2023 IEEE International Conference on Robotics and Automation (ICRA 2023)
May 29 - June 2, 2023. London, UK

A Deep Learning Human Activity Recognition Framework for Socially Assistive Robots to Support Reablement of Older Adults

Fraser Robinson, *Student Member* and Goldie Nejat, *Member, IEEE*

***Abstract*— Many older adults prefer to stay in their own homes and age-in-place. However, physical and cognitive limitations in independently completing activities of daily living (ADLs) requires older adults to receive assistive support, often necessitating transitioning to care centers. In this paper, we present the development of a novel deep learning human activity recognition and classification architecture capable of autonomously identifying ADLs in home environments to enable long-term deployment of socially assistive robots to aid older adults. Our deep learning architecture is the first to use multimodal inputs to create an embedding vector approach for classifying and monitoring multiple ADLs. It uses spatial mid-fusion to combine geometric, motion and semantic features of users, environments, and objects to classify and track ADLs. We leverage transfer learning to extract generic features using the early layers of deep networks trained on large datasets to apply our architecture to various ADLs. The embedding vector enables identification of unseen ADLs and determines intra-class variance for monitoring user ADL performance. Our proposed unique architecture can be used by socially assistive robots to promote reablement in the home via autonomously supporting the assistance of varying ADLs. Extensive experiments show improved classification accuracy compared to unimodal/dual-modal models and the ADL embedding space also incorporates the ability to distinctly identify and track seen and unseen ADLs.**

I. Introduction

The growing population of adults over 60 years of age is estimated to double to 2.1 billion by the year 2050 [1]. The overall aging process can potentially result in difficulties with completing activities of daily living (ADLs) such as personal hygiene, eating, and dressing due to functional limitations and cognitive decline [2]. Reablement is an early intervention strategy that aims to enhance [3]: 1) an older adult's physical and cognitive functioning; 2) increase or maintain their ADL independence; and 3) minimize their need for long-term care assistance. This strategy moves caregivers away from the "*do for*" norm (i.e., doing the ADL for the older adult) to the "*do with*" approach (supporting the older adult as they complete the ADL) [4]. Reablement trains or retrains skills for maintaining or increasing ADL abilities of multiple ADLs to prolong independence and improve overall quality of life [3]. This continued independence can help keep older adults living in their own homes (aging-in-place); aligning with their own preferences. Furthermore, reablement can reduce healthcare costs and the demands put on long-term care [5].

Early assessments of reablement programs with caregivers show they have been effective at enhancing ADL capabilities and improving overall health of older adults [6]. However, these programs are currently constrained by staffing resources, lack of qualitative data for describing activity performance, and the need for customization to the preferences of individual older adults [7]. Assistive technologies including robots can be a viable alternative to deploy such reablement programs [8]. Socially assistive robots (SARs) are an assistive technology which can provide activity specific social assistance, while also monitoring ADL performance and adapting to user preferences and needs [9]–[12]. A significant challenge in the long-term deployment of SARs into the private homes of older adults is that the current level of SAR activity autonomy is limited to very specific ADLs in structured scenarios [13].

In this paper, we address this specific challenge by presenting the development of a novel deep learning ADL recognition and classification architecture for SARs to intelligently assist with reablement strategies. Our work is the first to consider the utilization of an end-to-end deep multimodal neural network that can simultaneously learn user, environment, and object feature representations to generate an ADL embedding vector capable of classifying and monitoring numerous diverse ADLs from personal grooming to eating. Our main contributions are: 1) the design of a vector embedding approach for ADL representation using multimodal data and deep neural networks, 2) the utilization of a unique spatial mid-fusion paradigm for synthesizing geometric, motion, and semantic features to unify their dimensions and spatial reference, and 3) an ADL classification model for extracting generic ADL features from various seen and unseen ADL classes trained on older adults to enable robot autonomy for reablement.

II. Related Work

Existing work in human activity recognition (HAR) utilizes a variety of methods to improve accuracy, enable unsupervised learning, and adapt to specific scenarios and users. This section provides a detailed discussion on: 1) multimodal deep learning networks for HAR, 2) embedding of feature spaces for HARs, and 3) existing HAR methods used by socially assistive robots.

A. Multimodal Deep Learning Networks for HAR

Recent HAR research has focused on using data specific operations such as graphical convolution networks (GCNs) [14], [15] and learned fusion techniques [16] to combine multimodal inputs in order to extract complimentary features. These methods have combined a variety of inputs including human skeleton pose information [14], [15], RGB video [14]–[16], and motion information between frames [16].

