dots, m\}$ based on $\boldsymbol{s}$. When recognizing an activity, the vector $\boldsymbol{s}$ encodes the temporal patterns of the input (e.g., activity states) for the decoder to generate the ADL labels. If the output length (i.e., length of the HL-ADL label sequence) matches that of the input sequence, HL-ADLs are recognized for each time step at a finer granularity than in conventional approaches (Zhu et al. 2018).

Research Gaps and Questions

Our literature review revealed several research gaps. First, few ADLR studies use both human and object motion sensors, which often leads to scholars deploying a large volume of wearable human sensors that can be obtrusive, unrealistic in common home settings, and can limit comprehensive ADLR. Second, although deep learning-based ADLR studies address manual feature engineering issues, current deep learning models can only extract the single-axial or cross-axial temporal dependency within one sensor. Leveraging human-object interactions for comprehensive ADLR necessitates novel deep learning architectures that extract cross-sensor axial dependency between human and object motion sensors. Third, extant models are dedicated to a specific recognition task on a selected ADL level with a particular time scope (e.g., 5-10 second data segments to recognize a gesture). This limits the variety of features extracted from sensor data, as well as the ability of ADLR models to analyze ADLs with different granularities. In order to comprehensively understand a senior citizen's ADL performance, CNNs should be used to capture local data dependency for shorter (i.e., around 10 seconds), lower-level ADLs (e.g., interaction and gesture). However, the temporal patterns of these interactions/ML-ADLs should be modeled with RNNs (e.g., Seq2Seq) to recognize longer (e.g., 5 minutes), high-level ADLs. Based on these research gaps, the following research questions are posed for study:

- How can human and object motion sensors be used jointly for ADLR?
- How can cross-sensor-axial dependency be extracted from a pair of sensors and incorporated in ADLR?
- How can multiple levels of ADLs be recognized within an end-to-end framework?

A Hierarchical Multiphase ADL Recognition Framework

Guided by the ADL hierarchy, we design a novel hierarchical, multiphase ADL recognition framework. The framework consists of three components: interaction extraction, gesture (ML-ADL) recognition, and activity (HL-ADL) recognition. Each component outputs interpretable labels (e.g., human-object interaction, gesture) as suggested by the ADL hierarchy, enhancing subsequent task performance by limiting the intermediate result's dimensionality. Figure 5 illustrates the overall framework. The interaction extraction and gesture recognition components generate labels for local, short data samples (e.g., 5-8 seconds). A sliding window strategy generates ML-ADL sequences (e.g., five minutes) for the activity recognition component (Huynh et al. 2008). We discuss each component and their respective evaluations in the ensuing subsections.

Interaction Extraction

The ADL hierarchy indicates that gestures consist of human-object interactions (Roggen et al. 2010). The framework's first step extracts the dominant human-object interactions for gesture recognition. Gestures can be grouped due to their homogenous relative motions regardless of context (e.g., the object moves up in "pick up pillbox" and "pick up coffee cup" gestures, and the object moves toward the human in "open fridge door" and "open door inward" gestures). Capturing this homogeneity requires decomposing gestures along the three anatomical axes (sagittal, vertical, and frontal) into six generic interactions: "push," "pull," "pick up," "put down," "slide left," and "slide right" (Fan et al. 2011). The opposite of generic interaction is "no interaction." We design a novel interaction-based CNN to extract the interaction between each human-object sensor pair. Data from a pair of human and object motion sensors are stacked along the sensor channel direction as the input V of interaction extraction, as shown in Figure 6.

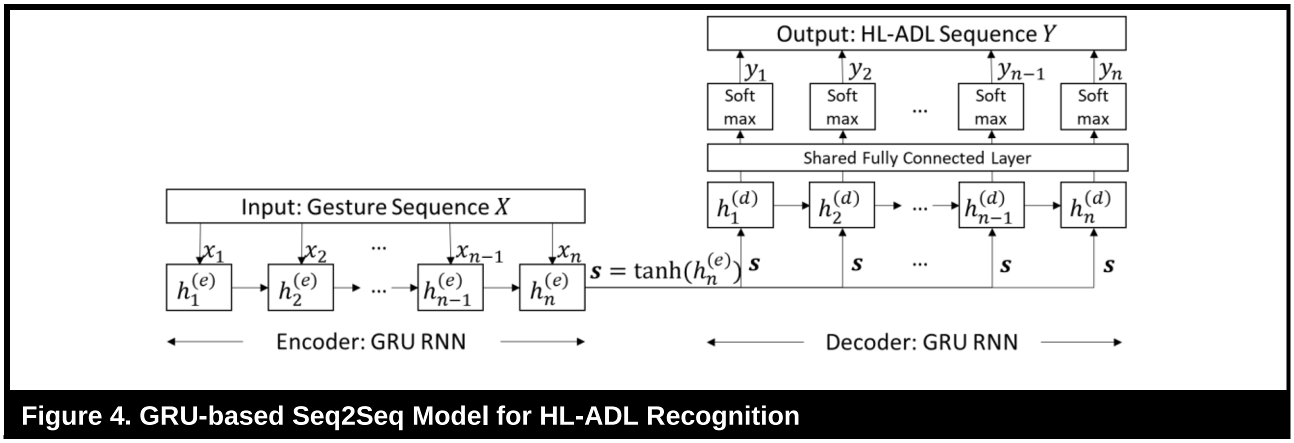


Figure 4. GRU-based Seq2Seq Model for HL-ADL Recognition

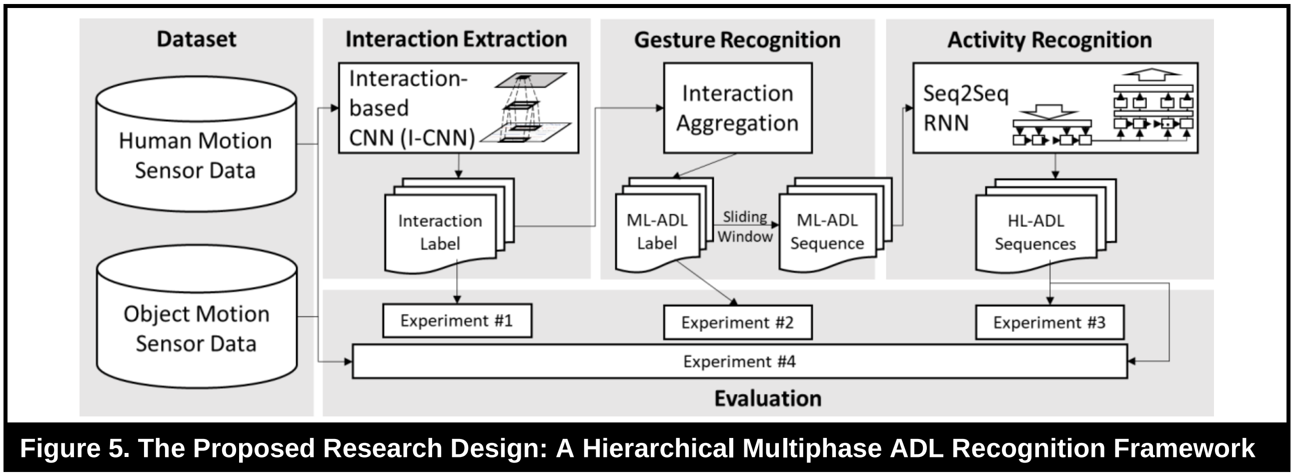
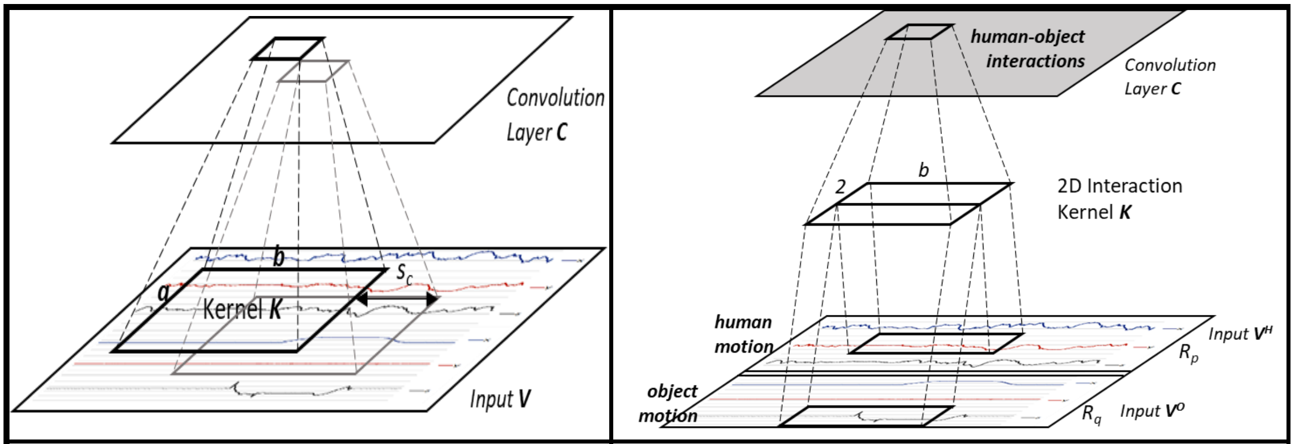


Figure 5. The Proposed Research Design: A Hierarchical Multiphase ADL Recognition Framework



Conventional 2D Kernel

- Operates on one input (e.g., one sensor)
- Kernel rows apply to consecutive rows in the input
- Convolution result represents the feature in the focal region

Proposed 2D Interaction Kernel

- Operates on two data sources
- Two kernel rows apply to one channel of each source
- Convolution result captures the feature between each pair of sensor channels

Figure 6. Illustration of Conventional 2D Kernel and Proposed 2D Interaction Kernel

Cross-sensor-axial dependency in motion sensor data represents human-object interactions. We utilize CNN's strength in local dependency extraction. Conventional 2D kernels $c_{i,j} = \Sigma_{n=1}^{a}\left(K_{n,[b]} \odot V_{i+n-1,[(j-1)\times s_c,b]}\right)$ are directly applied on a ($a > 1$) spatially adjacent rows of the input V to capture cross-axial patterns within a sensor (left side of Figure 6, Chen and Xue 2015). However, this conventional 2D convolution kernel cannot extract cross-sensor-axial patterns. We propose a novel 2D interaction kernel for CNNs (right side of Figure 6) formulated as follows:

$$c_{(p-1)\times R_q+q,j} = K_{1,[b]} \odot V^{H}_{p,[(j-1)\times s_c,b]} + K_{2,[b]} \odot V^{O}_{q,[(j-1)\times s_c,b]},$$

$$p \in \{1,\ 2,\ \ldots,\ R_p\},\ q \in \{1,\ 2,\ \ldots,\ R_q\},$$

where the input V consists of two separate input: V^H with R_p rows of human motion sensor channels and V^O with R_q rows of object motion sensor channels.

The 2D interaction kernel K includes two rows: the first row K_1 applies to a human sensor axis, and the second row K_2 applies to an object motion sensor axis. Both kernel rows extract information from the same local time window. These two rows jointly capture cross-sensor axial dependency from different sensor types. We install this novel kernel on the first layer of our five-layer interaction extraction convolutional neural network (I-CNN) to ensure the extracted patterns are temporally aligned. The I-CNN classifies the data from a pair of human and object motion sensors into interaction labels (e.g., "no interaction," "push," "pick up"). We detail I-CNN's full technical implementation in Appendix A.

Gesture (ML-ADL) Recognition

Unlike extant gesture recognition models (e.g., DeepConvLSTM) that directly classify raw sensor segments into gesture classes, we leverage the extracted interaction and recognize gestures with a heuristic approach. I-CNN outputs the most likely interaction I_n between human and object$_n$ (e.g., "push," "no interaction") and its probability p_n for each human-object data segment. We pass the extracted interactions from all human-object pairs to the gesture recognition phase to infer the most likely gesture. We propose a heuristic-based four-step process (I-CNN-GR) for the gesture (ML-ADL) recognition phase in our framework to aggregate and recognize the most salient gesture from all human-object pairs. We assume that (1) one person can interact with only one object at a time, and (2) if one gesture exists, its underlying interaction should be the most salient among all the interactions. These assumptions guarantee I-CNN-GR will provide robust recognition when deployed in a multiresident environment, where residents can coincidentally use different objects at the same time, resulting in various candidate interactions.

Step 1. If $\forall\ j \in \{1, 2, \ldots, n\}$, $I_j =$ "no interaction," then output "No Gesture."

Step 2. Sort $\{I_j\}$ based on p_j in descending order.

Step 3. Find the first I_k where $I_k \neq$ "no interaction."

Step 4. Map (I_k, $Object_k$) to the corresponding gesture and output the gesture.

Step 1 outputs "no gesture" if the human does not interact with any object. Otherwise, Steps 2 and 3 identify the most likely (confident) human-object interaction recognized by I-CNN. The corresponding object information is then incorporated with the interaction in Step 4 for gesture mapping (e.g., (push, fridge door) → "close the fridge").

Activity (HL-ADL) Recognition

Since HL-ADLs are motion sequences in longer time spans (e.g., 5 minutes), a one-second stride-length sliding window segments raw sensor data for I-CNN-GR. This generates a sequence of gestures (one gesture label per second) for the activity (HL-ADL) Recognition phase. The activity recognition phase adopts a GRU-based Seq2Seq model (S2S_GRU) to learn the temporal gesture patterns and assign HL-ADL labels for each gesture in the sequence (Zhu et al. 2018). The encoder network takes the gesture sequence $X = [x_1, x_2, \cdots, x_n]$ as input, where each x_t is a gesture label (e.g., "open fridge," "close drawer") at time step t.

The GRU cells process the entire gesture sequence and learn to extract and store the most salient short-term and long-term temporal gesture patterns in the hidden cell states. The final cell state is encoded as the semantics vector $\boldsymbol{s}$, which generates HL-ADL labels for X (Zhu et al. 2018). Vector $\boldsymbol{s}$ is repeated as the input for all n decoding time steps. The decoder's GRU cell extracts information from different dimensions of $\boldsymbol{s}$ for each time step.

