Learning ADL Daily Routines with Spatiotemporal Neural Networks

Shan Gao, *Member, IEEE,* Ah-Hwee Tan, *Senior Member, IEEE,* and Rossi Setchi, *Senior Member, IEEE,*

***Abstract*—The activities of daily living (ADLs) refer to the activities performed by individuals on a daily basis and are the indicators of a person's habits, lifestyle, and wellbeing. Learning an individual's ADL daily routines has significant value in the healthcare domain. Specifically, ADL recognition and inter-ADL pattern learning problems have been studied extensively in the past couple of decades. However, discovering the patterns performed in a day and clustering them into ADL daily routines has been a relatively unexplored research area. In this paper, a self-organizing neural network model, called the Spatiotemporal ADL Adaptive Resonance Theory (STADLART), is proposed for learning ADL daily routines. STADLART integrates multimodal contextual information that involves the time and space wherein the ADL is performed. By encoding spatiotemporal information explicitly as input features, STADLART enables the learning of time-sensitive knowledge. Moreover, a STADLART variation named STADLART-NC is proposed to normalize and customize ADL weighting for daily routine learning. A weighting assignment scheme is developed that facilitates the assignment of weighting according to ADL importance in specific domains. Empirical experiments using both synthetic and real-world public data sets validate the performance of STADLART and STADLART-NC when compared with alternative pattern discovery methods. The results show STADLART could cluster ADL routines with better performance than baseline algorithms.**

***Index Terms*—ADL sequence, fusion ART, activity pattern, spatiotemporal features**

I. Introduction

The **activities of daily living (ADLs)**, as used by healthcare professionals, refer to the daily self-care activities performed by an individual in his or her place of residence, outdoors, or both. In the elderly healthcare domain, ADLs are usually used to measure the functional status of an elderly patient. Generally speaking, there are two subcategories of ADLs: basic ADLs (BADLs) [1], which refer to the daily activities used in maintaining basic wellbeing, and instrumental ADLs (IADLs) [2] [3], which help an individual live independently and respectably in a community. Shopping, social activity, and finance management are some examples of IADLs. Because ADLs are indicators of people's wellbeing, health issues faced by the elderly are largely reflected in ADLs. For example, a longer length of time taken for a particular ADL may indicate certain physical or cognitive disfunctions. The knowledge of ADLs and their patterns could help caregivers discover issues, predict future health conditions, and advise the elderly. ADLs that build up elderly tenants' daily lives, such as grooming, shower, breakfast, watching TV, housework, and exercise, are particularly of interest.

An **ADL sequence**, S, refers to an ordered set of ADLs. Formally, we define

$$\mathbf{S} = (\mathbf{A_1}, \mathbf{A_2}, ..., \mathbf{A_n}) \tag{1}$$

and

$$\mathbf{A_i} = < a_i, s_i, e_i, l_i > \text{ for i = 1, ... , n,} \tag{2}$$

where a_i denotes the activity ID, s_i denotes the activity starting time, e_i denotes the activity ending time, l_i denotes the location of the ADL, and n denotes the total number of identified ADLs. As shown in this definition, the set of ADLs, as well as their spatiotemporal information, are of great importance in ADL sequences. In real life, different applications have different sets of ADLs. The time when the ADLs occurred represents the order of the ADLs. The duration of an ADL also carries important information (e.g., wellness value). For example, exercise for 10 minutes is different than exercise for 2 hours.

An **ADL routine** refers to an ADL sequence that describes a person's ADLs in a day. A routine can be viewed as a template that captures certain regularities in the ADL order, occurrence time, and duration. A person may have different ADL routines for different days [4]. ADL daily routines are important in the healthcare domain because they largely reflect a person's wellness. In real life, people give ADL routine advice to others for wellness purposes, for example, going out for a walk after dinner or not going to sleep too late. Beyond the healthcare for elderly people, activity routine learning could also provide knowledge for related domains such as manufacturing and office scenarios. To define an ADL daily routine in a computational model, we view a routine as the template of an ADL sequence that tolerates a certain level of variation in the ADL order, starting time, and duration.

A mathematical model of an ADL daily routine on any given day poses a challenge. First, ADL routines contain sets of an indefinite number of ADLs, and the number of ADLs within a day largely depends on the context of the day, as well as other factors, such as the time of the year or weather. Second, defining the set of ADL-related features, for example ADL types, temporal features, and spatial features, to be used in daily routine clustering is another problem. In particular, ADL temporal information may contain the starting time, duration, and related day information, whereas the spatial information may contain the room information and locations within rooms. The selection and multimodal representation of features will affect clustering performance. Furthermore, user-related information should be taken into consideration. This information provides another set of features that gives a better representation of a person's individualized connection between the spatiotemporal features of ADLs and his or her daily routine, and this understanding could provide better detection of activity patterns and recommendation services.

Daily routine applications are usually domain-specified that only a set of ADLs are of interest. In public ADL data sets [5], ADLs are usually collected (or annotated) from sensory inputs, and some ADLs (e.g., meal preparation) may be identified frequently throughout a day. In extreme cases, a single ADL type could take more than half of the total number of ADLs. To reduce the weights of each individual occurrence of the same ADL types, it is necessary to normalize the weighting of each ADL. On the other hand, in different application domains, for example, in finding eating habits or measuring exercise patterns, various ADLs contribute differently and have different importance in the clustering of daily routines. Hence, an ADL weight assignment algorithm and a systematical weight assignment scheme are needed.

In the current paper, we propose a three-layer self-organizing neural network model called the Spatiotemporal ADL Adaptive Resonance Theory (STADLART). Through the learning process that occurs across the three layers, STADLART is capable of learning spatiotemporally distinct ADLs and ADL daily routines by using encoded time, space, and activity information across multimodal pattern channels. STADLART models ADL and ADL daily routines in two different layers, forming a deep neural network. The newly introduced spatiotemporal features enable STADLART to organize ADLs based on their spatiotemporal features. Also, STADLART applies an algorithm adding weight normalization according to the number of the same type of ADLs in a day. This algorithm is capable of weight customization for different application domains. To facilitate this algorithm, a weight assignment scheme and a set of newly designed cluster measurements are proposed to incorporate expert knowledge on daily routine clustering in different applications.

Empirical experiments on a synthetic data set and public data set [5] were conducted to validate the performance of STADLART. The public CASAS data set is published by Washington State University [5] and contains long-term sensory data captured from several testbeds using real human participants. The CASAS data set is popular in the ambient intelligence field for studies on ADL-related problems. The experimental results show that STADLART, together with the ADL weight assignment algorithm, could cluster ADL daily routines according to different application requirements.

The rest of this paper is organized as follows: Section II provides a literature review on the related work. Section III provides a brief introduction on the fusion ART models. Section IV introduces the proposed STADLART neural network, including its data fields. Section V introduces the learning mechanism of STADLART. Section VI introduces the ADL weights normalization and customization algorithms and a weights assignment scheme. Section VII and Section VIII show the experimental results and discuss the limitations of STADLART. Finally, Section IX concludes the paper.