In [14], both RGB video and 3D poses were used to extract visual features for spatial embedding and pose driven attention using a Video-Pose Network consisting of a combination of GCNs and spatio-temporal convolution networks in order to classify indoor activities, such as putting

This research is supported by AGE-WELL Inc., an NSERC CREATE HeRo fellowship and the Canada Research Chairs Program.
F. Robinson and G. Nejat are with the Autonomous Systems and Biomechatronics Laboratory of the University of Toronto, Toronto, ON M5S3G8 CAN. (e-mail: fraser.robinson@mail.utoronto.ca, nejat@mie.utoronto.ca).

on headphones and clapping for monitoring of human behavior. End-to-end training with 3D ConvNet using a regularized loss term combining cross-entropy, embedding loss, and an attention regularizer resulted in significant improvements in classification accuracy for subtle actions such as reading compared to single mode networks using only RGB video streams. In [15], the architecture proposed in [14] was further extended through the incorporation of two separate distillation training sessions. Distillation transferred knowledge of pose to the feature extraction layers for improved model speed using only the RGB video input.

In [16], an RGB video stream and a motion information stream obtained from persistence of appearance (PA) were both used by a spatio-temporal convolutional neural network (CNN) with modality specific attention and late fusion for ADL classification. Training was accomplished using classification loss to learn consensus attention between the two modalities. Testing on segmented/unsegmented RGB video data of users performing ADLs, such as eating with a fork, showed improved accuracy over using a single modality.

B. Embedding of Feature Spaces for HARs

Embeddings of feature spaces are used to learn low-dimensional vector representations of ADLs to reduce the dimensionality of categorical information. They are used for data visualization [17], and classification [14], [18]–[20].

In [17], accelerometer and gyroscope sensory data from users completing ADLs was reduced to an embedding vector of activity features. An Autoencoder based on a Long-Short Term Memory Recursive Neural Network architecture was trained to reduce and reconstruct the sensory data for training the embedding. Temporal features were embedded using a sequence of recursive convolutions for activities of variable lengths. 2D Visualization of the embeddings based on stochastic neighbor embedding (t-SNE) [21] showed that the embedded features had improved inter-class separation compared to handcrafted features for the same data.

In [14], feature embeddings were used for multimodal fusion to improve the classification accuracy of the Video-Pose Network. An intermediate spatial embedding space was developed by combining RGB video visual features and pose spatial features. Embedding loss was added to the training loss function which improved inter-class separation.

In [18], data from inertial measurement units (IMUs) was mapped to a feature embedding vector to enable classification of sparsely labeled data. Feature embeddings were derived from the temporal input using CNNs and contrastive learning. Using the feature embeddings on partially labeled data for movement activities showed classification accuracy improvements over existing conventional autoencoders. In [19], a self-attention based approach was developed using a hierarchical window encoder (HWE) trained on temporal activity data using reconstruction loss to create feature embeddings. These embeddings were used for both classification training with unlabeled data and open-set recognition to identify unknown activities. A dense neural network with non-linear activations encoded and decoded the embedding features using an autoencoder architecture with an activity embedding vector. Training results showed improvements in closed-set classification over conventional autoencoders. Testing with unknown activities confirmed the ability of the network to identify such activities.

Robotic object manipulation has also used embedding vectors for classification. For example, in [20], point clouds of objects, natural language instructions, and robot manipulation trajectories were embedded in a common embedding space using linear deep neural layers with non-linear activations. The feature embeddings were used to select a new manipulation trajectory based on the embedding of an object-instruction pairing. Results showed improved accuracy and speed compared to embedding models using the same approach with larger and more complex embedding spaces.

C. HAR for Socially Assistive Robots

HAR has been used by social robots in human-robot interactions for numerous applications ranging from playing games to companionship [22]. A handful of SARs have been used to classify and track users performing ADLs using unimodal RGB video [11], [23], unimodal pose data from a depth sensor [12], [24], and multimodal data from RGB video and object-based sensors [11]. These activity tracking systems mainly use visual and depth data [11], [12], [23], [24] or natural language [12] classifiers, and heuristic rules [11], [23] to monitor and provide feedback to users via SARs.

In [11], the human-like Brian robot was used to facilitate meal eating of older adults. A sensor suite was used consisting of Wii motes for tracking custom IR utensils, a Kinect sensor for detecting user engagement from pose, and a meal tray with embedded load cells. The user and activity state, determined using the sensory information and Haar feature-based cascade classifiers and decision rules, and the robot state, based on task progression history, were used by a finite state machine (FSM) to determine the robot's assistive behavior. In [23], the Bandit robot was used to engage older adults in workout, imitation, and memory games. User hand and elbow joint positions were classified using an image segmentation algorithm and heuristic exercise rules. They were then used by an FSM to provide verbal praise by the robot for successful actions and corrections for unsuccessful actions.

In [24], the Leia robot was used to guide users in upper body exercises. An RGB-D sensor extracted user poses and a K-nearest neighbors classifier was used to classify these poses to determine exercise completion. An FSM was used to determine robot behaviors based on the exercise goal and user state. In [12], the human-like Casper robot learned to assist users in the ADL of making a cup of tea. The robot used a combination of Learning from Demonstration and reinforcement learning to determine its task-related assistive behaviors based on user cognitive functioning and activity states. Learning of task-related behaviors was based on demonstrators' speech using the onboard microphone and IBM Watson Speech-to-Text API [25] as well as gestures obtained by a depth camera and tracked using OpenNI [26].