Table 4. A Summary of the Proposed Hierarchical ADL Recognition Framework

Phase	Task	Model	Input	Output	Examples
1	Interaction extraction	I-CNN	Raw human and object motion sensor data from one human-object sensor pair (< 10 seconds)	**An interaction label:** No interaction, push, pull, pick up, put down	Raw sensor data → "push" interaction
2	Gesture recognition	Heuristic-based	**Interaction labels from all human-object sensor pairs:** No Interaction, Push, Pull, Pick up, Put Down (< 15 seconds)	**ML-ADL label**: open fridge, close door, etc.	"push" + "fridge" → "close fridge" gesture
3	Activity recognition	S2S_GRU	**A sequence of ML-ADL labels:** Open fridge – Close fridge – Use fork … (> 1 minute)	**A sequence of HL-ADL labels:** food prep – food prep – dining …	"open fridge – close fridge – use fork -…- open dishwasher" → "food prep – food prep – dining -…- clean-up" activity

Extracted information is transformed by a shared fully connected layer to a lower dimension that matches the number of HL-ADL labels for classification; a $softmax$ function selects the most probable predefined label (e.g., "food preparation" for the "open fridge" gesture) as the output y_t for time *t*. Through this hierarchical, multiphase ADL recognition framework, interactions, gestures, and activities are automatically recognized from human and object motion sensor data. We summarize the process of our proposed framework in Table 4.

Experimental Study

A core tenet of the computational design science paradigm is rigorously evaluating the proposed IT artifact against well-established benchmark methods. To this end, we evaluate each component and the entirety of the proposed framework.

Two datasets are used as ground truth: "Opportunity" (OPPO) and a human-object interaction dataset (INTER) (Table 5). OPPO is a publicly available morning activity EU dataset containing 133 human and 109 object motion sensor channels, sampling at 30 Hz, with hierarchical ADL labels (ML-ADLs and HL-ADLs). Roggen et al. (2010) deployed a comprehensive wearable sensor network consisting of accelerometers, gyroscopes, magnetometers, and integrated inertial measurement units. This dataset is widely used for benchmarking in ADLR studies (Ordóñez and Roggen 2016; Hammerla et al. 2016; Yang et al. 2015; Zeng et al. 2015). OPPO's rich labels allow us to divide the overall dataset into testbeds for different evaluations (Table 5). Based on relevance to the ML-ADL labels, we selected eight object motion sensors that are attached to the most representative objects. These sensors were placed on two doors, three drawers, a fridge door, a dishwasher door, and a cup. In total, our selected dataset contained 27 accelerometer channels (9 triaxial sensors) out of the 242 data channels in the original OPPO dataset.

We also collected a human-object interaction dataset (INTER) using the NSF-supported SilverLink smart home monitoring system (Chuang et al. 2015; Maimoon et al. 2016; Yu et al. 2017) to evaluate the interaction extraction phase. SilverLink consists of five coin-size triaxial accelerometers with a 25 Hz sampling rate and $\pm$2G sensitivity. The INTER dataset was generated by a controlled lab experiment, similar to the drill runs in the OPPO dataset (Roggen et al. 2010). One accelerometer was attached to the subject as a pendant on the chest. Four object motion sensors were placed on the fridge door, pillbox, bedroom door, and bathroom door to collect interactions during food-, medication-, and hygiene-related ADLs. Subjects were instructed to walk to the object, perform specific human-object interactions, and walk away, simulating the daily human-object interaction patterns. We labeled each segment as "push," "pull," "pick up," "put down," or "no interaction."

Table 5. A Summary of Research Testbeds

Datasets	Sensors Used	Testbed	Labels	Number or %
OPPO	**1 Wearable Acc.:** Back **8 Object Acc.:** Door 1 Door 2 Fridge Dishwasher Drawer 1 Drawer 2 Drawer 3 Coffee cup	**OPPO-HL:** 1,135 HL-ADL segments 300 ML-ADL & HL-ADL labels per segment (one per second) 9,000 raw data samples per segment 27 sensor channels per sample	Relaxing	17.91%
			Coffee time	15.49%
			Early morning	23.82%
			Clean up	12.09%
			Sandwich time	30.69%
		OPPO-ML: 2,394 gesture (ML-ADL) segments 240 raw data samples per segment 27 sensor channels per sample	Open door 1	115
			Close door 1	121
			Open door 2	113
			Close door 2	119
			Open fridge	209
			Close fridge	202
			Open dishwasher	131
			Close dishwasher	133
			Open drawer 1	124
			Close drawer 1	123
			Open drawer 2	118
			Close drawer 2	120
			Open drawer 3	129
			Close drawer 3	131
			Drink from cup	253
			Put away cup	253
INTER	**1 Wearable Acc.:** Chest **4 Object Acc.:** Bedroom door Bathroom door Fridge Pillbox	**INTER:** 8,000 interaction segments 240 data samples per segment 15 sensor channels per sample	No interaction	1,600
			Push	1,600
			Pull	1,600
			Pick up	1,600
			Put down	1,600

To evaluate our design in an unobtrusive, real-world setting, we used the back accelerometer data from OPPO and the chest pendant sensor data from INTER. These locations were less obtrusive but provided information about HL-ADLs and locomotion transition (Atallah et al. 2011). We obtained 8,000 interaction segments from INTER and extracted 2,394 gesture and 1,135 activity segments to form our three testbeds: INTER, OPPO-ML, and OPPO-HL, respectively. OPPO-ML segments are eight-second segments containing one ML-ADL at the center. OPPO-HL are five-minute segments extracted with a 0.5-minute sliding window (similar strategy to Huynh et al. 2008). The INTER testbed is balanced with 1,600 segments in each class. OPPO-ML is slightly imbalanced with more fridge and cup-related gestures. OPPO-HL is also imbalanced with more sandwich preparation and less coffee making, clean up, and relaxation. This reflects the varying actual time requirements for these daily activities. Each INTER and OPPO-ML segment has one label, while each OPPO-HL segment has 300 labels corresponding to each data sample to represent the interweaving ADLs correctly performed by the subject.

Experiment Design and Performance Metrics

We used both testbeds to conduct four sets of evaluations: interaction extraction, gesture recognition, activity recognition, and end-to-end evaluation. Each corresponds to one phase in our proposed framework. Table 6 provides a full summary of the experiments. Experiment 1 evaluates interaction extraction performance from human-object motion sensor pairs. We used the 8,000 interaction segments for this experiment. Signal features are extracted from each interaction segment for classical machine learning benchmarks. Extracted features include minimum, maximum, mean, standard deviation, energy, and entropy (Bao and Intille 2004). We evaluated the proposed I-CNN model against kNN, SVM (Cao et al. 2012), and CNN-1D using standard performance metrics of precision, recall, and F$_1$ score. For each class C_i, these metrics are as follows.

$$\text{Precision}^{C_i} = \frac{\text{Correctly predicted } C_i}{\text{Total predicted } C_i},$$

$$\text{Recall}^{C_i} = \frac{\text{Correctly predicted } C_i}{\text{Total True } C_i},$$

$$F_1^{C_i} = \frac{2 \times \text{Precision}^{C_i} \times \text{Recall}^{C_i}}{\text{Precision}^{C_i} + \text{Recall}^{C_i}}.$$

We calculated the macro-averaged F$_1$ score using each category's F$_1$ score to evaluate the classification performance over N categories ($N = 5$ in Experiment 1) (Forman 2003), where

$$\text{macro-averaged } F_1 = \frac{1}{N}\sum_{i=1}^{N} F_1^{C_i}.$$

Experiment 2 evaluated I-CNN-based Gesture Recognition (I-CNN-GR) against state-of-the-art gesture recognition benchmarks. I-CNN-GR mapped each gesture segment to one of the 16 OPPO-ML labels based on the eight possible interactions extracted by I-CNN from each human-object sensor pair. As in Experiment 1, signal features were extracted from each gesture segment for classical machine learning benchmarks. Raw data was used for deep learning models (I-CNN-GR, DeepConvLSTM, CNN-1D, and CNN-2D). We evaluated performance using precision, recall, and F$_1$, and also used the macro-averaged F$_1$ ($N = 16$).

Experiment 3 compares S2S_GRU against other sequential learning benchmarks (e.g., HMM, S2S_LSTM) by evaluating HL-ADL sequence quality. I-CNN-GR extracts 300 gesture labels from raw sensor data for each activity segment as the input for experiment models. With the input, experiment models predict HL-ADL label sequences (e.g., "food-food-food-medication-medication"). Two metrics proposed by Zhu et al. (2018)—accuracy and block Levenshtein distance (BLD)—evaluate the structure of these HL-ADL sequences (i.e., duration of an activity, boundary of different activities).

$$\text{Accuracy} = \frac{\text{Correct labels in the sequence}}{\text{Length of the sequence}}.$$

Accuracy evaluates how well the recognized HL-ADL label sequence reflects the real start and end time for HL-ADLs.

$$BLD = LD(\text{Recognized HL-ADL Blocks, True HL-ADL Blocks}),$$

where LD is the Levenshtein distance, denoting the number of deletions, insertions, or substitutions required to transform sequence A to sequence B (Levenshtein 1966). The successive HL-ADL labels are aggregated to HL-ADL blocks to condense the sequence while preserving the order of HL-ADLs (e.g., "food-food-food-non-med-med" HL-ADL label sequence, condensed as the "food-non-med" block sequence).

Table 6. A Summary of Experiment Designs

Exp.	Our Model	Benchmarks	Target Phase	Testbed	Evaluation Metrics	Prior Study
# 1	I-CNN	Signal features + kNN	Interaction extraction	INTER	Precision, recall, F_1, averaged F_1	Cao et al. 2012
		Signal features + SVM				
		CNN-1D				
# 2	I-CNN-GR	DeepConvLSTM	Gesture recognition	OPPO-ML	Precision, recall, F_1, averaged F_1	Ordóñez and Roggen 2016; Chen and Xue 2015; Cao et al. 2012; Bao and Intille 2004
		CNN-1D				
		CNN-2D				
		Signal features + SVM				
		Signal features + DT				
# 3	I-CNN-GR-S2S_GRU	I-CNN-GR-S2S_LSTM	Activity recognition	OPPO-HL	Accuracy, average block Levenshtein distance (ABLD)	Chowdhury et al. 2013; Zhu et al. 2018
		I-CNN-GR-HMM				
# 4	I-CNN-GR-S2S_GRU	Signal Features + NB + LDA SAE + SVM	End-to-End	OPPO-HL	Accuracy @ 1, Accuracy @ 2	Huynh et al. 2008; Almaslukh et al. 2017

Note: Exp. = Experiment

We further averaged BLDs on all sequences to obtain the averaged BLD (ABLD) in order to evaluate the model's HL-ADL sequence-generating quality.

$$ABLD = \frac{1}{n}\sum_{i=1}^{n} BLD_{\text{sequence } i}.$$

Experiment 4 compares the end-to-end ADLR performance of our hierarchical framework and selected state-of-the-art high-level activity recognition benchmarks. Because prevailing ADLR models produce a label for an input segment (Li et al. 2018; Ordóñez and Roggen 2016), the accuracy score is used to evaluate the predicted label against the ground truth. Our ADLR model processed the raw data hierarchically and obtained an HL-ADL label sequence for each segment. We selected the majority label (i.e., the label with the highest count) as the predicted activity label for each segment.

The first benchmark is a stacked autoencoder-based activity recognition model (Almaslukh et al. 2017; Li et al. 2018). Sensor data representation was automatically learned with two stacked and greedily trained autoencoders. SVM classified the condensed data representation to classify sensor segments into one of the HL-ADL labels. The second benchmark is a topic-modeling-based activity recognition model (Huynh et al. 2008; Ihianle et al. 2016; White 2018). Sensor data were considered as vocabularies in the corpora. Latent Dirichlet Allocation (Blei et al. 2003) extracted the underlying HL-ADL activities as topics in an unsupervised manner. In order to assign an HL-ADL label to each segment's most likely topic, we constructed a confusion matrix by mapping each segment's most likely topics and its majority HL-ADL label. The Hungarian algorithm (Kuhn 1955) calculated the optimized assignment that maximizes the label prediction accuracy (Table D8). Finally, each segment obtained a predicted HL-ADL label based on its most likely topic.

Each OPPO-HL segment contains interweaving HL-ADLs during the five-minute duration. On average, 90% of the data had a label with top-two counts (see Appendix B for additional details). Therefore, we created two sets of ground truth labels for evaluation: TOP-1 and TOP-2.

TOP-1 consists of the majority HL-ADL label in each segment, while TOP-2 contains the labels with the top and second highest counts. TOP-2 statistics are summarized in Appendix B. Inspired by metrics such as "precision at k" and "top-N accuracy" (Cremonesi et al. 2010; Kelly 2007), we named the accuracy scores evaluated on TOP-1 and TOP-2 "Accuracy @ 1" (Acc@1) and "Accuracy @ 2" (Acc@2), with formulas as follows:

$$\text{Accuracy @ 1} = \frac{\text{count}_{i=1}^{N}(\text{pred\_label}_i = \text{TOP-1}_i\ )}{N},$$

$$\text{Accuracy @ 2} = \frac{\text{count}_{i=1}^{N}(\text{pred\_label}_i \in \text{TOP-2}_i\ )}{N}.$$

Accuracy @ 1 measures how well a model can recognize the major activity within a segment. Accuracy @ 2 measures how well a model avoids mistakenly labeling a segment into less-likely activities. All models were trained and tested with 10-fold cross-validation. We conducted paired *t*-tests on all macro-averaged F_1 scores, accuracies, and ABLDs. All experiments were conducted in an Ubuntu 16.04-based Python 3.5 environment on a workstation with i5-4430 CPU and 24 GB memory. The deep learning models (i.e., I-CNN, DeepConvLSTM, S2S_GRU, S2S_LSTM) were implemented with Keras (Chollet 2015), the HMM with hmmlearn (Lebedev 2016), the LDA with Gensim (Řehůřek and Sojka 2010), the Hungarian algorithm with SciPy (Millman and Aivazis 2011), and other benchmarks with scikit-learn (Pedregosa et al. 2011). Appendix D summarizes each model's parameters for replicability purposes.

Experiment Results

Experiment 1: Interaction Extraction

We evaluated our proposed I-CNN model with a 2D interaction kernel with kNN, SVM, and CNN with 1D kernel (CNN-1D). The precision, recall, and F_1 scores of the five labels are summarized in Table 7. The highest scores are highlighted in boldface. Overall, I-CNN, kNN, and CNN-1D showed high precision, recall, and F_1 scores compared to SVM. Further statistical tests show that the proposed CNN model (I-CNN) achieved an averaged F_1 score of 0.736, significantly outperforming benchmarks kNN (0.669), SVM (0.614), and CNN-1D (0.665). Classical machine learning methods such as kNN and SVM classified more interaction segments to the "no interaction" category, resulting in lower recall in the four generic interaction categories. CNN-based methods were less likely to omit a generic interaction and could better distinguish between generic interactions and "no interaction" signals. In general, the CNN-based methods seem promising for interaction extraction in home monitoring use cases. The 2D interaction kernel in I-CNN extracts cross-sensor axial dependency, which helped I-CNN outperform CNN-1D. The results show the advantage of I-CNN in the interaction extraction phase, ensuring a sound foundation for the subsequent gesture recognition phase.