II. Related Work

A. ADL recognition and pattern learning

There has been extensive research on human activity recognition [6] [7] [8] and behavior tracking [9]. These works focus on the recognition of users' activities and behavior through sensor data, which makes learning the routine of daily activities possible. For activity pattern discovery at the ADL level, researchers [10] have formulated human activity modeling as a spatiotemporal pattern-matching problem on top of the sequence of symbolic information produced by a sensor network. The proposed algorithm generates a transitional probability model between key ADLs to represent human activity patterns.

In the literature [11], Episode Discovery (ED) [12] was applied to an ADL stream to find regular patterns. This work also proposed a set of measures, for example, the accuracy of a rule and compression rate, to evaluate the episode finding performance. In another work [13], episode patterns are studied in ADL streams using the terms regularity (the time gaps between the occurrences are bounded), periodicity (some occurrences form repeating cycles of time intervals), and time intervals. Periodicity is mined using Gaussian Misture models (GMM), and frequency is mined using a frequent episode lattice (FEL).

In our previous work [4], a multimemory neural network architecture named ADLART was proposed, which incorporates EM-ART (episodic memory adaptive resonance theory) [14] [15] to learn ADL sequence patterns. In EM-ART, the events in an episode are decayed exponentially, so that the most recent events have more weight in the code competition than the earlier events. This setting is not suitable for ADL daily routine learning because all ADLs in the same day should have the same level of importance, regardless of when they occur. In ADLART, an ADL daily routine is modeled as an episode of ADLs, using the normalized starting time over the day as the node activation values. The episode of an ADL daily routine is stored in both episodic memory and sematic memory. The sematic memory represents the routines for various days. However, there are several limitations of the ADLART model. First, because of the time representation of ADLs, ADLART only captures the starting time of ADLs, which loses the important ADL duration information. Second, ADLART does not incorporate the spatial information of ADLs.

B. Temporal information representation in neural networks

Generally speaking, there are at least two approaches to modeling activity temporal information in a neural network. In the first approach, temporal information, for example, activity sequences, are modeled as time series. In recurrent neural networks (e.g., LSTM) and spiking neural networks [16] [17], temporal relationships are implicitly modeled as input iterations. On the other hand, time could be explicitly encoded as the activation of input nodes [4]. Moreover, if complement coding [18] is applied, the encoded time and the complement code could be used to learn a range of time in a continuous space. This approach provides the possibility for temporal features to be used in clustering activities.

C. Customizable feature weights

For k-means and fuzzy k-means based clustering algorithms, feature weighting algorithms [19] [20] [21] [22] are proposed to automatically calculate a set of feature weights to maximize the specific internal cluster evaluation measures, for example, accuracy, F-1 score, and normalized mutual information.

For clustering algorithms that are based on the adaptive resonance theory (ART), fuzzy ARTMAP with adaptively weighted distances (FAMawd) [23] substitutes the regular L_1-norm with a weighted L_1-norm to measure the distances between categories and input patterns. The distance-related weights are a function of a category's shape, allowing for bias in the direction of a category's expansion during learning. Another work, FAMRFW [24], extended FAMawd with another new distance measure.

These measures make assumptions of the shapes of clusters in the sample space, and the feature weights assignment algorithms try to find weights to fit into these cluster shapes. However, for learning ADL daily routines, it is usually expert knowledge in different application domains that provides cluster shape assumptions. As features, different ADLs are weighted differently in different applications. For example, in food-related studies, meal-related ADLs are important, whereas in physical-wellness-related applications, the body-movement-related ADLs are of more interest.

III. Fusion ART

Various models of ART and their supervised learning versions have been used for pattern analysis and recognition tasks. Within the family of ART models, there is a group of networks, known as fusion ART [14], formerly the multichannel adaptive resonance associative maps (multichannel ARAM) [25], that formulates cognitive codes that associate multimodal patterns across multiple input channels. Self-organizing is an important feature of fusion ART in the sense that when no learned node is matched, the network will autonomously use the uncommitted node to represent the new pattern.

Based on a generic multi-channel architecture (see Figure 1), the dynamics of fusion ART are summarized as follows:

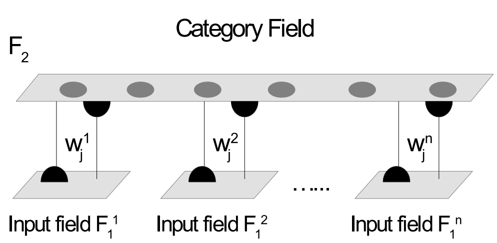

Fig. 1. The generic fusion ART architecture

Input fields: Let F_1^k denote the input field that holds the input patterns of channel *k*.

Input vectors: Let $\mathbf{I}^k = (I_1^k, \bar{I}_1^k, ..., I_n^k, \bar{I}_n^k)$ denote the input vector of channel *k* for *k = 1,...,n*, where $I_i^k \in [0, 1]$ are the input signal values and $\bar{I}_i^k = 1 - I_i^k$. Complement coding serves to normalize the magnitude of the input vectors and has been found effective in fuzzy ART systems when it comes to prevent the problem of code proliferation [18].

Category field: Let F_i, where $i > 1$, indicate the category field. In the standard multichannel ART, there is only one category field F_2.

Activity vectors: Let $\mathbf{x}^k$ denote the activity vector for input field F_1^k, and $\mathbf{y} = (y_1, y_2, ..., y_m)$ denote the activity vector of F_2. Initially, $\mathbf{x}^k = \mathbf{I}^k$ for $k = 1, 2, ..., n$.

Weight vectors: Let $\mathbf{w}_j^k$ denote the weight vector associated with the *j*th node in F_2 for learning the input patterns in F_1^k. Initially, F_2 contains only one uncommitted node with the weight vectors containing all 1s.

Parameters: Each field's dynamics are determined by the choice parameters $\alpha^k \geq 0$, learning rate parameters $\beta^k \in [0, 1]$, contribution parameters $\gamma^k \in [0, 1]$, and vigilance parameters $\rho^k \in [0, 1]$.

Code activation: Given the activity vectors $x^1, x^2, ..., x^k$, for each F_2 node *j*, the choice function T_j is as follows:

$$T_j = \sum_{k=1}^{n} \gamma^k \frac{|\mathbf{x}^k \wedge \mathbf{w}_j^k|}{\alpha^k + |\mathbf{w}_j^k|}, \tag{3}$$

where the fuzzy AND operator $\wedge$ is defined by $(\mathbf{p} \wedge \mathbf{q})_i \equiv min(p_i, q_i)$, and the norm $|.|$ is defined by $|\mathbf{p}| \equiv \sum_i p_i$ for vectors $\mathbf{p}$ and $\mathbf{q}$. Fundamentally, the choice function T_j computes the similarity of the activity verctors with their respective weight vectors of the F_2 node j with respect to the norm of individual weight vectors.

Code competition: The F_2 node with the highest choice function value is identified by the code competition process. The winner is indexed at *J*, where $T_J = max\{T_j :$ for all F_2 node j$\}$. When a category choice is made at node *J*, $y_J = 1$ and $y_j = 0$, $\forall j \neq J$. This indicates a winner-take-all strategy.