D. Summary

The existing multimodal deep learning for HAR methods have shown improvements in accuracy, especially for ADLs with subtle movement differences [14], [15]. Incorporating the embedding of feature spaces provides a low-dimensional representation of data for classification of unseen activities [17]-[19]. To-date, these embedding vectors have been primarily developed for user wearable sensors. Furthermore, HAR for SARs assisting with ADLs have focused on one specific [11], [12], [24] or a limited set of activities [23],

without generalizability to the various multiple ADLs. Our research focuses on the development of a multimodal deep learning ADL recognition and classification architecture that incorporates a novel combination of vision based multimodal deep learning, spatial mid-fusion of multiple features, and feature embedding. Therefore, our work enables generalization to multiple ADLs and contexts to address current limitations of social robots which can only track user activity progress for specific tasks in structured environments.

III. Deep Learning Human Activity Recognition and Classification Architecture

The objective of our deep learning human activity recognition and classification architecture is to identify ADL classes for SARs to assist with and monitor performance during activity completion. The overall proposed architecture is presented in Fig. 1. Environment, action, and object information is obtained from RGB-D videos. These videos are separated into multiple inputs to extract pertinent features. Namely, a downsized RGB video is obtained from the RGB channels and used by the *Video Backbone Network* to obtain a combination of scene and motion features. The 3D pose of the user is simultaneously obtained from the RGB video and depth streams and used by the *Pose Backbone Network* to obtain 3D user motion features independent of the scene context. Single RGB images are also extracted and used by the *Object Detection Network* to obtain semantic features for objects used in performing the ADLs.

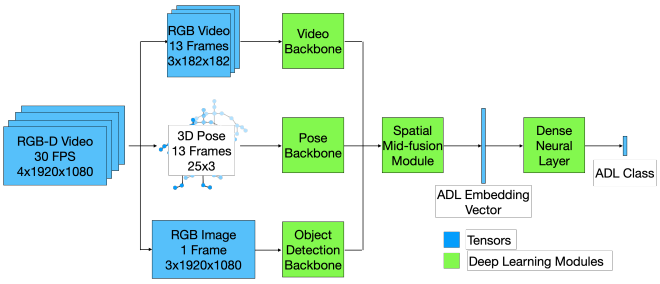

Figure 1. Proposed Deep Learning ADL Recognition and Classification Architecture.

The extracted feature set containing scene, motion, and semantic features from these backbone networks is then utilized by the *Spatial Mid-Fusion Module* to reshape and spatially scale the features for alignment before concatenation. This module condenses the features to a one-dimensional ADL embedding vector. The embedding vector is used by the *Dense Neural Layer* to determine the appropriate ADL class. The following subsections discuss these modules in more details.

A. Video Backbone Network

The objective of the *Video Backbone Network* is to extract scene and motion features, Fig. 2. The network takes as input a sequence of 13 video frames of size *182 x 182* pixels by down sampling and cropping from the 30 fps RGB video stream. The X3D small network [27] is adapted herein as the feature selection method as it is a deep network designed for optimized video feature extraction. The X3D small network progressively expands spatial and temporal convolutional layers based on the ResNet architecture [28] in dimensions of temporal duration, frame rate, spatial resolution, width, bottleneck width, and depth in order to iteratively add model depth to achieve accuracy while decreasing complexity [27].

The layers of the X3D small model used herein are ResNet Stem which consists of a 2D spatial convolution for spatial feature extraction, a 1D temporal convolution for temporal feature extraction, batch normalization [29] to increase training speed and model generalizability, and rectified linear units (ReLU) activation [30] to introduce non-linearities while avoiding vanishing or exploding gradients. Four successive ResNet Stages, each with varying branch quantities and compositions follow as demonstrated in Fig. 2. The output of the last ResNet Stage of X3D small is *13 x 192 x 6 x 6* (time, channels, feature grids) where each *6 x 6* video feature grid has inherent spatial understanding relative to the initial video.

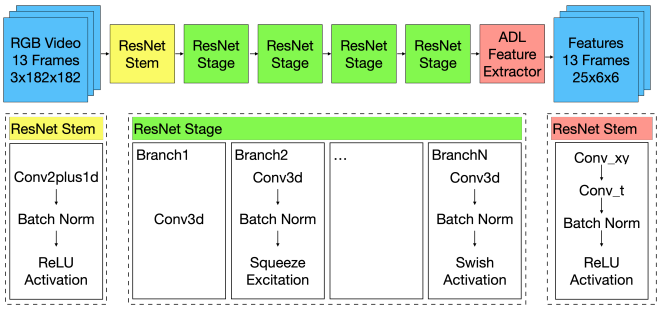

Figure 2. Video Backbone Network Architecture.