Experiment 2: I-CNN-GR vs. Gesture Recognition Benchmarks

Experiment 2 compared the gesture recognition performance between I-CNN-GR and benchmarks (DeepConvLSTM, CNN-1D, CNN-2D, SVM, and Decision Tree). Our I-CNN-GR model achieved an F_1 score of 0.85, outperforming all the benchmarks with statistically significant margins. Although Ordóñez and Roggen (2016) reported that DeepConvLSTM achieved an averaged F_1 score of 0.69 using 113 wearable accelerometer channels, this state-of-the-art model's performance dropped significantly ($F_1 = 0.25$) when its input was from different sensor types (one wearable and eight object motion sensors, 27 channels in total) and its model parameter size was restricted. Table 8 summarizes the detailed precision, recall, and F_1 scores of the 16-class classification. The highest scores are highlighted in bold font.

DeepConvLSTM classified most data segments to fridge-, dishwasher-, and cup-related gestures, and SVM classified data to cup-related gestures. Both approaches learned the majority of data classes instead of gesture features, leading to low recognition performances (weighted F_1 of 0.11 for SVM and 0.25 for DeepConvLSTM). CNN-1D and CNN-2D performed very well in recognizing object-specific patterns (e.g., drawer-related [F_1 ranges from 0.77 to 0.87 for CNN-1D] and fridge-/dishwasher-related [F_1 ranges from 0.90 to 0.98 for CNN-2D)] gestures), resulting in less generalizable models for arbitrary object combinations in real-life scenarios. I-CNN-GR demonstrated a balanced performance (SD = 0.069) across all 16 gesture classes. These results indicate that I-CNN-GR can leverage data from different sensor types by modeling human-object interactions and can effectively extract generalizable features for gesture recognition.

Table 7. Interaction Extraction Performance of I-CNN vs. Benchmarks

	I-CNN			**kNN (k = 1)**			**SVM (RBF kernel)**			**CNN-1D**		
	P	**R**	**F_1**	**P**	**R**	**F_1**	**P**	**R**	**F_1**	**P**	**R**	**F_1**
No Interaction	0.685	0.507	**0.583**	0.415	**0.611**	0.494	0.511	0.588	0.547	**0.686**	0.485	0.568
Push	0.682	**0.763**	**0.720**	**0.738**	0.701	0.719	0.540	0.626	0.580	0.710	0.667	0.688
Pull	0.682	0.833	**0.750**	**0.797**	0.684	0.736	0.628	0.564	0.594	0.632	**0.860**	0.729
Pick up	**0.830**	**0.749**	**0.787**	0.767	0.668	0.714	0.712	0.554	0.624	0.681	0.548	0.607
Put down	**0.836**	**0.847**	**0.842**	0.742	0.632	0.683	0.719	0.729	0.724	0.675	0.801	0.732
Averaged F_1	**0.736*** **			0.669			0.614			0.665		

Note: *** p-value<0.001. Highest scores are given in **bold**

Table 8. Gesture Recognition Performance of I-CNN-GR vs. Benchmarks

	I-CNN-GR			**DeepConvLSTM**			**CNN-1D**			**CNN-2D**			**SVM + Features**			**DT + Features**		
	P	**R**	**F_1**	**P**	**R**	**F_1**	**P**	**R**	**F_1**	**P**	**R**	**F_1**	**P**	**R**	**F_1**	**P**	**R**	**F_1**
Open door 1	**0.85**	**0.92**	**0.88**	0.00	0.00	N/A	0.47	0.84	0.60	0.51	0.81	0.63	0.23	0.12	0.16	0.74	0.71	0.73
Close door 1	**1.00**	**0.73**	**0.84**	N/A	0.00	N/A	0.64	0.47	0.54	0.66	0.52	0.58	0.23	0.05	0.09	0.71	0.72	0.71
Open door 2	**0.92**	**1.00**	**0.96**	0.00	0.00	N/A	0.58	0.84	0.69	0.76	0.90	0.83	0.15	0.05	0.08	0.66	0.63	0.64
Close door 2	**1.00**	**0.82**	**0.90**	N/A	0.00	N/A	0.73	0.53	0.62	0.89	0.72	0.79	0.27	0.06	0.10	0.64	0.69	0.67
Open fridge	0.81	0.81	0.81	0.22	0.58	0.32	0.87	0.88	0.88	**0.92**	**0.93**	**0.93**	0.13	0.05	0.07	0.73	0.75	0.74
Close fridge	0.73	0.80	0.76	0.12	0.35	0.17	0.89	0.83	0.86	**0.93**	**0.86**	**0.90**	0.23	0.11	0.15	0.73	0.73	0.73
Open dishwasher	0.85	**1.00**	0.92	0.60	0.58	0.59	0.94	0.91	0.92	**0.95**	0.95	**0.95**	N/A	0.00	N/A	0.66	0.68	0.67
Close dishwasher	0.95	**1.00**	0.97	0.46	0.30	0.37	0.94	0.96	0.95	**0.98**	0.98	**0.98**	0.67	0.02	0.03	0.70	0.67	0.69
Open drawer 1	0.75	**0.92**	0.83	0.07	0.04	0.05	**0.86**	0.88	**0.87**	0.59	0.68	0.63	0.15	0.05	0.07	0.80	0.82	0.81
Close drawer 1	**0.92**	0.73	0.82	N/A	0.00	N/A	0.92	**0.75**	**0.83**	0.75	0.63	0.69	0.23	0.06	0.09	0.72	0.73	0.73
Open drawer 2	0.70	0.70	0.70	N/A	0.00	N/A	**0.85**	0.65	**0.77**	0.67	0.49	0.57	0.00	0.00	N/A	0.76	**0.77**	0.76
Close drawer 2	0.82	**0.82**	**0.82**	N/A	0.00	N/A	**0.91**	0.62	0.73	0.76	0.63	0.69	0.00	0.00	N/A	0.73	0.70	0.71
Open drawer 3	**0.73**	**0.92**	**0.82**	0.29	0.05	0.09	0.70	0.77	0.73	0.58	0.78	0.67	0.44	0.03	0.06	0.70	0.66	0.68
Close drawer 3	**0.91**	0.71	0.80	0.2	0.05	0.08	0.76	**0.84**	**0.80**	0.77	0.81	0.79	0.18	0.02	0.04	0.74	0.73	0.74
Drink from cup	0.88	0.84	0.86	0.28	0.76	0.41	0.90	0.88	0.89	**0.91**	**0.88**	**0.89**	0.14	0.86	0.24	0.77	0.78	0.78
Put away Cup	**0.95**	**0.86**	**0.90**	0.18	0.14	0.16	0.81	0.78	0.80	0.90	0.82	0.85	0.21	0.31	0.25	0.75	0.75	0.75
Averaged F_1 (SD^+)	**0.85** (0.069)			0.25*** (0.173)			0.77* (0.117)			0.77* (0.133)			0.11*** (0.068)			0.72*** (0.043)		

Note: **p*-value < 0.05, ***p*-value < 0.01, ****p*-value < 0.001.+: SD = standard deviation. Highest scores are given in **bold.**

In real-life scenarios, some gestures are likely to follow particular gestures and co-occur in a short time period (e.g., "close the fridge" shortly after "open the fridge"). Therefore, an alternate design choice incorporates historical information (i.e., the most recent gestures) to recognize upcoming gestures.

We conducted an additional experiment to ascertain the effect of including historical information with these correlations on model performances. Appendix C summarizes all experiment details. Overall, we found that including historical information as features resulted in more accurate gesture recognition for non-interaction-based benchmarks (e.g., DT + signal features). However, none of the benchmark methods outperformed the proposed heuristic-based I-CNN-GR model.

Experiment 3: S2S_GRU vs. HMM/S2S_LSTM

Accuracy and ABLD measured HL-ADL sequence quality. As summarized in Table 9, Seq2Seq models (S2S_GRU and S2S_LSTM) outperformed HMM using both metrics. Statistical tests confirmed that S2S_GRU performed significantly better than S2S_LSTM using both metrics (p < 0.05). S2S_GRU's low ABLD score (0.44) and high accuracy (79.6%) indicate that it can better extract temporal high-level activity patterns and recognize HL-ADLs from the gesture sequences than other benchmarks. This is practically valuable for real-life home monitoring systems in anomaly detection, daily living pattern visualization, and other mobile health applications (e.g., food-intake frequency monitoring, drug adherence monitoring, etc.).

Experiment 4: Hierarchical ADLR Framework vs. Nonhierarchical HL-ADLR Benchmarks

Experiment 4 evaluates each model's predicted HL-ADL against two ground truth label sets: TOP-1 and TOP-2. Our proposed hierarchical ADLR framework outperforms nonhierarchical benchmarks with statistically significant margins (p-value < 0.001) on Acc@1 and Acc@2 (Table 10). Our model successfully predicted 65.9% of the majority labels (i.e., major activities), and our activity recognition results show that 90.1% match one of the top two HL-ADLs.

Three issues can explain the topic modeling-based approach's low recognition accuracy. First, only signal features were used; local interactions between sensors were ignored. Secondly, this model's "bag of gestures" assumption failed to capture the temporal dependencies. Third, in real life, ADLs are often interwoven. It is difficult to guarantee that the trained activity segment only belongs to one "topic," resulting in a biased parameter estimation. As a result, differences among extracted topics are minor (details can be found in Appendix D, Table D8), resulting in ambiguous pattern recognition and low ADLR performance (Acc@1 = 31.1% and Acc@2 = 46.6%). In addition, extracted topics rely on manual interpretation, which can be ad hoc and laborious, reducing the model's usability in real-life scenarios. The SAE-based approach extracted the general data representation with local and global dependencies, resulting in more accurate ADLR than the topic modeling-based model with statistical significance (p-value = 0.04 for Acc@1 and p-value = 0.03 for Acc@2). However, without ADL decomposition and meaningful dependency extraction (e.g., cross-sensor axial dependency), SAE has lower ADLR accuracy, and the extracted data representations were not interpretable. These results indicate that decomposing ADLs hierarchically and extracting salient dependencies at different ADL levels with carefully designed deep learning models help to accurately recognize HL-ADLs.

An End-to-End Case Study: Sandwich Time and Clean Up Activities

We illustrate the proof of concept and proof of value of our framework with an end-to-end case study. Since features extracted by the SAE-based approach were not interpretable, we compared our framework against the topic modeling-based approach. Figure 7 shows an end-to-end ADLR case from OPPO-HL based on our framework. Three object motion sensors attached to a fridge door, a cup, and a dishwasher were selected for demonstration. The human/fridge door sensor data segment shown at the top of Figure 7 shows that the subject walked to the fridge, opened the fridge (red box 1), closed the fridge (red box 2), and walked away. The fridge door shows an acceleration and deceleration pattern along the negative x direction. In red box 2, the fridge door shows another acceleration and deceleration pattern along the positive x direction. The motion sensor recorded a shock when the fridge door closed. The overall ADLR process is explained below.

In Phase 1, I-CNN processed all data segments from eight human-object pairs and generated interaction sequences for Phase 2. I-CNN extracted pull interactions ($p = 0.88$) around red box 1 and push interactions ($p = 0.97$) around red box 2.

Table 9. Activity Recognition Performance of S2S_GRU vs. S2S_LSTM & HMM

	S2S_GRU	**S2S_LSTM**	**HMM**
ABLD	**0.44**	0.57*	3.23***
Accuracy	**79.6%**	73.0%*	34.4%***

Note: *p-value < 0.05, ***p-value < 0.001

Table 10. End-to-End ADLR Accuracy Scores of Experiment Models

	Hierarchical ADLR	**SAE + SVM**	**Signal Features + NB + LDA (5 Topics)**
Acc@1	**65.9%**	42.3%***	31.1%***
Acc@2	**90.1%**	59.5%***	46.6%***

Note: ***p-value < 0.001

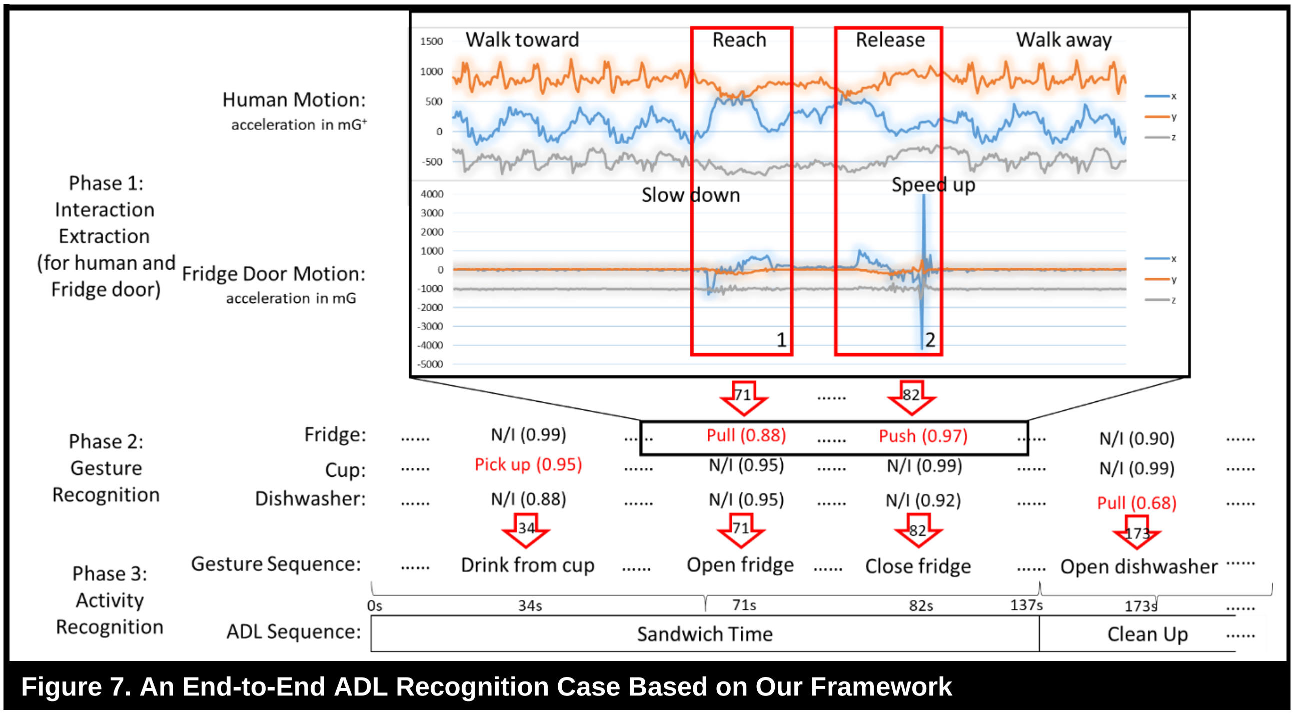


Figure 7. An End-to-End ADL Recognition Case Based on Our Framework

Note: "N/I" stands for "no interaction." +: 1 G (earth gravity) = 1,000 mG.