Template matching: Upon code competition, the template-matching process takes place to check if resonance is occurring. For each channel *k*, the match function is given as follows:

$$m_J^k = \frac{|\mathbf{x}^k \wedge \mathbf{w}_J^k|}{|\mathbf{x}^k|} \geq \rho^k. \tag{4}$$

The match function computes the similarity of the activity and weight vectors with respect to the norm of the activity vectors. The match function works together with the choice function to achieve stable coding and maximize code compression. The template-matching value of the chosen node *J* is checked to see whether it meets the vigilance criterion. If any of the vigilance constraints are violated, a mismatch reset occurs by setting the choice function T_J to 0 for the duration of the input presentation. The search process will keep selecting other F_2 nodes until resonance occurs. If the uncommitted node in F_2 is identified as the winner, after learning it becomes committed, a new uncommitted node is created and added to F_2.

Template learning: Once a node *J* is selected for learning, in each channel *k*, the weight vector is updated by the learning rule shown in the following:

$$\mathbf{w}_J^{k(new)} = (1-\beta^k)\mathbf{w}_J^{k(old)} + \beta^k(\mathbf{x}^k \wedge \mathbf{w}_J^{k(old)}). \quad (5)$$

The learning rule adjusts the weight values towards the fuzzy AND of their original values and the respective weight values. This is designed to learn by encoding the common attribute values of the input vectors and the weight vectors. For an uncommitted node, the learning rate β^k are typically set to 1. For committed nodes, β^k can be set to 1 for fast learning or below 1 for slow learning in a noisy environment.

IV. STADLART Architecture

To learn daily ADL routines, we propose a self-organizing neural network model called the Spatiotemporal ADL Adaptive Resonance Theory (STADLART), which is a three-layer fusion ART network, as shown in Figure 2. The first layer consists of the input fields that represent the information of the ADL type, time, day, and space. The second layer contains the spatiotemporal ADL field, wherein the category nodes encode the associations of ADL types and spatiotemporal information. The third layer contains the ADL routine field, wherein the ADL routine nodes encode the sequential combinations of the spatiotemporal ADLs. Across the three layers, the ADL patterns (F_1 layer) are generalized into spatiotemporally distinguished ADLs (F_2 layer) and then into ADL daily routines (F_3 layer). A detailed description of the three layers is discussed in the following subsections.

A. Encoding ADLs in STADLART

In STADLART (see Figure 2), the input fields in F_1 encode the ADL type, time, day, and spatial information.

1) ADL field: In different problem domains, the sets of ADLs used are different [1][2][3]. In STADLART, the set of ADLs are selected by considering the significance they have on wellbeing and availability in the public data sets. The list of ADLs used in STADLART is summarized in Table I.

The ADL field represents the type of the input ADL event. Let

$$\mathbf{x^a} = (x_1^a, \bar{x_1^a}, x_2^a, \bar{x_2^a}, ..., x_8^a, \bar{x_8^a}) \quad (6)$$

denote the activity vector, where x_i^a indicates the ADL type, while $\bar{x_i^a}$ is its complement. Although in the current settings the observed activities are assumed to be totally certain (x_i^a to be 1 or 0), the model is capable of handling the fuzzy inputs [0, 1] interval.

TABLE I
ADL types used by STADLART

Index	ADL	Index	ADL
1	Meal Preparation	5	Washing Dishes
2	Eating	6	Toilet
3	Working	7	Outside
4	Sleeping	8	Housekeeping

Our work adopts the ADL set used and annotated in the CASAS data set. In problem domains where the input samples contain unrecognized activities "other activity", our model will simply treat the "other activity" as a type of ADL and learn routines with "other activity" as part of the routine. However, as the "other activity" category is broad and unknown, it may not be very useful. Moreover, if later a new ADL type is identified, it is possible create new nodes in the x^a and create their links to the next layer. This process is similar to the template matching and node creation described in Section III and this will not affect the previous learnt associations of other nodes.

2) Time field: In STADLART, the time field F_1^t represents the starting time and duration of the activities performed by the user. In particular, F_1^t contains a vector with a normalized (over a day) starting time and the complement of the normalized ending time.

Let

$$\mathbf{x^t} = (x_1^t, \bar{x_2^t}) \quad (7)$$

denote the activity vector, where x_1^t represents the normalized starting time over a day, while $\bar{x_2^t}$ represents the complement of the normalized ending time. Based on fuzzy ART, this scheme that consists of a pair of complement-coded activity values (start time and complement of end time) is sufficient to encode a time interval.

3) Day field: The day field F_1^d contains the day type information, including day of week and special days. Let $\mathbf{x^d}$ denote the activity vector of F_1^d. We have

$$\mathbf{x^d} = (x_1^d, \bar{x_1^d}, x_2^d, \bar{x_2^d}, ..., x_{12}^d, \bar{x_{12}^d}), \quad (8)$$

where x_n^d indicates the activation value of the nth day type, while $\bar{x_n^d}$ is its complement. In STADLART, we identify a total of 12 day types, as listed in Table II.

TABLE II
Day information input

Index	Day Type	Index	Day Type
1	Monday	7	Sunday
2	Tuesday	8	Weekday
3	Wednesday	9	Weekend
4	Thursday	10	Public Holiday
5	Friday	11	Sick Day
6	Saturday	12	Vacation

More than one element could be activated at the same time in the day information vector $\mathbf{x^d}$. For example, if a public holiday falls on Monday, the values of x_1^d, x_8^d, and x_{10}^d will be 1s while others will be 0s. Complement coding is then applied accordingly.

4) Spatial field: The ADL field F_1^s represents the spatial information of the input ADL event. Let $\mathbf{x^s}$ denote the activity vector in this field, and we have

$$\mathbf{x^s} = (x_1^s, \bar{x_1^s}, x_2^s, \bar{x_2^s}, ..., x_6^s, \bar{x_6^s}), \quad (9)$$

where x_n^s indicates the spatial information, for example, room types or being outdoors, of the target ADL, while $\bar{x_n^s}$ is its complement. Referring to the literature on activity recognition, six room types (including outside are considered in STADLART, as listed in Table III. One example for $\mathbf{x^s}$ could be having a shower in the washroom, which is represented as an input

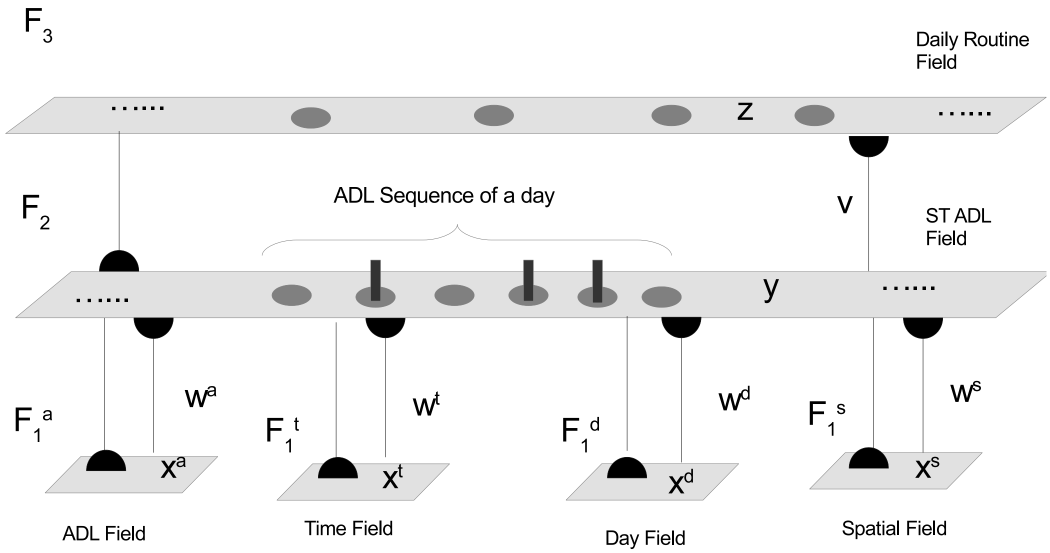

Fig. 2. STADLART model

vector of $\mathbf{x^s}$ = (0,1,0,1,0,1,1,0,0,1,0,1) (with the odd elements representing the complement of the even elements).