We have designed an ADL Feature Extractor to select the most significant video features for classifying ADLs using the general extracted features from the X3D layers as input. The ADL Feature Extractor uses a spatial convolution for geometric features, a temporal convolution for motion features, batch normalization for generalizability and ReLU activation for non-linearity. The final output is ADL feature grids of size *13 x 25 x 6 x 6.*

B. Pose Backbone Network

The *Pose Backbone Network* extracts scene and scale invariant pose motion action features. The input matches the temporal sampling of the *Video Backbone Network* with 13 frames, each with 25 skeleton joints having x_s,y_s,z_s positions. We have designed the *Pose Backbone Network* to consist of parallel paths for nearby, faraway, and positional joint motion features using GCN [31], self-attention [32], and skip connections, respectively, as shown in Fig. 3. The parallel branches are concatenated into a single tensor and passed to another GCN stage for joint variant motion features. The reshaped output is 13 frames and 25 channels of *6 x 6* feature grids for multimodal fusion, where each of the 25 channels is associated with a specific human skeleton joint. GCN stages are used to extract motion features independent of the environment by using message passing convolutions between nodes. In this work, human skeletons are transformed to a graph datatype where the nodes represent the 25 discrete skeleton joints and edges represent physical connections between adjacent joints.

For the GCN stage in the parallel section, the input data is the x_s,y_s,z_s position of each skeleton joint which is convolved with positions of adjacent joints for spatial feature extraction. In parallel, the same joint position data is also used in the self-attention module to transfer the data between nodes in the graph using a summation of weights. To determine the weights, the self-attention module assigns each node in the graph a query, key, and value grouping learned during training [32]. For each node, queries are compared to the keys of other

nodes and the resulting matching scores are multiplied by the attention node values using a dot-product of weights. The skip connection is a direct data transfer path to pass the positional input data to the next layer. The second GCN stage uses message passing to generate joint dependent motion features. The dimensions of the skeleton joint data after the second GCN stage are *13 x 25 x 36*. Reshaping is then performed on pose motion features in order to match the grid shape of the *Video Backbone Network* output so that the two feature grids can be concatenated in the *Spatial Mid-Fusion Module.*

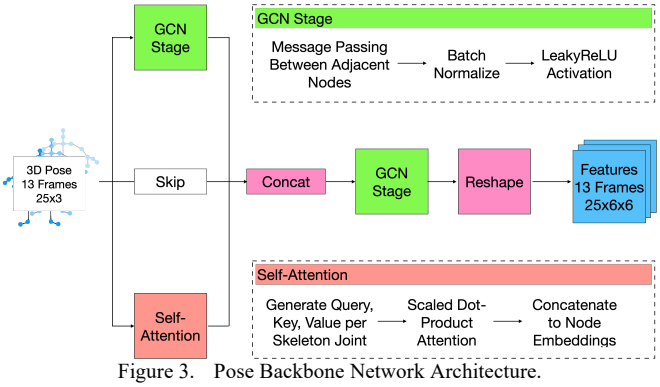

Figure 3. Pose Backbone Network Architecture.

C. Object Detection Backbone Network

The *Object Detection Backbone* is used to identify and localize objects in the scene during ADL classification. A rolling window approach is used to ensure a new RGB image is acquired with each timestep. The RGB images have an input size of *3 x 1920 x 1080* to use the full resolution available from the video stream to improve detection accuracy. Our *Object Detection Network* uses YOLOv5m60 [33] to extract object features from ADL-based home environments, Fig. 4. YOLOv5 was selected as it is a state-of-the-art real-time detector for household objects.

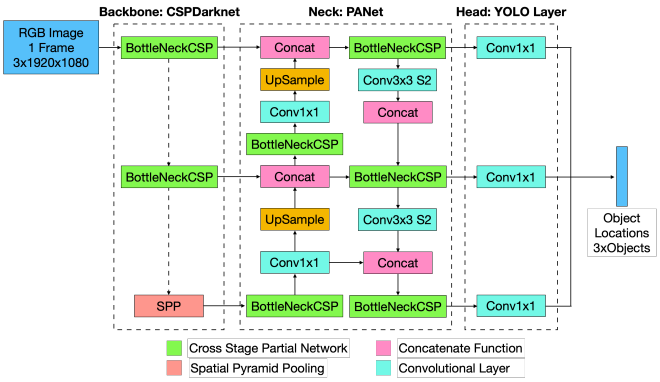

Figure 4. Object Detection Backbone Network Architecture (YOLOv5) Adapted from [33].

YOLOv5 uses: 1) a Cross Stage Partial (CSP) Network [34] approach to Darknet [35] for extracting high-level spatially invariant features while avoiding unnecessary duplicate gradients, 2) a Path Aggregation Network (PANet) [36] neck layer to use spatial features from each network layer to segment objects, and 3) three individual convolution layers to output object confidence scores. In Darknet, Spatial Pyramid Pooling (SPP) is used to perform information aggregation on inputs with varying sizes [37]. The output of the *Object Detection Backbone Network* is parsed to yield a list of object classes and their x_O,y_O locations. Object classes that were obtained from the COCO dataset [38] included indices 0 (person) and 31-80 (varying household objects, e.g., chair, bottle). A low confidence threshold of 0.25 was used to reduce the potential of false positives from scene and object variation.