Interaction labels extracted from each object sensor within one second were aggregated to a gesture. For gesture #71, the pull interaction had the highest probability of non-N/I interactions. The model then assigned the object information, "fridge door," to the pull interaction and labeled gesture #71 with the "open fridge" gesture. In Phase 3, S2S_GRU captured temporal dependencies among gesture sequences and assigned HL-ADL labels to each of the 300 recognized gestures. In Figure 7, gestures before #137 were labeled as "sandwich time." The remainder were labeled as "clean up." The ground truth HL-ADL labels were "sandwich time" for #1-#137, "relaxing" for #138-#164, and "clean up" for #165-#300. The top two activities in this case were "sandwich time" (n = 137) and "Clean up" (n = 135). Our framework mislabeled #138-#145 as "Clean up," leading to a wrong majority activity ("Clean up" n = 163). Thus, the Acc@1 score was 0, and Acc@2 score was 1.

However, when evaluated on sequence quality as in Experiment 3, our framework accurately labeled 97.3% of the sequences, and the BLD score was 1.

The topic modeling-based method successfully extracted gesture #71 using naive Bayes but misclassified gesture #82 as "close fridge" because of irregular high accelerations. LDA's "bag of gestures" strategy ignored temporal dependencies and wrongly assigned two topics for the entire segment: "Early morning" with a 0.232 likelihood and "clean up" with a 0.148 likelihood, resulting in zero accuracy in both TOP-1 and TOP-2. This comparison demonstrated that our hierarchical ADLR framework better leverages dependencies on different ADL levels than the topic-modeling-based approach, enabling accurate HL-ADL recognition. In addition, intermediate results produced by our proposed framework (e.g., interaction labels, gesture sequences, and so on) provide more information for assessing the label sequence's quality. These intermediate results were interpretable, allowing manual verification and further analysis (e.g., object usage pattern analysis).

Discussion: Contributions to the IS Knowledge Base and Practical Implications

Ensuring healthy and safe independent living for senior citizens is a growing societal concern. For this research, we carefully adhered to the guidelines prescribed by the computational design science paradigm to create a novel hierarchical ADLR framework suitable for advanced predictive health analytics for senior care. Through the process of searching through a possible solution space, designing the ADLR framework, and rigorously evaluating the framework and its constituent components, we make several key knowledge contributions to the IS knowledge base. The following subsections discuss these contributions and their practical implications.

Contributions to the IS Knowledge Base

IS scholars have posited that novel IT artifacts should contribute prescriptive knowledge back to the IS knowledge base to guide the development of future artifacts (Nunamaker et al. 1990; Hevner et al. 2004; Gregor and Hevner 2013). Knowledge contributions for computational IT artifacts can include a situated implementation of a model or methods in a selected application environment and/or design principles generalizable to domains with similar characteristics (Rai 2017). In this study, the end-to-end ADLR framework is a situated implementation for comprehensively identifying ADLs at varying levels of granularity. Beyond its direct ADLR application, this framework's use of multiple sensor types, a novel 2D interaction kernel, and carefully designed behavior decomposition follows three general design principles:

1. Capturing multiple motion sensor types (e.g., wearable and object) for predictive mobile analytics tasks
2. Capturing multiple types of data dependencies (e.g., cross-sensor-axial) to create a comprehensive representation of motion sensor signals
3. Decomposing human behaviors into interpretable intermediate features

These principles can offer scholars valuable references when searching through a solution space to design novel artifacts for selected research inquiries pertaining to cybersecurity, healthcare, and mobile analytics. Table 11 summarizes the framework component each design principle was drawn from, a brief description, the broad body of IS literature to which each principle can offer value, and selected promising classes of research inquiry. We then further elaborate on how these design principles can offer value to each listed body of IS literature.

Cybersecurity. The widespread and rapid proliferation of complex IS has introduced unprecedented benefits to modern society. Unfortunately, these systems are often targeted by malicious cybercriminals for espionage, cyberwarfare, and financial gain. Social engineering is a common method by which hackers circumvent security controls and breach selected technologies. Physical social engineering can have significant ramifications to the core infrastructure of selected IS facilities along various categories (e.g., phishing, phone spoofing). Common security controls to detect deceptive and illicit behavior use a sensor (Nunamaker et al. 2017; Pentland et al. 2017).

However, many attackers commonly employ multiple countermeasures (e.g., disguises, fake badges, etc.) to avoid detection. Design Principle 1 offers scholars an operational principle for simultaneously deploying multiple sensor types (e.g., badge swipe, motion sensor triggers, etc.). Such an approach can result in a layered, defense-in-depth approach to detect and mitigate physical social engineering attacks (e.g., unauthorized access, human behavior logging, etc.).

Table 11. Design Principles Followed by ADLR Framework for Selected Bodies and Classes of IS Research Inquiry

ADLR Framework Component	General Design Principle	Relevant Body of IS Literature	Potential Class of Research Inquiry
Data collection (e.g., wearable and object sensors)	Using multiple sensor types	Cybersecurity	• Detecting physical social engineering attacks • Deception detection
2D Interaction Kernel	Capturing multiple types of data dependencies	Healthcare	• Exploiting multichannel data from EEG/ECG* for seizure or arrhythmia detection
ADL Decomposition	Decomposing human behaviors into interpretable intermediate features	Mobile analytics	• Driver behavior profiling • Mobile phone addiction

Note: EEG = electroencephalogram; ECG = Electrocardiogram

Healthcare. Mobile sensing technologies deployed in numerous Internet of Things (IoT) devices (e.g., iPhone, Apple Watch) are increasingly being equipped with advanced capabilities. As a result, such devices offer significant promise in creating always-on, reliable, remote, fine-grained, and high-quality precision healthcare for critical health applications (e.g., ADLR, fall risk assessment). However, unlike traditional healthcare data sources (e.g., EHR), these devices often generate high volumes of untraditional data with rapid velocity. Consequently, novel approaches to extract salient representations for subsequent analytics is critical to executing better-informed, personalized healthcare decision-making. Design Principle 2 offers a mechanism to automatically extract representations from temporally aligned multichannel data. We developed a 2D Interaction Kernel for CNNs to capture the interaction patterns between different sensors types (wearable and object motion sensors). By extracting the human-object interactions, our ADLR framework can recognize ADLs at all levels more accurately than state-of-the-art models.

As an IT artifact for advanced mobile home care support, our approach can help practitioners understand the hierarchical and sequential sensor patterns for other sensor modalities such as ECG and EEG, where multichannel signals jointly capture heart or brain activities and abnormal signal patterns constitute complex symptoms (e.g., seizure, arrhythmia).

Mobile analytics. The advent of Web 3.0 (mobile, sensor-based, Internet of Things) era has enabled novel approaches to collect granular and timely data and consistently observe and assess human behavior (Chen et al. 2012). In order to harness the value of data volume and granularity, a multilevel analysis should be conducted. Design Principle 3 offers scholars a mechanism to (1) systematically extract interpretable intermediate results for short behavioral patterns, and (2) assess global behavior profiles based on local patterns. For example, a driver's driving behavior can be decomposed to high-level (e.g., driving profile), mid-level (e.g., car handling, speed control, traffic planning), and low-level (e.g., braking patterns, steering patterns, turning patterns, lane change patterns). GPS location information and vehicle electronic control unit recordings can be leveraged to effectively model and profile driving behavior. Analytical models can be selected based on the characteristics of a particular level (e.g., CNN for braking pattern, ensemble learning for high-level profiling). Similarly, decomposition can help understand, model, extract, and assess mobile phone usage for potential addiction prevention and intervention.

Practical Implications

Our case study helped demonstrate the proof of concept of our framework. Our model not only recognizes ADLs more accurately but also provides interpretable intermediate results. We discuss the practical value, utility, and impact on targeted and relevant stakeholders of interest, as required by the computational design science paradigm (Rai 2017). Our framework benefits three categories of stakeholders: health professionals, caregivers, and senior citizens and their families. We describe each in turn.

Health professionals. Motion sensors provide timely, reliable, and granular information compared to tasks performed during clinical visits. However, sensor signals are not intuitively interpretable without signal processing expertise. Our framework automatically extracts interpretable results (e.g., human-object interactions, gestures, and activities) from raw sensor data. Health professionals can leverage these results at any level to augment their decision-making based on their needs. For example, professionals can test the upper-limb function (or examine Parkinson's disease progression) if they recognize deteriorating patterns (e.g., tremors) during an interaction. Erratic or inconsistent pillbox-related gestures can suggest medical nonadherence. Professionals might infer cognitive decline when unfinished activities or night wandering are observed with increasing frequency.

Caregivers. Home care providers typically monitor care receivers' dietary activities, medication adherence, medical instruction compliance, and safe independent living onsite. However, these manual, scheduled visits cannot guarantee the timely identification of abnormal behavior patterns. The ADL sequence information provided by our framework can help identify anomalies that may notify caregivers to provide timely intervention. For example, irregular medication activities may indicate poor medication compliance (Conn et al. 2015). Information extracted from a care receiver's mobile activity patterns could help caregivers develop personalized care plans.

Senior citizens and their families. Our framework provides accurate and reliable ADLR for home activity monitoring. Disclosing the monitored ADL performance to caregivers and health professionals enables the timely detection of physical or cognitive impairment and early intervention. This further improves senior citizens' quality of life and reduces families' excessive financial loss caused by the onset of preventable conditions. Our framework can be integrated into smart home environments, online monitoring portals, and notification/reporting services to help relieve family members' concerns about the health and safety conditions of their loved ones.

Conclusion and Future Directions

Ensuring the health and safety of senior citizens living alone is a growing societal concern. Monitoring their Activity of Daily Living (ADL) performance with motion sensors has emerged as a novel approach to collect timely data for diagnosis and care. However, prevailing ADL recognition methods often fail to identify how senior citizens interact with the environment and capture insufficient information to accurately model ADLs level by level, preventing health progression monitoring at varying ADL granularities. These limitations have motivated IS scholars to search for and design alternative ADLR models.

In this study, we adopted the computational design science paradigm to develop a novel deep learning-based, hierarchical, multiphase ADLR framework to address these issues. Our framework makes three key contributions. First, it simultaneously leverages human and object motion-sensor data to capture more motion clues on how humans interact with objects during ADLs, enabling end-to-end ADLR. Second, a novel 2D interaction CNN is designed to automatically extract cross-sensor axial dependencies. This helps capture salient motion features that are missed by extant ADLR models and enables accurate ADLR at all levels. Finally, the framework provides interpretable intermediate results for all ADL levels, which could help healthcare professionals obtain indicators of a patient's physical (e.g., the ability to perform a certain interaction or gesture) and cognitive conditions (e.g., the ability to plan for complex HL-ADL). We rigorously evaluated the framework and its components against prevailing feature-engineering and deep learning-based ADLR models and validated them on selected real-life datasets. We further demonstrated the practical value of the framework with an end-to-end case study. Apart from the significant practical implications, our ADLR framework follows three design principles that could be implemented in future health, cybersecurity, and mobile analytics applications.

This work has several natural extensions. First, future work could monitor longer activity patterns (e.g., weeks, months, or years), associate changes in ADL patterns with disease progression, and identify behavioral anomalies. Such research would have strong healthcare relevance and greatly interest therapists and caregivers. Second, fusing motion sensor data with other sensor types (e.g., biophysical) could help construct more comprehensive ADL signatures. Third, individuals may have different behavioral habits when interacting with objects. Using human-object interaction as a signature to recognize an ADL performer would be an intriguing option for multiresident assisted living facilities. Finally, this sensor-interaction-based framework could be generalized to recognize patterns from other sensor networks whose signals may interact—for example, recognizing heart failure or seizures from ECG and EEG data. Each of these research avenues could potentially provide more fine-grained activity and health profiles to ensure the health and safety of senior citizens living independently.

References

Adipat, B., Zhang, D., and Zhou, L. 2011. "The Effects of Tree-View Based Presentation Adaption on Mobile Web Browsing," *MIS Quarterly* (35:1), pp. 99-122.

Adjerid, I., Adler-Milstein, J., and Angst, C. 2018. "Reducing Medicare Spending Through Electronic Health Information Exchange: The Role of Incentives and Exchange Maturity," *Information Systems Research* 29 (2) 341-361

Almaslukh, B., Jalal, A., and Abdelmonim, A. 2017. "An Effective Deep Autoencoder Approach for Online Smartphone-Based Human Activity Recognition," *International Journal of Computer Science and Network Security* (16:3), 197-205.

Anderson, C. L., and Agarwal, R. 2011. "The Digitization of Healthcare: Boundary Risks, Emotion, and Consumer Willingness to Disclose Personal Health Information," *Information Systems Research* (22:3), pp. 469-490.

Angst, C. M., Block, E. S., D'Arcy, J., and Kelley, K. 2017. "When Do IT Security Investments Matter? Accounting for the Influence of Institutional Factors in the Context of Healthcare Data Breaches," *MIS Quarterly* (41:3), pp. 893-916.

Atallah, L., Lo, B., King, R., and Yang, G.-Z. 2011. "Sensor Positioning for Activity Recognition Using Wearable Accelerometers," *IEEE Transactions on Biomedical Circuits and Systems* (5:4), pp. 320-329.

Avilés-Cruz, C., Ferreyra-Ramírez, A., Zúñiga-López, A., and Villegas-Cortéz, J. 2019. "Coarse-Fine Convolutional Deep-Learning Strategy for Human Activity Recognition," *Sensors* (19:7), Article 1556.

Ayabakan, S., Bardhan, I., Zheng, Z. (Eric), and Kirksey, K. 2017. "The Impact of Health Information Sharing on Duplicate Testing," *MIS Quarterly* (41:4), pp. 1083-1103.

Bao, L., and Intille, S. S. 2004. "Activity Recognition from User-Annotated Acceleration Data, in *Proceedings of the International Conference on Pervasive Computing,* Vienna, Austria.

Bardhan, I., Oh, J. (Cath), Zheng, Z. (Eric), and Kirksey, K. 2015. "Predictive Analytics for Readmission of Patients with Congestive Heart Failure," *Information Systems Research* (26:1), pp. 19-39.

Blei, D. M., Ng, A. Y., and Jordan, M. I. 2003. "Latent Dirichlet Allocation," *Journal of Machine Learning Research* (3), pp. 993-1022.