TABLE III
SPATIAL TYPES

Index	Room
1	Living Room
2	Bedroom
3	Kitchen
4	Washroom
5	Reading Room
6	Outside

B. Encoding Daily ADL Routines

An ADL daily routine refers to an ADL sequence that describes a person's ADLs in a day. The self-organizing spatiotemporal ADL layer F_2 contains one field that stores the spatiotemporal ADL categories associated with all the input information. Every basic ADL type may have multiple spatiotemporal ADL categories inside the F_2 layer. For example, exercising in the morning in the living room is a different spatiotemporal ADL from exercising in the afternoon in the bedroom. The activity vector $\mathbf{y}$ is learned from the F_1 layer, and a winning node will be identified in $\mathbf{y}$. Let another vector $\mathbf{y}'$ denote the ADLs performed in a day, and we have

$$\mathbf{y}' = (y_1, \bar{y_1}, y_2, \bar{y_2}, ..., y_m, \bar{y_m}), \tag{10}$$

where m is the number of spatiotemporal ADL categories in the F_2 layer, and y_m indicates whether the mth spatiotemporal ADL category is performed in the day, while $\bar{y_m}$ is its complements. $\mathbf{y}'$ is used to learn the next F_3 layer. The activities in $\mathbf{y}'$ form a day's ADL sequence. In other words, $\mathbf{y}'$ has the size of all learned spatiotemporal ADLs, and the spatiotemporal ADLs performed in the current day have their activation value set to 1, while those not performed have their activation values set to 0.

The F_3 layer is the ADL daily routine layer. The nodes inside the F_3 layer are learned from the ADL vector $\mathbf{y}'$ of the F_2 layer, and each node represents a unique ADL daily routine of the user. The activity vector of F_3 is denoted by $\mathbf{z}$. The detailed spatiotemporal information of each component of the ADLs could be retrieved by tracing them down through the STADLART architecture.

V. LEARNING AND RETRIEVAL OF DAILY ROUTINES

A. Model Training

The STADLART neural network consists of three layers. The F_2 layer focuses on individual ADLs, while the F_3 layer combines the ADLs from the F_2 layer to form the daily ADL sequences. In the training phase, STADLART needs to go through two steps to start learning. First, STADLART learns through the F_1 layer to the F_2 layer for each ADL input. STADLART learns other individual ADLs in the same way until all ADLs in the day have been learned. Second, STADLART will learn the F_3 layer from the ADLs sequence. The training algorithm is summarized in Algorithm 1.

B. Daily Routine Readout

After training the STADLART model, each node in the F_3 layer encodes the learned daily ADL routines. By reading the activation values of spatiotemporal ADLs associated with the daily routine categories, a list of routines can be generated from the F_2 layer. At the F_1 layer, the learned starting time, duration, and spatial information associated with the spatiotemporal ADL categories can be retrieved.

Algorithm 1 STADLART training process

Require: A sequence of ADLs, each in the form of $(\mathbf{x}^a, \mathbf{x}^t, \mathbf{x}^d, \mathbf{x}^s)$
Ensure: Learn an ADL routine
for each input ADL $(\mathbf{x}^a, \mathbf{x}^t, \mathbf{x}^d, \mathbf{x}^s)$ **do**
    Compute activity vector $\mathbf{y}$ in the F_2 layer using input patterns $(\mathbf{x}^a, \mathbf{x}^t, \mathbf{x}^d, \mathbf{x}^s)$
    The winning F_2 node learns a spatiotemporal ADL category
    Update the performed spatiotemporal ADL list $\mathbf{y}'$
end for
Update activity vector $\mathbf{z}$ in the F_3 layer using input patterns $\mathbf{y}'$
The winning F_3 node learns an ADL routine category from $\mathbf{z}$

C. Spatiotemporal Information Retrieval

The spatiotemporal characteristic of STADLART allows the retrieval of a particular ADL by using the user's spatial and temporal preference. This is STADLART's capability for learning the ADL spatiotemporal patterns of the user. Specifically, for a particular ADL type input $\mathbf{x}^a$, STADLART will activate every category *j* in the F_2 layer using the choice function

$$T_j^2 = \sum_{k=1}^{n} \gamma_2^k \frac{|\mathbf{x}^k \wedge \mathbf{w}_j^k|}{\alpha_2^k + |\mathbf{w}_j^k|} \tag{11}$$

revised from Formula (3).

All the spatiotemporal ADL nodes in the F_2 layer that match the ADL type $\mathbf{x}^a$ will be selected. The spatial and temporal information associated with the selected categories will be retrieved and output. The algorithm is summarized in Algorithm 2.

Algorithm 2 Retrieve spatiotemporal information using ADL

Require: ADL vector $\mathbf{x}^a$ from F_1^a
Ensure: x^s, x^t, and x^d associated with ADL type $\mathbf{x}^a$
read in the ADL vector $\mathbf{x}^a$ from F_1^a
Activate every category *j* in F_2 by choice function $T_j^{F_2} = \sum_{k=1}^{n} \gamma_2^k \frac{|\mathbf{x}^k \wedge \mathbf{w}_j^k|}{\alpha_2^k + |\mathbf{w}_j^k|}$
while selecting a new category *J* and $T_J^{F_2} > 0$ **do**
    Readout the x^t, x^d, and x^s associated with *J*
end while

VI. ADL Weight Normalization and Customization

A. ADL weight normalization

With real-life ADL data, for example, [5], ADLs are usually collected (or annotated) based on sensory inputs. As such, certain ADLs (e.g., meal preparation) may occur frequently in a day. For example, a stove may be turned on and off seven times a day, and thus, seven meal preparation ADLs would be logged. To normalize the contribution of ADLs in a day, a weighting scheme is introduced and applied to the STADLART choice function (Formula (11)):

$$T_j^2 = \sum_{k=1}^{n} \gamma_2^k \frac{|\mathbf{n}(\mathbf{x}^k \wedge \mathbf{w}_j^k)|}{\alpha_2^k + |\mathbf{n}\mathbf{w}_j^k|}. \tag{12}$$

where n is the feature normalization vector for $n_i = 1/a_i$, where a_i is the count of spatiotemporal ADL type i in the day. On different days, n is counted separately. The new algorithm is named STADLART-N. With the introduction of n, the frequently occurred ADLs will have less weight for each of their occurrences while the less frequent ADLs will have a higher weight for each occurrence. For example, if there are seven meal preparation ADLs identified within one day, after applying the normalization vector, a, each meal preparation ADL contributes one seventh compared with before.