D. Spatial Mid-Fusion Module

We have developed a *Spatial Mid-Fusion* module to reshape, scale, and concatenate geometric, motion, and semantic features from the three *Video*, *Pose*, and *Object Backbone Network* modalities, Fig. 5. The size of the input to the *Spatial Mid-Fusion* module is 14 timesteps (13 temporal frames from video/pose and 1 frame for object detection) with 50 channels of feature grids of size *6x6; 14 x 50 x 6 x 6*. The *Spatial Mid-Fusion* consists of: 1) a skip connection for video features to propagate the video feature grids to later layers, and 2) reshaping and scaling on both the skeleton joint motion data for pose features and on the object positions for spatial features. A concatenation step combines video and pose features along the channel dimension and then combines the object location features in the temporal dimension.

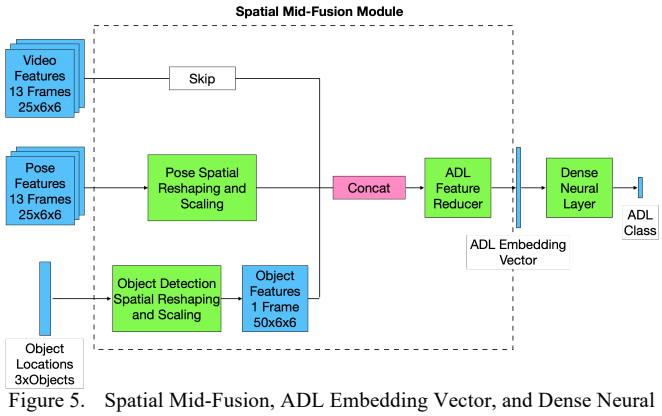

Figure 5. Spatial Mid-Fusion, ADL Embedding Vector, and Dense Neural Layers Modules for ADL Classification.

The pose spatial reshaping and scaling sub-module uses a series of mathematical operations to add spatial context to the pose features relative to the video spatial context. It takes as input the output of the *Pose Backbone Network* (*6 x 6* feature girds *x 25* skeleton joints *x 13* time steps) and uses the x_s,y_s position of each skeleton joint for creating 2D distance maps. For each node, these distance maps are determined by first initializing a Spatial Map *S* of dimensions *6 x 6 x 2* which contains *x,y* positions from -1 to 1 in equal increments. Next, the inverse Euclidian distances are calculated between the normalized x_s,y_s skeleton joint node position and each *x,y* position in *S* to obtain the distance grid *D* of size *6 x 6*. *D* represents the position of the node as a heatmap, where larger values are closer to this node in 2D space. Given a joint feature grid *F*, the new feature grid *F'* is calculated as $F' = D \times F$.

The object detection spatial reshaping and scaling sub-module uses the list of potential objects and their x_O,y_O locations. For each object, we initialize an identity matrix *I* and a Spatial Map *R* identical to *S* with sizes of *6 x 6* and *6 x 6 x 2*. Next, the inverse Euclidian distances between the object location x_O,y_O and each *x,y* position in *R* are used to form the distance grid *E* of size *6 x 6*. The object feature grid *G* is then calculated as $G = I \times E$. If multiple objects of the same class exist, object feature grid *G'* is $G' = G' \times E$, where *E* is the distance grid for each successive object within the same class.

Our new ADL Feature Reducer consists of: 1) a 2D spatial convolution layer for spatial feature extraction between newly

fused feature grids, 2) a 1D temporal convolution layer for temporal feature extraction and batch normalization, and 3) (ReLU) activation. Data is then flattened into a single vector of length 25,200 and passed to a linear neural layer to condense the features and create spatio-temporal dependences. Within the linear layer, batch normalization improves generalizability, leaky ReLU [39] activation limits vanishing gradients, and a dropout rate of 0.2 decreases overfitting [40]. The output is the ADL embedding vector of size 128.

ADL Embedding Vector: The ADL Embedding Vector is a low-dimensional representation of a specific ADL containing geometric, motion, and semantic features that are dependent on action timing, locations, motions, and object interactions. The size of the embedding vector follows the dimensionality reduction of the network architecture such that classification accuracy is unaffected. The nature of the embedding space results in ADLs with feature similarity being close in proximity to one another using metrics such as Euclidian distance. The ability to compare features in a low-dimensional space enables contextualization of unseen ADLs based on which existing ADL centroids have the lowest distance to the embedding of the new ADL. Within an ADL class, variations in intra-class embedding vector values determine if an ADL is being performed correctly overtime. ADL embeddings learned from relatively small sets of training data enable generalization within the range of observable features within the dataset. Given that datasets for supervised learning are diverse, the ADL embedding can generalize to new data within the known feature variations, eliminating the need for fully supervised training and decreasing data cost.