Bravo, J., Cook, D., and Riva, G. 2016. "Ambient Intelligence for Health Environments," *Journal of Biomedical Informatics* (64), pp. 207-210.

Bryant, M. S., Rintala, D. H., Hou, J.-G., and Protas, E. J. 2015. "Relationship of Falls and Fear of Falling to Activity Limitations and Physical Inactivity in Parkinson's Disease," *Journal of Aging and Physical Activity* (23:2), pp. 187-193.

Cao, H., Nguyen, M. N., Phua, C., Krishnaswamy, S., and Li, X. 2012. "An Integrated Framework for Human Activity Classification," in *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, pp. 331-340.

Census Bureau. 2016. "American Community Survey (ACS)." United States Census Bureau (https://www.census.gov/programs-surveys/acs/data/pums.html).

Census Bureau. 2017. "The Nation's Older Population Is Still Growing" United States Census Bureau (https://www.census.gov/newsroom/press-releases/2017/cb17-100.htm).

Chen, Y., and Xue, Y. 2015. "A Deep Learning Approach to Human Activity Recognition Based on Single Accelerometer," in *2015 IEEE International Conference on Systems, Man, and Cybernetics*, pp. 1488-1492.

Cho, K., van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. 2014. "Learning Phrase Representations Using RNN Encoder-Decoder for Statistical Machine Translation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pp. 1724-1734.

Chollet, F. 2015. "Keras," GitHub (https://github.com/keras-team/keras).

Chowdhury, S. D., Bhattacharya, U., and Parui, S. K. 2013. "Online Handwriting Recognition Using Levenshtein Distance Metric," in *2013 12th International Conference on Document Analysis and Recognition*, IEEE, August, pp. 79-83.

Chuang, J., Maimoon, L., Yu, S., Zhu, H., Nybroe, C., Hsiao, O., Li, S.-H., Lu, H., and Chen, H. 2015. "SilverLink: Smart Home Health Monitoring for Senior Care," in *International Conference on Smart Health 2015,* pp. 3-14.

Chung, J., Ozkaynak, M., and Demiris, G. 2017. "Examining Daily Activity Routines of Older Adults Using Workflow," *Journal of Biomedical Informatics* (71), pp. 82-90.

Conn, V. S., Ruppar, T. M., Chase, J.-A. D., Enriquez, M., and Cooper, P. S. 2015. "Interventions to Improve Medication Adherence in Hypertensive Patients: Systematic Review and Meta-Analysis," *Current Hypertension Reports* (17:12), Article 94.

Cremonesi, P., Koren, Y., and Turrin, R. 2010. "Performance of Recommender Algorithms on Top-n Recommendation Tasks," in *Proceedings of the Fourth ACM Conference on Recommender Systems,* pp. 39-46.

Emi, I. A., and Stankovic, J. A. 2015. "SARRIMA: Smart ADL Recognizer and Resident Identifier in Multi-Resident Accommodations," in *Proceedings of the Conference on Wireless Health,* Article 4.

Fan, G. C., Fitriani, and Goh, W.-B. 2011. "Generic Motion Gesture Detection Scheme Using Only a Triaxial Accelerometer," in *2011 IEEE 15th International Symposium on Consumer Electronics (ISCE)*, IEEE, pp. 151-155.

Forman, G. 2003. "An Extensive Empirical Study of Feature Selection Metrics for Text Classification," *Journal of Machine Learning Research* (3), pp. 1289-1305.

Foti, D., and Koketsu, J. S. 2013. "Activities of Daily Living," in *Pedretti's Occupational Therapy: Practical Skills for Physical Dysfunction,* 7th ed, H. McHugh Pendleton and W. Schultz-Krohn (eds.), Amsterdam: Elsevier, pp. 157-232.

Gong, J., Lach, J., Stankovic, J. A., Rose, K. M., Emi, I. A., Specht, J. P., Hoque, E., Fan, D., Dandu, S. R., Dickerson, R. F., and Perkhounkova, Y. 2015. "Home Wireless Sensing System for Monitoring Nighttime Agitation and

Incontinence in Patients with Alzheimer's Disease," in *Proceedings of the Conference on Wireless Health,* Article 5.

Goodfellow, I., Bengio, Y., Courville, A., and Bengio, Y. 2016. *Deep Learning*, vol. 1, Cambridge, MA: MIT Press.

Gregor, S., and Hevner, A. R. 2013. "Positioning and Presenting Design Science Research for Maximum Impact," *MIS Quarterly* (37:2), pp. 337-355.

Haghi, M., Thurow, K., and Stoll, R. 2017. "Wearable Devices in Medical Internet of Things: Scientific Research and Commercially Available Devices," *Healthcare Informatics Research* (23:1), pp. 4-15.

Hammerla, N. Y., Halloran, S., and Ploetz, T. 2016. "Deep, Convolutional, and Recurrent Models for Human Activity Recognition Using Wearables," in *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, pp. 1533-1540

Hardy, S. E. 2014. "Consideration of Function & Functional Decline," in *Current Diagnosis & Treatment: Geriatrics*, 2nd ed., B. A. Williams, A. Chang, C. Ahalt, H. Chen, R. Conant, C. S. Landefeld, C. Ritchie, and M. Yukawa (eds.), New York: McGraw-Hill, pp. 3-4.

Hassan, M. M., Uddin, M. Z., Mohamed, A., and Almogren, A. 2018. "A Robust Human Activity Recognition System Using Smartphone Sensors and Deep Learning," *Future Generation Computer Systems* (81), pp. 307-313.

Hevner, A. R., March, S. T., Park, J., and Ram, S. 2004. "Design Science in Information Systems Research," *MIS Quarterly* (28:1), pp. 75-105.

Huang, Y., Wang, W., and Wang, L. 2015. "Bidirectional Recurrent Convolutional Networks for Multi-Frame Super-Resolution," in *Proceedings of the 28th International Conference on Neural Information Processing Systems (NIPS)*, pp. 235-243.

Huynh, T., Fritz, M., and Schiele, B. 2008. "Discovery of Activity Patterns Using Topic Models," in *Proceedings of the 10th International Conference on Ubiquitous Computing*, pp. 10-18

Ihianle, I. K., Naeem, U., and Tawil, A.-R. 2016. "Recognition of Activities of Daily Living from Topic Model," *Procedia Computer Science* (98), pp. 24-31.

Jekel, K., Damian, M., Wattmo, C., Hausner, L., Bullock, R., Connelly, P. J., Dubois, B., Eriksdotter, M., Ewers, M., Graessel, E., Kramberger, M. G., Law, E., Mecocci, P., Molinuevo, J. L., Nygård, L., Olde-Rikkert, M. G., Orgogozo, J.-M., Pasquier, F., Peres, K., Salmon, E., Sikkes, S. A., Sobow, T., Spiegel, R., Tsolaki, M., Winblad, B., and Frölich, L. 2015. "Mild Cognitive Impairment and Deficits in Instrumental Activities of Daily Living: A Systematic Review," *Alzheimer's Research & Therapy* (7:1), Article 17.

Kankanhalli, A., Ye, H., and Hai Teo, H. 2015. "Comparing Potential and Actual Innovators: An Empirical Study of Mobile Data Services Innovation," *MIS Quarterly* (39:3), pp. 667-682.

Katz, S. 1983. "Assessing Self-Maintenance: Activities of Daily Living, Mobility, and Instrumental Activities of Daily Living," *Journal of the American Geriatrics Society* (31:12), pp. 721-727.

Kelly, D. 2007. "Methods for Evaluating Interactive Information Retrieval Systems with Users," *Foundations and Trends® in Information Retrieval* (3:1-2), pp. 1-224.

Kim, J.-H., Hong, G.-S., Kim, B.-G., and Dogra, D. P. 2018. "DeepGesture: Deep Learning-Based Gesture Recognition Scheme Using Motion Sensors," *Displays* (55), pp. 38-45.

Kuhn, H. W. 1955. "The Hungarian Method for the Assignment Problem," *Naval Research Logistics Quarterly* (2:1-2), pp. 83-97.

Kwon, H. E., So, H., Han, S. P., and Oh, W. 2016. "Excessive Dependence on Mobile Social Apps: A Rational Addiction Perspective," *Information Systems Research* (27:4), pp. 919-939.

Kwon, J., and Johnson, M. E. 2014. "Proactive Versus Reactive Security Investments in the Healthcare Sector," *MIS Quarterly* (38:2), pp. 451-471.

Lebedev, S. 2016. "HMMlearn," GitHub (https://github.com/ hmmlearn/ hmmlearn).

LeCun, Y., Bengio, Y., and Hinton, G. 2015. "Deep Learning," *Nature* (521:7553), pp. 436-444.

Levenshtein, V. I. 1966. "Binary Codes Capable of Correcting Deletions, Insertions, and Reversals," *Soviet Physics Doklady* (10:8), pp. 707-710.

Li, F., Shirahama, K., Nisar, M., Köping, L., and Grzegorzek, M. 2018. "Comparison of Feature Learning Methods for Human Activity Recognition Using Wearable Sensors," *Sensors* (18:3), Article 679.

Li, X.-B., and Qin, J. 2017. "Anonymizing and Sharing Medical Text Records," *Information Systems Research* (28:2), pp. 332-352.

Lin, Y.-K., Chen, H., Brown, R. A., Li, S.-H., and Yang, H.-J. 2017. "Healthcare Predictive Analytics For Risk Profiling In Chronic Care: A Bayesian Multitask Learning Approach," *MIS Quarterly* (41:2), pp. 473-495.

Lipton, Z. C., Berkowitz, J., and Elkan, C. 2015. "A Critical Review of Recurrent Neural Networks for Sequence Learning" (https://arxiv.org/pdf/1506.00019.pdf).

Liu, Q., Zhou, Z., Shakya, S. R., Uduthalapally, P., Qiao, M., and Sung, A. H. 2018. "Smartphone Sensor-Based Activity Recognition by Using Machine Learning and Deep Learning Algorithms," *International Journal of Machine Learning and Computing* (8:2), pp. 121-126.

Maimoon, L., Chuang, J., Zhu, H., Yu, S., Peng, K.-S., Prayakarao, R., Bai, J., Zeng, D., Li, S.-H., Lu, H., and Chen, H. 2016. "SilverLink: Developing an International Smart and Connected Home Monitoring System for Senior Care," in *International Conference on Smart Health 2016*, pp. 65-77.

Millman, K. J., and Aivazis, M. 2011. "Python for Scientists and Engineers," *Computing in Science & Engineering* (13:2), pp. 9-12.

Mukhopadhyay, T., Singh, P., and Kim, S. H. 2011. "Learning Curves of Agents with Diverse Skills in Information Technology-Enabled Physician Referral Systems," *Information Systems Research* (22:3), pp. 586-605.

Murad, A., and Pyun, J.-Y. 2017. "Deep Recurrent Neural Networks for Human Activity Recognition," *Sensors*

(17:11), Article 2556.

Nunamaker, J. F., Twyman, N. W., Giboney, J. S., and Briggs, R. O. 2017. "Creating High-Value Real-World Impact through Systematic Programs of Research," *MIS Quarterly* (41:2), pp. 335-351.

Nunamaker Jr, J. F., Chen, M., and Purdin, T. D. M. 1990. "Systems Development in Information Systems Research," *Journal of Management Information Systems* (7:3), pp. 89-106.

Oborn, E., Barrett, M., and Davidson, E. 2011. "Unity in Diversity: Electronic Patient Record Use in Multidisciplinary Practice," *Information Systems Research* (22:3), pp. 547-564.

Ordóñez, F., and Roggen, D. 2016. "Deep Convolutional and LSTM Recurrent Neural Networks for Multimodal Wearable Activity Recognition," *Sensors* (16:12), Article 115.

Ozdemir, Z., Barron, J., and Bandyopadhyay, S. 2011. "An Analysis of the Adoption of Digital Health Records Under Switching Costs," *Information Systems Research* (22:3), pp. 491-503.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., and others. 2011. "Scikit-Learn: Machine Learning in Python," *Journal of Machine Learning Research* (12), pp. 2825-2830.

Pentland, S. J., Twyman, N. W., Burgoon, J. K., Nunamaker, J. F., and Diller, C. B. R. 2017. "A Video-Based Screening System for Automated Risk Assessment Using Nuanced Facial Features," *Journal of Management Information Systems* (34:4), pp. 970-993.

Pires, I. M., Garcia, N. M., Pombo, N., Flórez-Revuelta, F., Spinsante, S., and Teixeira, M. C. 2018. "Identification of Activities of Daily Living through Data Fusion on Motion and Magnetic Sensors Embedded on Mobile Devices," *Pervasive and Mobile Computing* (47), pp. 78-93.

Rai, A. 2017. "Editor's Comments: Diversity of Design Science Research," *MIS Quarterly* (41:1), iii-xviii.

Řehůřek, R., and Sojka, P. 2010. "Software Framework for Topic Modelling with Large Corpora," in *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pp. 45-50.

Reyes-Ortiz, J.-L., Oneto, L., Samà, A., Parra, X., and Anguita, D. 2016. "Transition-Aware Human Activity Recognition Using Smartphones," *Neurocomputing* (171), pp. 754-767.

Roggen, D., Calatroni, A., Rossi, M., Holleczek, T., Forster, K., Troster, G., Lukowicz, P., Bannach, D., Pirkl, G., Ferscha, A., Doppler, J., Holzmann, C., Kurz, M., Holl, G., Chavarriaga, R., Sagha, H., Bayati, H., Creatura, M., and Millan, J. del R. 2010. "Collecting Complex Activity Datasets in Highly Rich Networked Sensor Environments," in *Proceedings of the 7th International Conference on Networked Sensing Systems (INSS)*, pp. 233-240.

Safi, K., Mohammed, S., Attal, F., Khalil, M., and Amirat, Y. 2016. "Recognition of Different Daily Living Activities Using Hidden Markov Model Regression," in *Proceedings of the 3rd Middle East Conference on Biomedical Engineering (MECBME)*, pp. 16-19.

Salge, T. O., Kohli, R., and Barrett, M. 2015. "Investing in Information Systems: On the Behavioral and Institutional Search Mechanisms Underpinning Hospitals' Is Investment Decisions.," *MIS Quarterly* (39:1), pp. 61-90.

Silva, B. M. C., Rodrigues, J. J. P. C., de la Torre Díez, I., López-Coronado, M., and Saleem, K. 2015. "Mobile-Health: A Review of Current State in 2015," *Journal of Biomedical Informatics* (56), pp. 265-272.

Singh, I., Varanasi, A., and Williamson, K. 2014. "Assessment and Management of Dementia in the General Hospital Setting," *Reviews in Clinical Gerontology* (24:03), pp. 205-218.