B. Customizable ADL weights

Besides feature normalization for the different frequencies of ADLs, in different application domains, different ADLs play different roles, bringing in different level of importance. To emphasize the ADLs of interest in specific applications, similar to the feature weight normalization vector, a feature weight customization vector, c, is introduced to the choice function (Formula (11)) in the ADL daily routine learning algorithm

$$T_j^2 = \sum_{k=1}^{n} \gamma_2^k \frac{|\mathbf{n}\mathbf{c}(\mathbf{x}^k \wedge \mathbf{w}_j^k)|}{\alpha_2^k + |\mathbf{n}\mathbf{c}\mathbf{w}_j^k|}. \tag{13}$$

where c_i is the weight vector that assigns weights to ADL i according to its importance. The new algorithm is named STADLART-NC. To formalize c systematically, a five-level scale is used to measure the importance of each ADL and assign corresponding weight values. As shown in Table IV, very important ADLs will have their weight equal to 1 while other ADLs will have their weight below 1 according to their domain knowledge.

TABLE IV
Assignment of ADL weight according to its importance

ADL importance	**weight**
Not Important	0.2
Less Important	0.4
Normal	0.6
Important	0.8
Very Important	1.0

VII. Experiments on Synthetic Data

A. Synthetic Data Generation

The performance of STADLART is first evaluated using a synthetic data set that is generated using a set of predefined ADL routines. The experiments on synthetic data are served to verify the veracity of the output that the samples are generated from few templates and the model could generates clusters that corresponding to the templates which is hidden from the model. To gain more confidence on veracity, real-world scenarios will be gradually added to the synthetic data making it more “real”. In the first experiment, five synthetic ADL routines were created, with each one describing a specific type of day, namely the normal day, the wake up early day, the hungover day, the outing day, and the housework day. About 2,000 samples (days of ADLs) are generated. When generating a daily ADL sequence sample, the day is first randomly assigned a routine template, and then, all the ADL types are generated according to the assigned routine template, with every ADL having a 15-minute random variation in the starting time and duration.

A normal day routine, which is an example of a normal day, as shown in Figure 3, starts at 7:00. The person performs personal hygiene (Toilet) and has breakfast (Eating) before 8:00. Lunch preparation (MealP) is around 10:00, followed by having the lunch (Eating) before noon. The person prepares dinner (MealP) at 16:00, and dinner (Eating) starts at 18:00. The dishes are washed (WashDish) shortly after finishing dinner and finally the person goes to sleep (Sleep) at about 22:00. In the “wake up early” day routine, the person wakes up at about 6:00, and every ADL is performed about one hour earlier. While in the “hungover” day routine, the person wakes up late, near noon, with no breakfast, and all other ADLs are postponed by around one hour. In the “outing day” routine and the “housework” day routine, there are outside ADL types and housework ADL types that distinguish these two types of days from other days.

B. Experiment and Evaluation

STADLART is trained throughout the experiments with setting the slow learning ART parameter to β = 0.2. Consequentially, a spatiotemporal ADL in a routine that is not observed in the subsequence inputs of this routine will have its association strength decreased gradually.

In fusion ART networks, the vigilance parameters control the level of generalization. In this set, STADLART was run with the vigilance parameters of both F_1 and F_2 fields set to 0.8, 0.85, 0.9, and 0.95. The number of categories learned in the F_3 layer are listed in Table V. With the vigilance values of 0.85 and 0.9, STADLART learned exactly five daily routine categories. Comparing the readouts of the F_3 layer with the synthetic data, it is clear that the categories in the F_3 layer represent the routine templates in the synthetic data set.

TABLE V
THE NUMBER OF DAILY ROUTINE CATEGORIES GENERATED USING DIFFERENT VIGILANCE VALUES

Vigilance	Categories
0.80	4
0.85	5
0.90	5
0.95	17
1.00	220

Using the experiment with ρ_2 = 0.85 as an example, Figure 4 shows a learned daily routine category that corresponds to the normal day routine template.

The extended periods of ADLs, for example, eating, in the learned categories occur because of two main reasons. The first reason is that ADL generation has a 15-minute random variation and thus has longer ADL periods. The second reason is that the generation of ADLs spans multiple days.

Because the templates for generating samples provide the label information, the STADLART output clusters are evaluated with external evaluation indices, specifically accuracy, F-1 score, normalized mutual information (NMI), and the Jaccard index. STADLART is compared with our previous work ADLART, baseline algorithms (namely K-means), and the LSTM network (the implementation in deeplearning4j project [26] is used). Because of the limitation of ADLART, ADLs at different times of the day (e.g., morning, afternoon, and evening) are identified as different features, for example, meal preparation in the morning is considered a different feature from meal preparation in the afternoon. However, ADLs of the same type within the same part of day are only counted once. For K-means, each input sample is an ADL routine containing a fixed number of ADLs (if the ADLs are not performed in the day the fields will left 0s). Each cluster center represents one routine learned from several samples. The difference of element ADLs between the new set of ADLs and the cluster center is calculated as the distance. It is weighted in temporal and spatial differences.

TABLE VI
THE EXTERNAL INDEX SCORES FOR THE SYNTHETIC DATA SET

Algorithm	Accuracy	F-1	NMI	Jaccard Index
STADLART	0.99	0.94	0.92	0.91
ADLART	0,95	0.88	0.88	0.85
K-means	0.94	0.88	0.87	0.84
LSTM	0.99	0.93	0.92	0.90

As shown in Table VI, STADLART outperformed the ADLART model and baseline clustering algorithms. The advantage of STADLART is largely because of the spatiotemporal features that could differentiate the same type of ADLs based on their temporal differences. At the same time, LSTM shows a similar level of performance with STADLART on the synthetic data set.

VIII. Experiment on Real-life Data Set

A. CASAS Data Set and Preprocessong

The CASAS data set consists of a total of 38 data sets that contain sensory inputs collected from well-equipped testbeds in several cities across the world. The 17th data set was chosen for the following experiment. The data set, collected from the testbed in Aruba, contains the sensory readings of an elderly female tenant over a period of 220 days. The data collected are primarily sensory readings with annotated ADLs as ground truths, which are taken as the inputs for STADLART. For the real-word data set, the result will be checked against the original data set manually and verify whether the generated routines are representative.

Data preprocessing was performed on the raw CASAS data set. First, the ADL type Enter_Home is combined with the ADL type Leave_Home. The duration of Leave_Home is hence the time span from Leave_Home to Enter_Home. Second, the ADL type “Resperate” has only six instances.