F. Dense Neural Layer

The *Dense Neural Layer* consists of batch normalization and a single fully connected linear layer. These determine scale independent feature interactions within the ADL embedding vector for classification. A dropout rate of 0.5 is used to classify the ADL embedding vector to an ADL class. The output of the *Dense Neural Layer* is the probabilities for each of the ADL classes.

G. Transfer Learning

Deep transfer learning is used for both the *Video Backbone* and *Object Detection Backbone Networks*. For the *Video Backbone Network*, transfer learning uses the first five layers of X3D small as a spatio-temporal feature extractor. X3D small is pretrained for classification of human activities from the Kinetics dataset [41]. For the *Object Detection Backbone Network*, the entirety of YOLOv5 is pretrained on the COCO dataset [38] for precise location detection of everyday objects in diverse environments.

IV. Architecture Training

Two variations of the architecture were trained using the ETRI-Activity-3D dataset [42] and the Toyota Smarthome with Refined Skeleton Data V1.2 dataset [43] to show robustness to different datasets. Training used gradient descent based on classification loss.

ETRI-Activity-3D dataset (ETRI): This dataset contains 112,620 samples of 55 activities performed by 50 younger and 50 older adult subjects [42]. Each sample contains an RGB video stream, a depth map, and a skeleton sequence of 3D joint positions. We chose ETRI as the primary dataset due to its inclusion of activities that directly correspond to typical ADLs performed older adults. It was used for hyperparameter tuning including the depth of layers and the number of convolutional channels. The ADL classes used in our training include: eating food with a fork, taking medicine, drinking water, brushing teeth, washing hands, washing face, hanging out laundry, putting on jacket, taking off jacket, putting on/taking off shoes, and brushing hair.

Toyota Smarthome with Refined Skeleton Data V1.2 dataset (Smarthome): This dataset consists of 31 activity classes in 16,000 samples of RGB video, depth video, and human skeleton pose sequences of older adults in smart home environments [43]. We trained with all 31 classes including basic activities such as "take pills" and compounded activities that have a distinct class such as "cook and cleanup" and "cook and cut". As the pose data from Toyota Smarthome contains 13 skeleton joints rather than 25, architectural modifications to the GCNs were required.

Both datasets were randomized using PyTorch random sampling utilities to ensure an even distribution of ADL classes between the training, validation, and test sets. The data was split into the standard 70% training, 20% validation, and 10% testing sets. Training was accomplished with a learning rate of 2×10^{-4}, a batch size of 128 and 20 epochs. Cross entropy [44] was used for classification loss to consider class confidence. The Adam optimizer [45] was used to introduce stochastic behavior for faster convergence using gradient descent. Early stopping was used to select the model with the lowest validation loss. Training loss stabilized after 15 epochs for ETRI and 7 for Smarthome, where training accuracy was 99.9% and 99.7%, respectively. Validation accuracies of 86.9% were obtained for ETRI with optimal hyperparameter selection and 74.1% for Smarthome with more challenging data and without optimized hyperparameters.

V. Experiments

We performed several experiments to evaluate the performance of our ADL detection and classification architecture. Network performance is measured by classification accuracy on test sets from the ETRI and Smarthome datasets. We determined the effect of adding individual modalities using an ablation study which compared our multimodal to dual-modal (as primarily used in the literature) and unimodal networks. For evaluating the quality of ADL vector embeddings, the ETRI test set embeddings were used to construct an embedding space for visualization using t-SNE and numerical analysis of distance metrics. Comparison to an embedding space developed solely using RGB video is conducted to measure the impact of multimodality on generating ADL vector embeddings; as embeddings using visual data are a fairly new procedure.

A. Experiment #1: Architecture Testing

To evaluate classification accuracy, we tested our multimodal network on the two large aforementioned ADL datasets. On the ETRI test set with 11 ADL classes (with only basic activities) our low-latency architecture obtained an accuracy of 86.0%, the first to consider real-time applications on ETRI [46]. On the Smarthome test set with high duration variation, basic and compounded activities, and 31 classes, the accuracy obtained was 73.5%. Cross-subject accuracies for Smarthome have been reported to be below 70% [16].

B. *Experiment #2: Ablation Study*

We performed an ablation study that removed single modalities from our three-modality architecture. Table I shows classification accuracy results for ETRI. Multimodality improves model accuracy compared to unimodal and dual-modal networks with the same architecture. The proposed architecture benefits from combining complementary feature data for ADLs to improve classification performance.

TABLE I. MODEL MODALITY TEST ACCURACY

Modality	Test Accuracy
Pose	73.7%
RGB Video	75.1%
Pose and RGB Video	82.4%
Multimodal (Pose+RGB Video+Object)	**86.0%**

C. *Experiment #3: ADL Embedding Performance*

The ADL embedding vector quality was evaluated using: 1) t-SNE [21] visualization to generate a low-dimensional and high contrast data representation based on neighboring samples by measuring similarities between points in the high-dimensional space, and 2) intra-class variance and inter-class distance. Embedding vectors were created using the ETRI test set and concatenated into the embedding space.