Sun, J., Fu, Y., Li, S., He, J., Xu, C., and Tan, L. 2018. "Sequential Human Activity Recognition Based on Deep Convolutional Network and Extreme Learning Machine Using Wearable Sensors," *Journal of Sensors* (2018), Article 8580959.

Sutskever, I., Vinyals, O., and Le, Q. V. 2014. "Sequence to Sequence Learning with Neural Networks," in *Proceedings of the 27th International Conference on Neural Information Processing Systems*, pp. 3104-3112.

Venkatesh, V., Rai, A., Sykes, T. A., and Aljafari, R. 2016. "Combating Infant Mortality in Rural India: Evidence from a Field Study of EHealth Kiosk Implementations," *MIS Quarterly* (40:2), pp. 353-380.

Wang, J., Chen, Y., Hao, S., Peng, X., and Hu, L. 2019. "Deep Learning for Sensor-Based Activity Recognition: A Survey," *Pattern Recognition Letters* (119), pp. 3-11.

White, R. J. 2018. "Using Topic Models to Detect Behaviour Patterns for Healthcare Monitoring," unpublished doctoral dissertation, University of Reading, Reading, UK.

Whitehead, P. J., Worthington, E. J., Parry, R. H., Walker, M. F., and Drummond, A. E. R. 2015. "Interventions to Reduce Dependency in Personal Activities of Daily Living in Community Dwelling Adults Who Use Homecare Services: A Systematic Review," *Clinical Rehabilitation* (29:11), pp. 1064-1076.

World Health Organization. 2016. "Life Expectancy Increased by 5 Years since 2000, but Health Inequalities Persist." (http://www.who.int/en/news-room/detail/19-05-2016-life-expectancy-increased-by-5-years-since-2000-but-health-inequalities-persist).

Yang, J. B., Nguyen, M. N., San, P. P., Li, X. L., and Shonali, K. 2015. "Deep Convolutional Neural Networks on Multichannel Time Series for Human Activity Recognition," in *Proceedings of the 24th International Conference on Artificial Intelligence (IJCAI)*, pp. 3995-4001.

Ye, H. (Jonathan), and Kankanhalli, A. 2018. "User Service Innovation on Mobile Phone Platforms: Investigating Impacts of Lead Userness, Toolkit Support, and Design Autonomy," *MIS Quarterly* (42:1), pp. 165-187.

Yu, S., Chen, H., and Brown, R. A. 2017. "Hidden Markov Model-Based Fall Detection with Motion Sensor Orientation Calibration: A Case for Real-Life Home Monitoring," *IEEE Journal of Biomedical and Health Informatics* (22:6), pp. 1847-1853.

Zeng, M., Nguyen, L. T., Yu, B., Mengshoel, O. J., Zhu, J., Wu, P., and Zhang, J. 2014. "Convolutional Neural Networks for Human Activity Recognition Using Mobile Sensors," in

Proceedings of the 6th International Conference on Mobile Computing, Applications and Services, pp. 197-205.

Zhu, H., Chen, H., and Brown, R. 2018. "A Sequence-to-Sequence Model-Based Deep Learning Approach for Recognizing Activity of Daily Living for Senior Care," *Journal of Biomedical Informatics* (84), pp. 148-158.

About the Authors

Hongyi Zhu is an assistant professor in the Department of Information Systems and Cyber Security at the College of Business at the University of Texas at San Antonio. He received his Ph.D. in management information systems from the University of Arizona in December 2019. He has primarily worked on designing advanced mobile analytics for smart home care. His research focuses on the recognition, extraction, and analysis of subjects' in-house behaviors (e.g., activities, object usage) from raw mobile sensors data. His work has been published or accepted in journals such as *Journal of Biomedical Informatics, IEEE Intelligent Systems*, *Journal of Management Information Systems,* and others. He has contributed to a variety of projects supported by the National Science Foundation.

Sagar Samtani is currently an assistant professor and Grant Thornton Scholar in the Department of Operations and Decision Technologies at the Kelley School of Business at Indiana University. Samtani graduated with his Ph.D. in Management Information Systems from the University of Arizona's Artificial Intelligence Lab in May 2018 where he served as a Scholarship-for-Service fellow from 2014-2017. Samtani's AI for cybersecurity and dark web analytics research initiatives have garnered nearly $1.5M (in PI and co-PI roles) in prestigious funding from the National Science Foundation CISE Research Initiation Initiative and Cybersecurity Innovation for Cyber Infrastructure programs. His research has been published in journals such as *Journal of Management Information Systems* and *IEEE Intelligent Systems*. His research has also received significant media coverage and citations from outlets such as the *Miami Herald*, *Fox News*, and *Science*. He is a member of the IEEE, ACM, AIS, and INFORMS.

Randall A. Brown received his medical degree from Rush Medical College in Chicago and completed residency training in internal medicine at the University of Michigan in Ann Arbor, MI. He has an MBA from the University of Arizona, Eller College of Management. He has over 25 years of clinical medical experience teaching and practicing medicine in tertiary academic medical centers. He was an assistant professor of medicine from 1989 to 2003 with the Henry Ford Health System in Detroit, MI. At the University of Arizona College of Medicine, he was an assistant professor of medicine, clinical scholar from 2005 to 2018 and a research consultant with the Artificial Intelligence Laboratory of the Eller College of Management, in the MIS Department, from 2010 to 2019. He is currently working on the Virtual College of Pharmacy Project with Department of Pharmacy of the University of Arizona.

Hsinchun Chen is Regents Professor and Thomas R. Brown Chair in Management and Technology in the Management Information Systems Department at the Eller College of Management, University of Arizona. He received his Ph.D. in Information Systems from New York University. He is the author/editor of 20 books, 300 SCI journal articles, and 200 refereed conference articles covering digital library, data/text/web mining, business analytics, security informatics, and health informatics. He served as the lead program director of the Smart and Connected Program at the National Science Foundation (NSF) for 2014-2015, a multi-year multi-agency U.S. health IT research program. He founded the Artificial Intelligence Lab at The University of Arizona in 1989, which has received $50M+ research funding from the NSF, National Institutes of Health, National Library of Medicine, Department of Defense, Department of Justice, Central Intelligence Agency, Department of Homeland Security, and other agencies (100+ grants, 50+ from NSF). He is a Fellow of ACM, IEEE, and AAAS.

Appendix A

Model Specifications

Interaction-Based Convolutional Neural Network (I-CNN)

Figure A1 depicts our I-CNN model. The 2D Interaction Kernel is installed on the first convolutional layer, transforming input data segments into interaction representations. For a faster training/testing speed, we implemented two convolutional layers parsimoniously to extract the temporal local-dependency within the representations. Both convolutional layers are activated with hyperbolic tangent function ($tanh$). Pooling layers are implemented after convolutional layers to condense the representations. Two fully connected (dense) layers and a $softmax$ function classify the representation into five interaction categories. Unlike past studies (Yang et al. 2015), we implemented dropout operations on both pooling and fully connected layers to avoid overfitting. The dropped feature maps and nodes are colored black in Figure A1. I-CNN model parameters are summarized in Table A1.

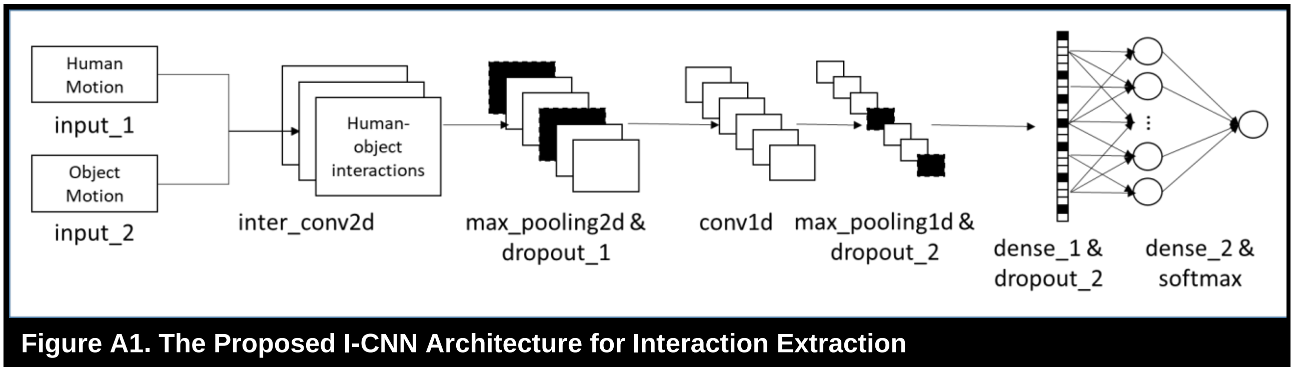


Figure A1. The Proposed I-CNN Architecture for Interaction Extraction

Table A1. I-CNN Model Specifications

Layer	Kernel Size	Stride	Output Shape	Param #
input_1			(None, 3, 240, 1)	0
Input_2			(None, 3, 240, 1)	0
inter_conv2d	(2, 13)	(1, 1)	(None, 9, 228, 10)	270
max_pooling2d	(1, 19)		(None, 9, 12, 10)	0
dropout_1 (rate=0.125)			(None, 9, 12, 10)	0
conv1d	(1, 7)	(1, 1)	(None, 9, 6, 10)	710
max_pooling1d	(3, 3)		(None, 3, 2, 10)	0
dropout_2 (rate=0.125)			(None, 3, 2, 10)	0
flatten			(None, 60)	0
dense_1 (sigmoid)			(None, 40)	2,440
dropout_3 (rate=0.25)			(None, 40)	0
dense_2 (softmax)			(None, 5)	205
Total params: 3,625; trainable params: 3,625				

GRU-Based Seq2Seq HL-ADL Recognition Model

We adopt Zhu et al.'s (2018) S2S_GRU design for our HL-ADL recognition phase (Figure 4). Model parameters are summarized in Table A2.

Table A2. S2S_GRU Model Specifications

Layer	**Output Shape**	**Param #**
gru_1	(None, 40)	6,960
dense_1	(None, 40)	1,640
repeat_vector	(None, 300, 40)	0
gru_2	(None, 300, 40)	9,720
time_distributed	(None, 300, 5)	205
Total params: 18,525; trainable params: 18,525		

Appendix B

OPPO-HL Statistics

OPPO-HL contains 1,135 samples with interweaving HL-ADLs. Table B1 shows the distribution of the segments with N distinct HL-ADL labels. The majority (96.74%) of the dataset contains two to four HL-ADL activities in a segment. Table B2 then summarizes the average proportion (length) of the top-N HL-ADL labels in each segment. From this table, we observe that each segment has on average two major HL-ADLs, taking up 90.11% of the segment's time span. Tables B3 and B4 further demonstrate detailed distributions of these labels. Table B5 sets up baseline Acc@2 scores when assigning the same label to all segments.

Table B1. Distribution of Segments with N Distinct HL-ADL Labels

N	**1**	**2**	**3**	**4**	**5**
Count (%)	27 (2.38%)	205 (18.06%)	606 (53.39%)	287 (25.29%)	10 (0.88%)

Table B2. Average Proportion of Top-N HL-ADL Labels

Top-N Labels	**1**	**2**	**3**	**4**	**5**
Proportion	61.19%	90.11%	98.82%	99.99%	100.00%

Table B3. A Label with the Highest & Second-highest Count

	Relaxing	**Coffee Time**	**Early Morning**	**Cleanup**	**Sandwich Time**
Highest	15.24%	11.89%	28.90%	7.41%	36.56%
Second highest	23.79%	28.01%	12.07%	23.00%	13.13%

Table B4. Label Distribution of the TOP-2 Label Set

Count	**Relaxing**	**Coffee Time**	**Early Morning**	**Cleanup**	**Sandwich Time**
Relaxing	3 (0.26%)				
Coffee Time	2 (0.18%)	0			
Early Morning	232 (20.44%)	219 (19.30%)	0		
Cleanup	134 (11.81%)	0	5 (0.44%)	0	
Sandwich Time	69 (6.08%)	232 (20.44%)	9 (0.79%)	206 (18.15%)	24 (2.11%)

Table B5. Acc@2 Scores When Assigning One Label to All Segments

	Relaxing	**Coffee Time**	**Early Morning**	**Cleanup**	**Sandwich Time**
Acc@2	38.77%	39.91%	40.97%	30.40%	47.58%

Appendix C

Recognizing Gestures by Incorporating Historical Information

In the main text, we noted that certain gestures may closely follow each other in practice (e.g., “close the fridge” shortly after “open the fridge”). These co-occurring and correlated gestures can form local (i.e., short-term) patterns that contribute to gesture recognition or prediction (Chen et al. 2012). To this end, we examine our OPPO-ML dataset for potential gesture co-occurrence.

According to the ADLR hierarchy summarized in Table 1, a gesture typically lasts for less than 15 seconds. Therefore, it is likely to identify at least one gesture in the 15-second time window before a data segment. Gesture labels within 15 seconds prior to each OPPO-ML segment are generated using a one-second stride length sliding window strategy, resulting in a 15-label sequence. A majority vote strategy is used to summarize a gesture label for each sliding window. Since the time between two potentially correlated gestures can vary, we adopt a “bag-of-gestures” approach to model and record the co-occurrence relationship. If a gesture G_j occurred at least once within the 15-second window prior to an OPPO-ML segment whose gesture label is G_i $(i \neq j)$, we increment the co-occurrence count G_{ij} by one. After all the co-occurrences are recorded for OPPO-ML, we calculate the probability that G_j occurred before G_i within the 15-second time window with

$$p_{ij} = \frac{G_{ij}}{\#\ G_i \text{ in OPPO-ML}}.$$

We present the co-occurred gestures within 15 seconds prior to the current gesture with the probability in Table C1.