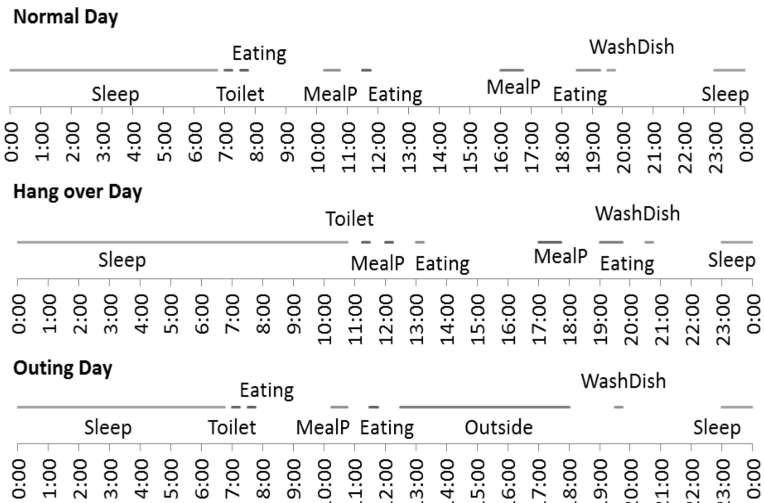

Fig. 3. Synthetic routine templates

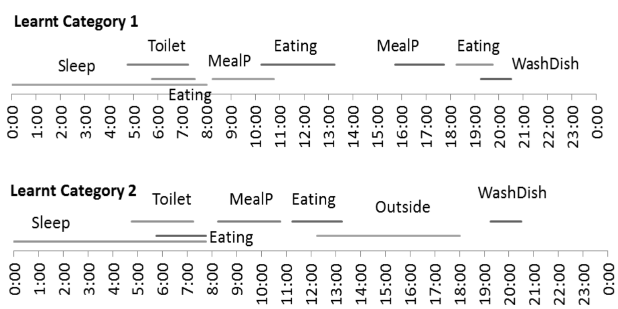

Fig. 4. A sample of a learned ADL routine

TABLE VII
ANNOTATED ADL TYPES IN THE PREPROCESSED CASAS DATA SET

ADL	Count
Meal_Preparation	1604
Eating	257
Working	171
Sleeping	400
Washing_Dishes	65
Bed_to_Toilet	157
Leave_Home	318
Housekeeping	33

Because it is not significant, this ADL type is removed. Third, there are 2,910 instances of the ADL type “relax” over the 220 days, with an occurrence count of more than 10 times per day spanning very short intervals each. This ADL type has no significance in differentiating the daily routine and is thus also removed. In the CASAS data set, dates were given instead of days of the week. Therefore, a conversion with reference to a calendar was performed. Because ADL are recognized from fixed sensors (stove, water tap, door, sofa, etc.) in the six rooms, the ADLs are associated with fixed spatial information. The final set of used ADL types is shown in Table VII, giving a total of 3,005 ADL samples.

With this preprocessed data set, in later subsections, experiments are conducted to test the generalization behavior of STADLART in terms of categories learned in different layers with different parameter configurations. After doing this, STADLART is compared with baseline algorithms, ADL-ART, and LSTM. Finally, the feature normalization and customization versions, STADLART-N and STADLART-C, are evaluated.

B. Experiments on spatiotemporal ADL category generalization

In this experiment, we look at the spatiotemporal ADL category layer. The ADL types associated with temporal and spatial information are recognized and stored in this layer. We evaluate the generalization behavior and find the key ADL categories to represent a user’s ADL preference. We conduct this experiment in a fast learning setting, for example, with a learning rate of 1.0. The contribution parameters are set to 0.4 for ADL type, 0.4 for temporal, and 0.2 for spatial. As discussed in the previous subsection, the vigilance parameter values for all the fields are set to 1.0, which means all nodes in the spatiotemporal ADL category represent an ADL type with a fixed temporal and spatial category. As a result, we generated 375 spatiotemporal ADL categories from the 3,005 input samples. In other words, each spatiotemporal ADL category represents about five inputs.

Among the 375 categories, the top 10 categories represent 33, 32, 23, 22, 22, 21, 20, 20, 19, and 19 inputs. At the same time, there are 100 categories that represent only one input and 62 categories that represent two inputs. The top categories mostly represent Sleep, Meal_Preparing ADL, and Leave_Home. This is because these three ADLs appear in the most number of input instances.

C. Experiment on daily routine categories generation

In this experiment, the learning process from the Spatiotemporal ADL category layer, F_2, to the ADL daily routine category layer, F_3, is evaluated. F_2 contains the learned spatial temporal ADL categories that this person performed in his daily life. The last step is to learn the ADL daily routines that could be used to describe his personal behavior or lifestyle.

In the first experiment, fast learning is used, that is, learning rate $\beta = 1.0$. The relationship between the choice of the vigilance parameter value of spatiotemporal ADL field and the number of daily routine categories generated are shown in Figure 5.

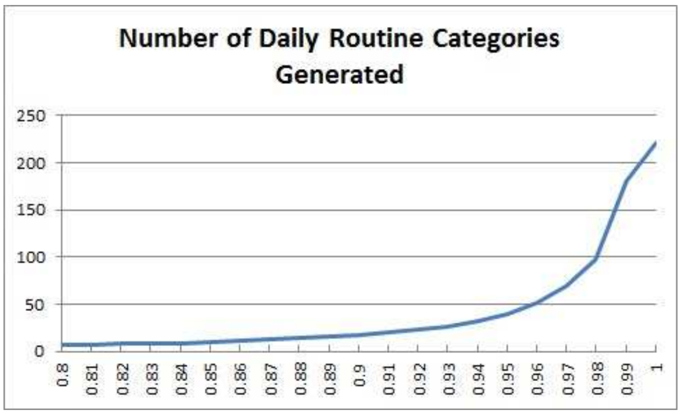

Fig. 5. The number of daily routine categories generated with different vigilance parameter values

Vigilance parameters ρ_2 and ρ_3 are chosen to be 0.9 and 0.2 for F_2 and F_3, respectively. Because more generalization is preferred, ρ_3 for learning daily routines is set to a low value. Based on 220 samples, the STADLART model learned a total of 19 daily routines from the CASAS data set. To verify the veracity of the result, the top five routines are manually examined by randomly picking up five days of that routine and checking the raw data. The top three routines represent 25, 20, and 17 days, respectively. The common ADL types in the top two routines are shown in Figure 6. In the first routine, it can be seen that the elderly woman visits the toilet regularly in the morning between 4:00 and 6:00. However, in the second routine, on certain days, the elderly woman prepares meals throughout the day, and she may be preparing a big dinner with her visiting daughter.The result of the manual verification is consistent with the learnt routines. From the results, we could see that in reality, the activity routines are more diverse than the synthetic data, and there is no single dominant daily routine over the 220 days. However, certain interesting ADL patterns can be observed.

D. Comparing STADLART with other algorithms

Because the routines are not labeled in the CASAS data set, the results from STADLART are compared with the ADLART and baseline algorithms using an intercluster measure. Specifically, the overall average silhouette width [27] is used, which indicates the quality of the underlying structure of the clusters: a higher value indicates a stronger structure. For

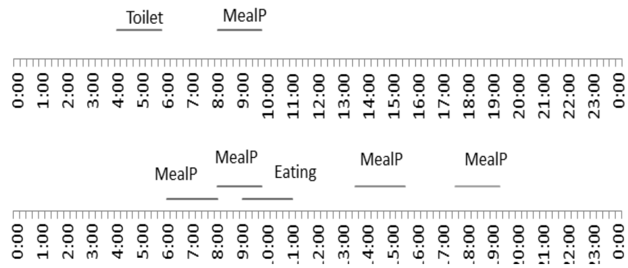

Fig. 6. Two ADL routines learned by STADLART

ADLART and other clustering algorithms, the CASAS data set is preprocessed much like how the synthetic data set that ADLs in different parts of day are treated as different features. The evaluation results from clustering the spatiotemporal ADLs and ADL routines are shown in Table VIII. The results show that the learned clusters from the CASAS data set are less structural than the clusters learned from the synthetic data, and STADLART outperforms ADLART and the baseline clustering algorithms, e.g., K-means. Because of the small sample size (220 in the CASAS data set), other deep neural networks such as LSTM do not significantly out perform the baseline algorithms.