We use t-SNE to map the 128 dimensions of the ADL vector embeddings to 2D cartesian plots. ADL vector embeddings from the multimodal network and the RGB video network were compared, Fig 6(a) and (b). RGB video was selected as the unimodal model of comparison since it showed higher accuracy than pose for ETRI as shown in Table I. The t-SNE visualization shows that the multimodal network has more distinct groupings of similar classes. Namely, when using the RGB video modality, classes with similar environments and large-scale movements such as washing hands or face, and brushing teeth are overlapping in the embedding space (with low separation centroids). Using our multimodal embedding, the centroids have higher separation and visually superior inter-class distinction for similar ADLs. Fig. 6 shows distinctions between clothing-based ADLs (putting on/taking off jacket) and consumption-based ADLs (eating, drinking, and taking medicine).

Intra-class variance represents the variance in Euclidian distances of vector embeddings from the same class. On the other hand, inter-class distance measures distances between centroids of classes. We use Euclidian distance between embeddings as our metric as it provides equal weighting of features and computational efficiency [47]. Table II shows both intra-class variance and inter-class distance for the multimodal and RGB video embedding spaces. The multimodal embedding space has less variation within classes and greater separation between classes. The lower maximum intra-class variance shows *greater within class* grouping in the embedding space for classes with high levels of activity variability such as drinking water which can occur in many different environments. The higher minimum inter-class distance (by a factor of 1.79) increases separation between the most similar ADL classes within the embedding space.

We tested the contextualization of unseen ADLs by using 5 new samples for each of 5 new ADLs (25 inputs) from the ETRI dataset in our multimodal architecture to obtain their ADL embeddings, Fig. 6(c). These activities included "doing freehand exercises", "spreading bedding/folding bedding", "putting on/taking off glasses", "putting on cosmetics", and "peeling vegetables". The unseen ADLs of "putting on/taking off glasses" and "putting on cosmetics" are near the trained ADL of "brushing hair", as they are similar ADLs with subtle arm movements. However, there is clear distinction between their distributions indicting that they are unique ADLs. For ADLs that have large distributions (e.g., doing freehand exercises), their centroids also show large separations from the centroids of known ADLs, again emphasizing uniqueness.

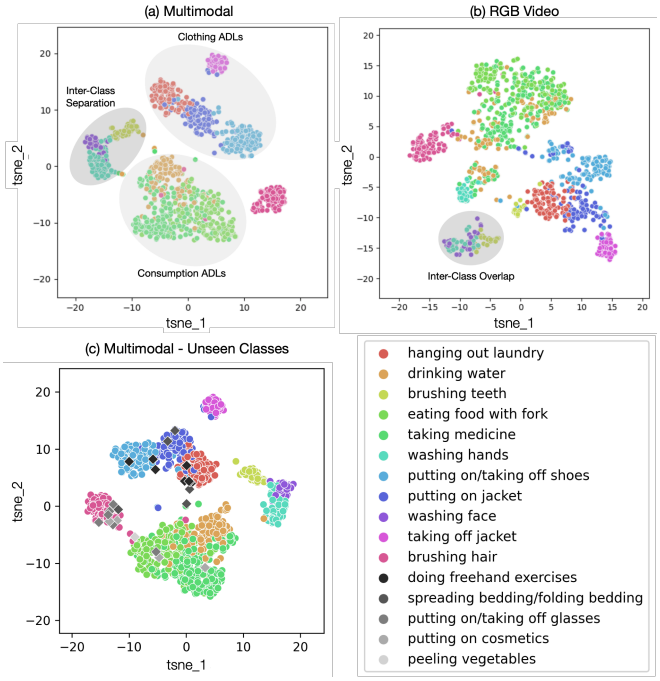

Figure 6. ADL Embedding Spaces.

TABLE II. INTRA-CLASS VARIATION AND INTER-CLASS DISTANCE FOR EMBEDDING SPACES

Modality	Mean Intra-Class Variance	Maximum Intra-Class Variance	Mean Inter-Class Distance	Minimum Inter-Class Distance
RGB Video	0.67	0.99	3.97	1.90
Multimodal	**0.55**	**0.78**	**4.60**	**3.40**

VI. CONCLUSION

We present the development of a novel multimodal deep learning ADL recognition and classification architecture for SARs to assist with ADL reablement. The architecture can simultaneously learn user, environment, and object feature representations to generate an ADL embedding vector capable of classifying numerous diverse ADLs via its three backbone networks. Spatial mid-fusion reshapes and scales these features into unified feature grids, while condensing them into the ADL embedding vector. Transfer learning extracts generic features from early layers of the network to apply our architecture to various ADLs by training on large datasets. Results show higher ADL classification accuracy for our multimodal method over unimodal/dual-modal methods. Visualization of the ADL embedding space shows the inter-class separation necessary for training with unlabeled data and groupings of similar activities to support contextualization of unseen ADLs. Future work includes integrating the ADL embedding space on physical SARs for classifying and monitoring user performance overtime to provide assistance.