Table C1. Gestures Co-occurrence Probability within 15 Seconds[+]

Current Gesture*	G1	G2	G3	G4	G5	G6	G7	G8	G9	G10	G11	G12	G13	G14	G15	G16
G1	--	0.03	0.03	0.02	0.01	0.02	0.04	0.14	0.33	0.49	0.55	0.62	0.70	0.71	0.01	0.02
G2	0.66	--	0.02	0.09	0.03	0.04	0.01	0.05	0.07	0.17	0.40	0.50	0.54	0.66	0.01	0.01
G3	0.75	0.72	--	0.01	0.04	0.04	0.01	0.03	0.04	0.04	0.04	0.08	0.23	0.50	0.02	0.02
G4	0.55	0.72	0.66	--	0.03	0.04	0.01	0.01	0.01	0.01	0.02	0.03	0.08	0.13	0.02	0.02
G5	0.00	0.01	0.02	0.04	--	0.07	0.03	0.04	0.02	0.05	0.04	0.05	0.04	0.04	0.41	0.42
G6	0.00	0.01	0.02	0.02	0.91	--	0.04	0.05	0.01	0.01	0.01	0.01	0.02	0.02	0.38	0.39
G7	0.00	0.01	0.00	0.00	0.65	0.67	--	0.05	0.05	0.06	0.05	0.05	0.08	0.09	0.45	0.47
G8	0.00	0.01	0.00	0.00	0.62	0.63	0.81	--	0.03	0.05	0.05	0.05	0.05	0.05	0.31	0.41
G9	0.05	0.05	0.02	0.06	0.54	0.64	0.66	0.65	--	0.02	0.04	0.05	0.07	0.08	0.13	0.27
G10	0.03	0.02	0.02	0.05	0.49	0.56	0.65	0.65	0.98	--	0.04	0.04	0.06	0.07	0.07	0.10
G11	0.03	0.02	0.03	0.06	0.33	0.52	0.57	0.68	0.80	0.82	--	0.01	0.06	0.07	0.07	0.11
G12	0.02	0.00	0.02	0.05	0.17	0.42	0.53	0.60	0.80	0.81	0.96	--	0.05	0.05	0.03	0.08
G13	0.02	0.07	0.02	0.05	0.12	0.29	0.44	0.48	0.69	0.71	0.70	0.72	--	0.00	0.02	0.04
G14	0.02	0.05	0.02	0.04	0.09	0.13	0.28	0.47	0.57	0.71	0.71	0.70	0.96	--	0.02	0.02
G15	0.00	0.00	0.04	0.11	0.03	0.05	0.01	0.01	0.00	0.00	0.00	0.00	0.01	0.01	--	0.42
G16	0.00	0.00	0.00	0.04	0.02	0.03	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.01	0.93	--

Note: [+] The first author can provide the table with other time windows upon request. * G1 = Open Door 1, G2 = Close Door 1, G3 = Open Door 2, G4 = Close Door 2, G5 = Open Fridge, G6 = Close Fridge, G7 = Open Dishwasher, G8 = Close Dishwasher, G9 = Open Drawer 1, G10 = Close Drawer 1, G11 = Open Drawer 2, G12 = Close Drawer 2, G13 = Open Drawer 3, G14 = Close Drawer 3, G15 = Drink from Cup, G16 = Put away Cup.

Several co-occurrence patterns can be identified in Table C1. For example, the “put away cup” gesture is likely to follow the “drink from cup” gesture in the next 15 seconds. Therefore, current gestures may be successfully recognized based on a combination of current lower-level information (i.e., current extracted interactions demonstrated in Experiment 2) and previous mid-level information (i.e., recent gestures). We design and conduct an additional experiment to evaluate the performance gain by including recent gestures as features for gesture recognition. Like Experiment 2, we compare the gesture recognition performance between our proposed I-CNN-GR model and four sets of benchmarks that incorporate *k* most recent gestures (i.e., historical information) as features. Table C2 summarizes the experiment design.

Table C2. Experiment Design: I-CNN-GR vs. Four Benchmark Sets

Benchmark Set	Models*	Testbed	Evaluation Metrics	Prior Study
Interaction + *k* most recent gestures	I-CNN-SVM-k	OPPO-ML	Precision, recall, F$_1$, averaged F$_1$	Chen and Xue 2015 Cao et al. 2012 Bao and Intille 2004
	I-CNN-DT-k			
	I-CNN-NN-k			
Data representation + *k* most recent gestures	CNN-1D-k			
	CNN-2D-k			
Signal Features + *k* most recent gestures	SVM-k + Signal Features			
	DT-k + Signal Features			
k most recent gestures	SVM-k			
	DT-k			

Note: * Model specifications and parameters are detailed in Appendix D.

The first benchmark set consists of I-CNN-GR variations that use classifiers (DT, SVM, and a fully connected neural network (NN)) instead of heuristics to incorporate historical information (i.e., the *k* most recent gestures). This benchmark set classifies OPPO-ML segments based on the interactions extracted by I-CNN and the recent gestures. We extended CNN-1D and CNN-2D, the two deep learning benchmarks we used in Experiment 2, to form our second benchmark set. For both models, we integrated the *k* most recent gestures into the input to the last dense layer. This enables both models to recognize gestures based on the data representation (temporal or cross-axial dependencies) and recent gestures. For the third benchmark set, we incorporated the *k* most recent gestures to complement the signal features used by classical machine learning models (i.e., SVM and DT). The fourth benchmark set consisted of classical machine learning approaches (i.e., SVM and DT) that only use recent *k* gestures as features. Benchmark parameters are summarized in Appendix D. Performances are evaluated using precision, recall, and F$_1$. All models are trained and tested with 10-fold cross-validation. We conduct the above experiments with six different selections of *k* (*k*=1, 5, 10, 15, 20, and 25) for all four benchmark sets. These variations can help to understand optimal look-back time windows that help boost gesture recognition performance. Table C3 summarizes the overall benchmark performances with different *k* selections. The best performance of each model (when *k*>1) is highlighted in boldface.

Table C3. Benchmark Performance (Macro-averaged F$_1$ Score) with *k* Most Recent Gestures

***k* most recent gestures**	**0***	**1**	**5**	**10**	**15**	**20**	**25**
I-CNN-DT-k	0.849	0.829	0.830	**0.830**	0.828	0.828	0.828
I-CNN-SVM-k	0.849	**0.570**	0.176	0.117	0.095	0.084	0.084
I-CNN-NN-k	0.849	**0.321**	0.210	0.206	0.187	0.186	0.168
CNN-1D-k	0.775	0.776	0.764	**0.794**	0.781	0.768	0.743
CNN-2D-k	0.773	0.765	0.778	**0.777**	0.759	0.765	0.754
SVM-k + Signal Features	0.110	0.135	0.135	0.135	0.135	0.135	0.135
DT-k + Signal Features	0.721	0.780	0.829	**0.833**	0.830	0.830	0.830
SVM-k	N/A	N/A	0.624	**0.640**	0.622	0.618	0.609
DT-k	N/A	N/A	0.747	0.749	**0.754**	0.750	0.740

Note: * Macro-averaged F$_1$ score of Experiment 2 benchmarks extracted from Table 8: 0.849 for I-CNN-GR, 0.775 for CNN-1D, 0.773 for CNN-2D, 0.110 for SVM + Signal Features, and 0.721 for DT + Signal Features.

Overall, the performance of the proposed I-CNN-GR method exceeded all methods in each benchmark set. This indicates that including historical information with gesture co-occurrences at any level of *k* decreases the overall gesture recognition performance. The performances for most benchmarks increased until a peak of *k*=10 or *k*=15 (0.830 for I-CNN-DT-10, 0.794 for CNN-1D-10, 0.777 for CNN-2D-10, 0.833 for DT-10 + Signal Features, 0.640 for SVM-10, and 0.754 for DT-10). Performances decreased for all methods when given additional historical information. These concave performances indicate that gestures recognized recently (10 - 15 seconds) have a higher predictive value for gesture recognition models. We further compare the performance of I-CNN-GR against the best performing model in each benchmark set when *k*=10. Paired t-tests are conducted on all macro-averaged F_1 scores. Table C4 summarizes the results, with the best precision, recall, and F_1 for each gesture highlighted in boldface**.**

Table C4. Gesture Recognition Performance of I-CNN-GR vs. Selected Benchmarks Using 10 Recent Gestures

Gestures	**I-CNN-GR**			**I-CNN-DT-10**			**CNN-1D-10**			**DT-10 + Signal Features**			**DT-10**		
	P	**R**	**F_1**	**P**	**R**	**F_1**	**P**	**R**	**F_1**	**P**	**R**	**F_1**	**P**	**R**	**F_1**
Open Door 1	0.85	**0.92**	**0.88**	**0.95**	0.76	0.84	0.73	0.61	0.67	0.79	0.79	0.79	0.79	0.65	0.71
Close Door 1	**1.00**	0.73	0.84	0.82	**0.94**	**0.88**	0.52	0.85	0.65	0.79	0.80	0.79	0.79	0.71	0.75
Open Door 2	**0.92**	**1.00**	**0.96**	0.88	0.82	0.85	0.73	0.76	0.75	0.75	0.76	0.75	0.80	0.73	0.76
Close Door 2	**1.00**	0.82	**0.90**	0.84	**0.88**	0.86	0.77	0.82	0.80	0.77	0.82	0.80	0.80	0.67	0.73
Open Fridge	0.81	0.81	0.81	0.84	0.88	0.86	**0.90**	**0.90**	**0.90**	0.81	0.82	0.81	0.44	0.30	0.36
Close Fridge	0.73	0.80	0.76	0.88	0.83	0.86	0.93	0.84	0.88	0.84	0.83	0.84	**0.96**	**0.84**	**0.89**
Open Dishwasher	0.85	**1.00**	0.92	**0.96**	0.97	**0.96**	0.91	0.90	0.90	0.78	0.75	0.77	0.71	0.59	0.64
Close Dishwasher	0.95	**1.00**	**0.97**	**0.98**	0.94	0.96	0.92	0.95	0.94	0.85	0.83	0.84	0.92	0.73	0.82
Open Drawer 1	0.75	**0.92**	0.83	0.73	0.87	0.79	0.82	0.82	0.82	**0.82**	0.84	**0.83**	0.67	0.58	0.63
Close Drawer 1	0.92	0.73	0.82	0.84	0.69	0.76	0.85	0.75	0.80	0.80	0.78	0.79	**0.93**	**0.87**	**0.90**
Open Drawer 2	0.70	0.70	0.70	0.77	**0.88**	0.82	0.77	0.63	0.69	0.82	0.84	**0.83**	**0.86**	0.72	0.79
Close Drawer 2	0.82	0.82	0.82	0.87	0.73	0.79	0.83	0.59	0.69	0.88	0.86	0.87	**0.96**	**0.91**	**0.93**
Open Drawer 3	0.73	**0.92**	0.82	0.62	0.79	0.69	0.70	0.74	0.72	**0.86**	0.79	**0.82**	0.77	0.64	0.70
Close Drawer 3	0.91	0.71	0.80	0.68	0.64	0.66	0.68	0.75	0.71	0.90	**0.92**	0.91	**0.96**	0.91	**0.93**
Drink from Cup	0.88	0.84	0.86	0.84	**0.93**	0.88	0.91	0.90	0.90	**0.93**	**0.93**	**0.93**	0.36	0.79	0.49
Put away Cup	0.95	0.86	0.90	0.88	0.73	0.80	0.88	0.89	0.89	0.95	0.95	0.95	**1.00**	**0.93**	**0.97**
Averaged F_1 (SD^+)	**0.85** (0.069)			0.83 (0.025)			0.79* (0.027)			0.83 (0.034)			0.75* (0.026)		

Note: * p-value<0.05. +: SD=standard deviation

Overall, I-CNN-GR (proposed method in main text) had the highest performance over the four selected benchmarks. It outperformed CNN-1D-10 and DT-10 with statistically significant margins. DT-10 was able to capture the gesture co-occurrence to make accurate gesture prediction for specific gestures (e.g., "Close Drawer 1"). However, this method cannot consistently detect gestures that do not have a dominant co-occurring gesture. For models that did not leverage human-object interactions (e.g., DT-k + Signal Features), including recently recognized gestures resulted in more accurate gesture recognition. For example, the DT-10 + Signal Features model improved significantly compared to the original DT + Signal Features model to achieve comparable gesture recognition performance to I-CNN-GR. However, I-CNN-GR's variations did not outperform the original heuristic-based approach, suggesting that the local gesture correlation is not as informative as the proposed interaction-gesture mapping heuristics. The included recent gestures diluted the predictive power of extracted human-object interactions for I-CNN-SVM-k and I-CNN-NN-k. This indicates that the co-occurred recent gestures can introduce noise to the predictive model. Future researchers could explore and develop effective and systematic mechanisms to selectively include the most informative recent gesture information (e.g., systematically and computationally prune irrelevant information) that helps improve overall predictive performance.

Appendix D

Benchmark Model Specifications

Signal Features Extraction

Classical machine-learning-based ADLR approaches rely on two classes of manually engineered signal features: temporal and frequency/spectral features (Bao and Intille 2004). For each sensor channel, the mean, minimum, maximum, and standard deviation of the data segment are extracted as temporal features. Energy and spectral entropy are extracted as frequency/spectral features. The energy of the sequence $X = \{x_1, x_2, \dots, x_t\}$ is defined as the sum of its squared Fast Fourier Transformation (FFT) coefficients. Spectral Entropy is defined as the Shannon Entropy of the Power Spectral Density (PSD) of the sequence X. Temporal features and energy are calculated with the NumPy package (Oliphant 2006); FFT is implemented using the SciPy package (Millman and Aivazis 2011); and spectral energy is implemented with the EntroPy package (Vallat 2018).

CNN-1D

We implemented CNN-1D (Figure D1) with a model structure similar to I-CNN (Table A1) for a controlled experiment. The first convolution layer has a 1D kernel as adopted by Zeng et al. (2014) and Yang et al. (2015). The model parameters are summarized in Table D1.

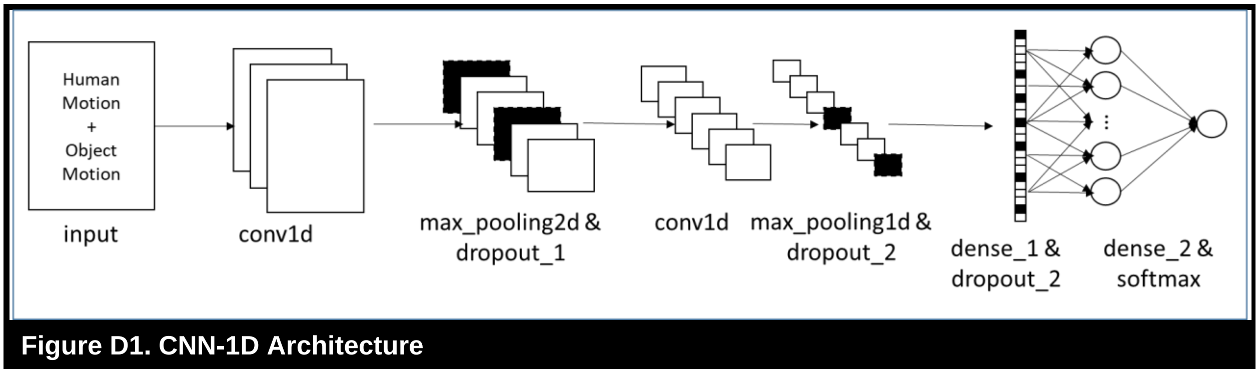

Human Motion + Object Motion
input
conv1d
max_pooling2d & dropout_1
conv1d
max_pooling1d & dropout_2
dense_1 & dropout_2
dense_2 & softmax

Figure D1. CNN-1D Architecture

Table D1. CNN-1D Model Specifications

Layer	Kernel Size	Stride	Output Shape	Param #
input			(None, 27, 240, 1)	0
conv1d_1	(1, 13)	(1, 1)	(None, 27, 228, 10)	140
max_pooling2d_1	(3, 19)		(None, 9, 12, 10)	0
dropout_1 (rate=0.125)			(None, 9, 12, 10)	0
conv1d_2	(1, 7)	(1, 1)	(None, 9, 6, 10)	710
max_pooling2d_2	(3, 3)		(None, 3, 2, 10)	0
dropout_2 (rate=0.125)			(None, 3, 2, 10)	0
flatten			(None, 60)	0
dense_1 (sigmoid)			(None, 40)	2,440
dropout_3 (rate=0.25)			(None, 40)	0
dense_2 (softmax)			(None, 16)	656
Total params: 3,946; trainable params: 3,946				

CNN-2D

We implemented a 2D kernel-based CNN benchmark for gesture recognition following Chen and Xue's design (2015). CNN-2D shares a similar design with I-CNN for our controlled experiment. CNN-2D parameters are summarized in Table D2.