TABLE VIII
THE OVERALL AVERAGE SILHOUETTE WIDTH FOR THE DATA SETS

Algorithm	Synthetic-ASW	CASAS-ASW
STADLART	0.93	0.32
ADLART	0.90	0.31
K-means	0.90	0.30
LSTM	0.93	0.30

E. Experiment on feature weight normalization and customization

In this subsection, the original STADLART is first compared with its feature weight normalization variation, STADLART-N. After doing this, two application scenarios, including a meal analysis and outing behavior analysis, are assumed, and the feature weight customization variation, STADLART-C, and normalization and customization variation, STADLART-NC, are compared with the original STADLART.

In the first experiment, the feature weight normalization variation, STADLART-N, is applied to the CASAS data set. Similar to previous experiments, fast learning is used here with a learning rate of $\beta = 1.0$. Vigilance parameters ρ_2 and ρ_3 are set to be 0.9 and 0.2 for F_2 and F_3, respectively.

TABLE IX
COMPARING STADLART AND STADLART-N USING THE CASAS DATA SET

Algorithm	Routines	Membership of Top Three Routines
STADLART	19	25, 20, 17
STADLART-N	18	31, 28, 25

As shown in Table IX, with feature weight normalization, STADLART-N generalizes a similar number of routines to STADLART on the CASAS data set. However, the top routines now represent more entries in the data set. By reading out the associations at F_2 and F_1 layers, the top two routines are plotted in Fig. 7. Compared with the top routines generated by ADLART (Fig. 6), which contain mostly meal_preparation ADL, STADLART-N generated more meaningful routines from the data set.

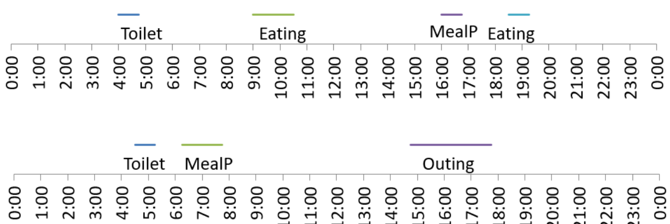

Fig. 7. Top two ADL routines learned by STADLART-N

In the second experiment, two application scenarios are the meal analysis and outing behavior analysis, with an emphasis on eating and going on an outing ADL, respectively. According to the ADL weight assignment guideline in Table IV and expert knowledge in the meal analysis experiment, the weight of eating is set to "very important" (i.e., 1.0), the weight of meal preparation is set to "normal" (i.e, 0.6), and all other ADLs are set to be "not important" (i.e., 0.2). Similarly, in the outing behavior analysis experiment, the outing ADL is set to "very important" (i.e., 1.0), while all other ADLs are set to be "not important" (i.e., 0.2). The experiment results are summarized in Table X. From the results, it is seen that with feature customization, STADLART-C and STADLART-NC could cluster daily ADL sequences into routines while focusing on key ADLs.

TABLE X
COMPARING STADLART, STADLART-C, AND STADLART-NC ON THE CASAS DATA SET

Experiment	Algorithm	Routines
Meal	STADLART	19
Meal	STADLART-C	10
Meal	STADLART-NC	8
Outing	STADLART	19
Outing	STADLART-C	7
Outing	STADLART-NC	6

IX. CONCLUSION

In this paper, a spatiotemporal fusion ART neural network model named STADLART has been presented to learn human daily activity routines. In contrast to the early work of ADLART, STADLART takes into consideration the ADL start time, duration, and spatial information. Experiments conducted based on a synthetic data set have shown that STADLART could learn ADL routines consistently with the ADL templates that are used to generate synthetic data. For the CASAS data set, STADLART manages to generalize some interesting routines of a person's life across days and provides information for further investigation on the person's behavior. In both experiments, STADLART was compared with ADLART and baseline clustering algorithms such as nearest neighbour and K-means. The results show that STADLART outperformed the baseline clustering algorithms in various aspects. Moreover, a STADLART variation named STADLART-NC was presented to normalize and customize ADL weights for different ADLs in daily routines. The ADL weight normalization successfully reduces the influence of ADL frequency while the ADL weight customization promotes ADLs of interest from expert knowledge. A guideline for ADL weight assignment on ADL weighting customization is also provided for ease of use. Experiments on the CASAS data set further demonstrate that STADLART-NC could learn more meaningful daily routines with normalized and customized ADL weighting for different application configurations.

By utilizing the STADLART models, intelligent systems will have the knowledge of typical life routines of the user. Based on this knowledge, various applications are made possible. First of all, the intelligent system can detect abnormalities of the user and notify caregivers or his relatives through messages. The intelligent system can also make predictions of the users following activities and give recommendations of activities, or provide other activity advices. The user activity routines also provide samples for long-term analysis, for example, activity routine change pattern over the years for certain age group.

Going forward, STADLART has some limitations to be resolved. The normalization formula makes use of the total number of the same ADLs performed in the day, making STADLART not suitable for online learning as and when partial ADLs are collected. An online adaptive normalization method would be desirable to enhance the feature weighting algorithm. Another limitation of STADLART is the lack of automatic routine explanation capability. Currently, the learned routines and activity patterns are explained manually. An automatic algorithm for translating learned patterns to meaningful symbolic representation will be an important direction for our future research.

REFERENCES

[1] S. Roley, J. DeLany, C. Barrows, S. Brownrigg, D. Honaker, D. Sava, V. Talley, K. Voelkerding, D. Amini, E. Smith *et al.*, "Occupational therapy practice framework: Domain & practice," *The American journal of occupational therapy.: official publication of the American Occupational Therapy Association*, vol. 62, no. 6, p. 625, 2008.

[2] M. Lawton and E. Brody, "Assessment of older people: Self-maintaining and instrumental activities of daily living." *The Gerontologist*, vol. 9, no. 3, pp. 179–186, 1969.

[3] R. Bucks, D. Ashworth, G. Wilcock, and K. Siegfried, "Assessment of activities of daily living in dementia: Development of the bristol activities of daily living scale," *Age and Ageing*, vol. 25, no. 2, pp. 113–120, 1996.

[4] S. Gao and A.-H. Tan, "User daily activity pattern learning: A multi-memory modeling approach," in *2014 International Joint Conference on Neural Networks*, 2014, pp. 1542–1548.

[5] D. J. Cook, "Learning setting-generalized activity models for smart spaces," *IEEE intelligent systems*, vol. 2010, no. 99, p. 1, 2010.

[6] J. Aggarwal and M. S. Ryoo, "Human activity analysis: A review," *ACM Computing Surveys (CSUR)*, vol. 43, no. 3, p. 16, 2011.

[7] A. A. Chaaraoui, P. Climent-Pérez, and F. Flórez-Revuelta, "A review on vision techniques applied to human behaviour analysis for ambient-assisted living," *Expert Systems with Applications*, vol. 39, no. 12, pp. 10 873–10 888, 2012.

[8] M. Karg and A. Kirsch, "Low cost activity recognition using depth cameras and context dependent spatial regions," in *Proceedings of the 2014 International Conference on Autonomous Agents and Multi-Agent Systems*, 2014, pp. 1359–1360.

[9] A. Fansi Tchango, V. Thomas, O. Buffet, F. Flacher, and A. Dutech, "Simulation-based behavior tracking of pedestrians in partially observed indoor environments," in *Proceedings of the 2014 International Conference on Autonomous Agents and Multi-Agent Systems*, 2014, pp. 1589–1590.

[10] D. Lymberopoulos, A. Bamis, and A. Savvides, "Extracting spatiotemporal human activity patterns in assisted living using a home sensor network," *Universal Access in the Information Society*, vol. 10, no. 2, pp. 125–138, 2011.

[11] J. Soulas, P. Lenca, and A. Thépaut, "Monitoring the habits of elderly people through data mining from home automation devices data," in *Portuguese Conference on Artificial Intelligence*. Springer, 2013, pp. 343–354.

[12] E. Heierman, M. Youngblood, and D. J. Cook, "Mining temporal sequences to discover interesting patterns," in *KDD Workshop on mining temporal and sequential data*. Citeseer, 2004, pp. 104–111.

[13] J. Soulas and P. Lenca, "Periodic episode discovery over event streams," in *Portuguese Conference on Artificial Intelligence*. Springer, 2015, pp. 547–559.

[14] A.-H. Tan, G. A. Carpenter, and S. Grossberg, "Intelligence through interaction: Towards a unified theory for learning," in *Advances in Neural Networks–ISNN 2007*. Springer, 2007, pp. 1094–1103.

[15] W. Wang, B. Subagdja, A.-H. Tan, and J. A. Starzyk, "Neural modeling of episodic memory: Encoding, retrieval, and forgetting," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 23, 2012.

[16] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[17] J. Hu, H. Tang, K. C. Tan, and H. Li, "How the brain formulates memory: A spatio-temporal model research frontier," *IEEE Computational Intelligence Magazine*, vol. 11, no. 2, pp. 56–68, 2016.

[18] G. A. Carpenter, S. Grossberg, and D. B. Rosen, "Fuzzy art: An adaptive resonance algorithm for rapid, stable classification of analog patterns," in *Neural Networks, 1991., IJCNN-91-Seattle International Joint Conference on*, vol. 2. IEEE, 1991, pp. 411–416.

[19] D. S. Modha and W. S. Spangler, "Feature weighting in k-means clustering," *Machine learning*, vol. 52, no. 3, pp. 217–237, 2003.

[20] J. Z. Huang, M. K. Ng, H. Rong, and Z. Li, "Automated variable weighting in k-means type clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 5, pp. 657–668, 2005.

[21] C.-C. Hung, S. Kulkarni, and B.-C. Kuo, "A new weighted fuzzy c-means clustering algorithm for remotely sensed image classification," *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, no. 3, pp. 543–553, 2011.

[22] J. Li, X. Gao, and L. Jiao, "A new feature weighted fuzzy clustering algorithm," in *International Workshop on Rough Sets, Fuzzy Sets, Data Mining, and Granular-Soft Computing*. Springer, 2005, pp. 412–420.

[23] D. Charalampidis, G. C. Anagnostopoulos, M. Georgiopoulos, and T. Kasparis, "Fuzzy art and fuzzy artmap with adaptively weighted distances," in *AeroSense 2002*. International Society for Optics and Photonics, 2002, pp. 86–97.

[24] R. Andonie, L. M. Sasu, and A. Caţaron, "A novel fuzzy artmap architecture with adaptive featureweights based on onicescu's informational energy," *International Journal of Computers Communications & Control*, vol. 4, no. 2, pp. 104–117, 2009.

[25] A.-H. Tan and H.-S. V. Soon, "Concept hierarchy memory model: A neural architecture for conceptual knowledge representation, learning, and commonsense reasoning," *International Journal of Neural Systems*, vol. 7, no. 3, p. 305, 1996.

[26] "Deep learning for java," https://deeplearning4j.org/lstm.html, accessed: 2018-05-01.

[27] L. Kaufman and P. J. Rousseeuw, *Finding groups in data: an introduction to cluster analysis*. John Wiley & Sons, 2009, vol. 344.

Shan Gao received the Bachelors degree in computer science from National University of Singapore, Singapore, in 2009. He is currently pursuing the Ph.D. degree in Interdisplinary Graduate School, Nanyang Technological University, Singapore. Prior to his Ph.D. study, he was a project officer with the Emerging Research Lab in Nanyang Technological University, Singapore. His current research interests include human activity modeling, brain-inspired modeling of multimemory system, and causal reasoning.

Ah-Hwee Tan (SM'04) received the B.Sc. (Hons.) and M.Sc. degrees in computer and information science from the National University of Singapore, Singapore, in 1989 and 1991, respectively, and the Ph.D. degree in cognitive and neural systems from Boston University, Boston, MA, USA, in 1994. He is currently a Professor of Computer Science and the Associate Chair (Research) of the School of Computer Science and Engineering at Nanyang Technological University (NTU). Prior to joining NTU, he was a Research Manager with the Institute for Infocomm Research, Agency for Science, Technology and Research (A*STAR), Singapore, spearheading the Text Mining and Intelligent Agents research programs. His current research interests include cognitive and neural systems, brain inspired intelligent agents, machine learning, knowledge discovery, and text mining. Dr. Tan is an Editorial Board Member of IEEE ACCESS and Associate Editor for the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS and the IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS: SYSTEMS.

Rossi Setchi received her Double Honours degree in Mechanical and Electrical Engineering (first class, Summa Cum Laude) from Moscow Technological University in Russia, and her PhD from Cardiff University, UK. She joined the School of Engineering as a Lecturer in 2000; she was promoted to Senior Lecturer in 2007 and Chair in 2011. She leads the Knowledge Engineering Systems (KES) at the School of Engineering. She has a distinguished track record of research in a range of areas including AI, robotics, systems engineering, Cyber-Physical Systems and Industry 4.0, and, in particular, has built an international reputation for excellence in knowledge-based systems, computational semantics and human-machine systems. Over a period of 20 years, Professor Setchi has published over 200 peer-reviewed, high-quality publications and has been able to secure, with colleagues, external grant support totalling more than £22 million. She has collaborated with over 20 UK and 30 overseas universities, 15 research organisations and 30 industrial companies from more than 20 countries in Europe, Asia and Australia, and has provided research leadership on over 25 collaborative projects funded by UK and overseas funding bodies. Professor Setchi is Chartered Engineer, Chartered IT Professional, and European Engineer. She is Fellow of the Institution of Mechanical Engineers, Fellow of the Institution of Engineering and Technology, Fellow of the British Computer Society, and Senior Member of IEEE. Rossi is member of IFIP Technical Committee on Computer Aided Innovation (WG5.4), IFAC Technical Committee on Human-Machine Systems (WG4.5) and IEEE Task Force on Human-like Intelligence of the IEEE Computational Intelligence Society. She acts as Associate Editor of the International Journal of Systems Science (IJSS).