Table D2. CNN-2D Model Specifications

Layer	Kernel Size	Stride	Output Shape	Param #
input			(None, 27, 240, 1)	0
conv2d_1	(2, 13)	(1, 1)	(None, 26, 228, 10)	270
max_pooling2d_1	(2, 19)		(None, 13, 12, 10)	0
dropout_1 (rate=0.125)			(None, 13, 12, 10)	0
conv2d_2	(2, 7)	(1, 1)	(None, 12, 6, 10)	1,410
max_pooling2d_2	(6, 3)		(None, 2, 2, 10)	0
dropout_2 (rate=0.125)			(None, 2, 2, 10)	0
flatten			(None, 40)	0
dense_1 (sigmoid)			(None, 40)	1,640
dropout_3 (rate=0.25)			(None, 40)	0
dense_2 (softmax)			(None, 16)	656
Total params: 3,976; trainable params: 3,976				

DeepConvLSTM

We implemented the DeepConvLSTM model by Ordóñez and Roggen (2016) as a gesture recognition benchmark. The original DeepConvLSTM model is comprised of four convolutional layers and two LSTM layers, resulting in over 3.9 million training parameters, which is overly complicated for training/testing and incomparable with our proposed models and other gesture recognition benchmarks (~4k parameters). Therefore, while keeping the convolution-recurrent network structure, we control the parameter size by reducing the number of convolutional layers and the dimensionality of the LSTM layers in the DeepConvLSTM model. The final model specifications are summarized in Table D3.

Table D3. DeepConvLSTM Model Specifications

Layer	Kernel Size	Stride	Output Shape	Param #
input			(None, 27, 240, 1)	0
conv1d_1	(1, 5)	(1, 1)	(None, 27, 236, 10)	60
conv1d_2	(1, 5)	(1, 1)	(None, 27, 232, 10)	510
permute			(None, 232, 27, 10)	0
reshape			(None, 232, 270)	0
lstm_1			(None, 232, 2)	2,184
lstm_2			(None, 14)	952
dense (softmax)			(None, 16)	240
Total params: 3,946; trainable params: 3,946				

I-CNN-DT-k

The I-CNN-DT benchmark has two components: I-CNN for interaction extraction and DT for gesture classification. I-CNN outputs the extracted interaction and the corresponding probability from a human-object sensor pair. All interactions and probabilities together with the *k* most recent gestures form the features for the DT classifier. The DT classifier is implemented with scikit-learn (Pedregosa et al. 2011) using default configurations.

I-CNN-SVM-k

We replace the DT classifier in I-CNN-DT with an SVM classifier. SVM is implemented with scikit-learn (Pedregosa et al. 2011) using an RBF kernel and default configurations.

I-CNN-NN-k

We replace the DT classifier in I-CNN-DT with a fully connected neural network. The network specifications are summarized in Table D4 for *k*=10.

Table D4. I-CNN-NN-10 Model Specifications

Layer	Output Shape	Param #
input	(None, 26)	0
dense (softmax)	(None, 16)	432
Total params: 432; trainable params: 432		

CNN-1D-k

We modify the CNN-1D benchmark model to incorporate the *k* most recent gestures. The gestures are concatenated with the dense data representation output (dropout_3). The concatenated representation is then classified by a fully connected layer (dense_2). The parameters are summarized below in Table D5 for *k*=10.

Table D5. CNN-1D-10 Model Specifications

Layer	Kernel Size	Stride	Output Shape	Param #
input_1			(None, 27, 240, 1)	0
conv1d_1	(1, 13)	(1, 1)	(None, 27, 228, 10)	140
max_pooling2d_1	(3, 19)		(None, 9, 12, 10)	0
dropout_1 (rate=0.125)			(None, 9, 12, 10)	0
conv1d_2	(1, 7)	(1, 1)	(None, 9, 6, 10)	710
max_pooling2d_2	(3, 3)		(None, 3, 2, 10)	0
dropout_2 (rate=0.125)			(None, 3, 2, 10)	0
flatten_1			(None, 60)	0
dense_1 (sigmoid)			(None, 40)	2,440
dropout_3 (rate=0.25)			(None, 40)	0
input_2			(None, 10, 1)	0
flatten_2			(None, 10)	0
concatenate			(None, 50)	0
dense_2 (softmax)			(None, 16)	816
Total params: 4,106; trainable params: 4,106				

CNN-2D-*k*

Following the same process to include *k* recent gestures as in CNN-1D, we concatenate the *k* gestures with the data representation output (dropout_3) for classification. Parameters are summarized in Table D6 for *k*=10.

Table D6. CNN-2D-10 Model Specifications

Layer	Kernel Size	Stride	Output Shape	Param #
input_1			(None, 27, 240, 1)	0
conv2d_1	(2, 13)	(1, 1)	(None, 26, 228, 10)	270
max_pooling2d_1	(2, 19)		(None, 13, 12, 10)	0
dropout_1 (rate=0.125)			(None, 13, 12, 10)	0
conv2d_2	(2, 7)	(1, 1)	(None, 12, 6, 10)	1,410
max_pooling2d_2	(6, 3)		(None, 2, 2, 10)	0
dropout_2 (rate=0.125)			(None, 2, 2, 10)	0
flatten_2			(None, 40)	0
dense_1 (sigmoid)			(None, 40)	1,640
dropout_3 (rate=0.25)			(None, 40)	0
input_2			(None, 10, 1)	0
flatten_2			(None, 10)	0
concatenate			(None, 50)	0
dense_2 (softmax)			(None, 16)	816
Total params: 4,136; trainable params: 4,136				

LSTM-Based Seq2Seq HL-ADL Recognition Model

To compare the HL-ADL recognition performance of the GRU cell against the LSTM cell, we implemented an alternate LSTM-based Seq2Seq model with the same model structure (Figure 4). The parameters are summarized in Table D7.

Table D7. S2S_LSTM Model Specifications

Layer	Output Shape	Param #
lstm_1	(None, 40)	9,280
dense_1	(None, 40)	1,640
repeat_vector	(None, 300, 40)	0
lstm_2	(None, 300, 40)	12,960
time_distributed	(None, 300, 5)	205
Total params: 24,085; trainable params: 24,085		

Topic Modeling-based ADL Recognition Model

Based on Huynh et al. (2008), signal features were extracted from OPPO-HL’s raw sensor data. A Naïve Bayes classifier (NB) was trained with 10-fold cross-validation to classify data samples into ML-ADL labels, with a non-overlapping sliding window of one second. Using the ML-ADL label sequence as corpora, a Latent Dirichlet Allocation-based topic model (Blei et al. 2003) then models similar label sequence patterns as topics. Since these topics are learned with unsupervised methods, we adopted the Hungarian algorithm (Kuhn 1955) to map them to the five HL-ADL labels. Finally, each segment obtained an HL-ADL label characterized by its most likely activity topic.

During the LDA learning process, we experimented with 5, 10, 15, and 20 topics. We observed that the generated topics were highly similar, and less than five topics were populated as the most likely topic of each OPPO-HL segment. Therefore, we present the top-10 activities that characterize each topic in Table D8. The last row of Table D8 is the topic-label assignment calculated by the Hungarian algorithm (e.g., Topic 1 was assigned the "Early Morning" HL-ADL label).

Table D8. Activity Topics Extracted by LDA Model (N=5)

	Topic 1	**Topic 2**	**Topic 3**	**Topic 4**	**Topic 5**
A1	Relax	Relax	Relax	Relax	Relax
A2	Open fridge	Open fridge	Pick up cup	Open door 2	Pick up cup
A3	Pick up cup	Pick up cup	Close fridge	Close drawer 2	Open fridge
A4	Close fridge	Close fridge	Put down cup	Close fridge	Close fridge
A5	Put down cup	Put down cup	Open fridge	Open door 1	Open dishwasher
A6	Open door 2	Open door 2	Close dishwasher	Open fridge	Put down cup
A7	Open door 1	Open dishwasher	Open drawer 3	Close door 1	Open drawer 3
A8	Open dishwasher	Open drawer 3	Open dishwasher	Close drawer 3	Open door 1
A9	Close dishwasher	Close door 1	Open door 2	Open drawer 3	Open door 2
A10	Open drawer 3	Close dishwasher	Open door 1	Pick up cup	Close door 2
HL-ADL Assignment	Early morning	Clean up	Sandwich time	Relaxing	N/A

Stacked AutoEncoder-based HL-ADL Recognition Model

The Stacked AutoEncoder (SAE) model proposed by Almaslukh et al. (2017) implemented two dense layers to learn the condensed data representation. However, due to our input data size (243,000 values), implementing a dense layer with 80 hidden neurons as the first layer will result in 19.44 million trainable parameters for both this encoding and as its corresponding decoding layer. We leveraged CNNs' weight-sharing characteristics (Goodfellow et al. 2016) and implemented a 2D convolution layer as the first encoding layer. Model parameters are summarized in Table D9. Mean Squared Error (MSE) was used to measure the reconstruction error.

Table D9. Stacked AutoEncoder Model Specifications

Layer	**Kernel Size**	**Stride**	**Output Shape**	**Param #**
input			(None, 27, 9,000, 1)	0
conv2d	(27, 30)	(1, 30)	(None, 1, 300, 1)	811
flatten			(None, 300)	
dense_1			(None, 10)	3,010
dense_2			(None, 300)	3,300
reshape			(None, 1, 300, 1)	0
conv2d_transpose	(27, 30)	(1, 30)	(None, 27, 9,000, 1)	811
Total params: 7,923; trainable params: 7,923				

Following Almaslukh et al. (2017) and Li et al. (2018), we trained our SAE in a greedy approach. We first trained a simple AutoEncoder with conv2d as the encoder and with a 2D decomposition layer (conv2d_transpose) as the decoder. After the training converges, we fixed the conv2d and conv2d_transpose layers and trained the next encoding/decoding layer (dense_1/dense_2) in the SAE. Finally, all layers were configured as trainable for fine-tuning. The condensed data representation generated from dense_1 was used as the input for subsequent SVM learning.

Appendix References

Almaslukh, B., Jalal, A., and Abdelmonim, A. 2017. “An Effective Deep Autoencoder Approach for Online Smartphone-Based Human Activity Recognition,” *International Journal of Computer Science and Network Security* (16:3), pp. 197-205.

Bao, L., and Intille, S. S. 2004. “Activity Recognition from User-Annotated Acceleration Data,” in *Proceedings of the International Conference on Pervasive Computing*, Vienna, Austria.

Blei, D. M., Ng, A. Y., and Jordan, M. I. 2003. “Latent Dirichlet Allocation,” *Journal of Machine Learning Research* (3), pp. 993-1022.

Chen, L., Hoey, J., Nugent, C. D., Cook, D. J., and Zhiwen Yu. 2012. “Sensor-Based Activity Recognition,” *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* (42:6), pp. 790-808.

Chen, Y., and Xue, Y. 2015. “A Deep Learning Approach to Human Activity Recognition Based on Single Accelerometer,” in *2015 IEEE International Conference on Systems, Man, and Cybernetics*, pp. 1488-1492.

Goodfellow, I., Bengio, Y., Courville, A., and Bengio, Y. 2016. *Deep Learning*, vol. 1, Cambridge: MIT Press.

Huynh, T., Fritz, M., and Schiele, B. 2008. *Discovery of Activity Patterns Using Topic Models*. (https://doi.org/10.1145/1409635.1409638).

Kuhn, H. W. 1955. “The Hungarian Method for the Assignment Problem,” *Naval Research Logistics Quarterly* (2:1-2), pp. 83-97.

Li, F., Shirahama, K., Nisar, M., Köping, L., and Grzegorzek, M. 2018. “Comparison of Feature Learning Methods for Human Activity Recognition Using Wearable Sensors,” *Sensors* (18:3), Article 679.

Millman, K. J., and Aivazis, M. 2011. “Python for Scientists and Engineers,” *Computing in Science & Engineering* (13:2), pp. 9-12.

Oliphant, T. E. 2006. *A Guide to NumPy* (https://web.mit.edu/dvp/Public/numpybook.pdf).

Ordóñez, F., and Roggen, D. 2016. “Deep Convolutional and LSTM Recurrent Neural Networks for Multimodal Wearable Activity Recognition,” *Sensors* (16:12), Article 115.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., and others. 2011. “Scikit-Learn: Machine Learning in Python,” *Journal of Machine Learning Research* (12), pp. 2825-2830.

Vallat, R. 2018. “EntroPy: Complexity of (EEG) Time-Series in Python,” Github (https://github.com/raphaelvallat/entropy).

Yang, J. B., Nguyen, M. N., San, P. P., Li, X. L., and Shonali, K. 2015. “Deep Convolutional Neural Networks on Multichannel Time Series for Human Activity Recognition,” in *Proceedings of the 24th International Conference on Artificial Intelligence (IJCAI)*, pp. 3995-4001.

Zeng, M., Nguyen, L. T., Yu, B., Mengshoel, O. J., Zhu, J., Wu, P., and Zhang, J. 2014. “Convolutional Neural Networks for Human Activity Recognition Using Mobile Sensors,” in *Proceedings of the 6th International Conference on Mobile Computing, Applications and Services*, ICST, pp. 197-205.

Zhu, H., Chen, H., and Brown, R. 2018. “A Sequence-to-Sequence Model-Based Deep Learning Approach for Recognizing Activity of Daily Living for Senior Care,” *Journal of Biomedical Informatics* (84), pp. 148-158.