REFERENCES

[1] World Health Organization, "Ageing and health." WHO Newsroom, Fact Sheets, Oct. 04, 2021. Accessed: Nov. 21, 2021. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/ageing-and-health

[2] D. E. Bloom, D. Canning, and A. Lubet, "Global Population Aging: Facts, Challenges, Solutions & Perspectives," *Daedalus*, vol. 144, no. 2, pp. 80–92, Apr. 2015, doi: 10.1162/DAED_a_00332.

[3] F. Aspinal, J. Glasby, T. Rostgaard, H. Tuntland, and R. G. J. Westendorp, "New horizons: Reablement - supporting older people towards independence," *Age Ageing*, vol. 45, no. 5, pp. 574–578, Sep. 2016, doi: 10.1093/ageing/afw094.

[4] T. H. Rooijackers *et al.*, "Economic Evaluation of a Reablement Training Program for Homecare Staff Targeting Sedentary Behavior in Community-Dwelling Older Adults Compared to Usual Care: A Cluster Randomized Controlled Trial," *Clin. Interv. Aging*, vol. Volume 16, pp. 2095–2109, Dec. 2021, doi: 10.2147/CIA.S341221.

[5] K. M. Hjelle, H. Tuntland, O. Førland, and H. Alvsvåg, "Driving forces for home-based reablement; a qualitative study of older adults' experiences," *Health Soc. Care Community*, vol. 25, no. 5, pp. 1581–1589, Sep. 2017, doi: 10.1111/hsc.12324.

[6] C. Pettersson and S. Iwarsson, "Evidence-based interventions involving occupational therapists are needed in re-ablement for older community-living people: A systematic review," *Br. J. Occup. Ther.*, vol. 80, no. 5, pp. 273–285, May 2017, doi: 10.1177/0308022617691537.

[7] K. Vik and A. Eide, "Older adults who receive home-based services, on the verge of passivity: the perspective of service providers," *Int. J. Older People Nurs.*, vol. 8, no. 2, pp. 123–130, May 2013, doi: 10.1111/j.1748-3743.2011.00305.x.

[8] J. Morato, S. Sanchez-Cuadrado, A. Iglesias, A. Campillo, and C. Fernández-Panadero, "Sustainable Technologies for Older Adults," *Sustainability*, vol. 13, no. 15, p. 8465, Jul. 2021, doi: 10.3390/su13158465.

[9] F. Robinson, Z. Cen, H. E. Naguib, and G. Nejat, "Socially Assistive Robotics and Wearable Sensors for Intelligent User Dressing Assistance," presented at the 2022 31st IEEE International Conference on Robot and Human Interactive Communication (RO-MAN), Naples, Italy.

[10] L. Woiceshyn, Y. Wang, G. Nejat, and B. Benhabib, "A Socially Assistive Robot to Help With Getting Dressed," in *Design of Medical Devices Conference*, Minneapolis, Minnesota, USA, Apr. 2017, p. V001T11A012. doi: 10.1115/DMD2017-3467.

[11] D. McColl and G. Nejat, "Meal-Time with a Socially Assistive Robot and Older Adults at a Long-term Care Facility," *J. Hum.-Robot Interact.*, vol. 2, no. 1, pp. 152–171, Mar. 2013, doi: 10.5898/JHRI.2.1.McColl.

[12] C. Moro, G. Nejat, and A. Mihailidis, "Learning and Personalizing Socially Assistive Robot Behaviors to Aid with Activities of Daily Living," *ACM Trans. Hum.-Robot Interact.*, vol. 7, no. 2, pp. 1–25, Oct. 2018, doi: 10.1145/3277903.

[13] T. L. Mitzner, T. L. Chen, C. C. Kemp, and W. A. Rogers, "Identifying the Potential for Robotics to Assist Older Adults in Different Living Environments," *Int. J. Soc. Robot.*, vol. 6, no. 2, pp. 213–227, Apr. 2014, doi: 10.1007/s12369-013-0218-7.

[14] S. Das, S. Sharma, R. Dai, F. Bremond, and M. Thonnat, "VPN: Learning Video-Pose Embedding for Activities of Daily Living." arXiv, Jul. 06, 2020. Accessed: Jul. 12, 2022. [Online]. Available: http://arxiv.org/abs/2007.03056

[15] S. Das, R. Dai, D. Yang, and F. Bremond, "VPN++: Rethinking Video-Pose embeddings for understanding Activities of Daily Living," *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 1–1, 2021, doi: 10.1109/TPAMI.2021.3127885.

[16] H. Kim, D. Kim, and J. Kim, "Learning Multi-modal Attentional Consensus in Action Recognition for Elderly-Care Robots," in *2021 18th International Conference on Ubiquitous Robots (UR)*, Gangneung, Korea (South), Jul. 2021, pp. 308–313. doi: 10.1109/UR52253.2021.9494666.

[17] A. Ghods and D. J. Cook, "Activity2Vec: Learning ADL Embeddings from Sensor Data with a Sequence-to-Sequence Model." arXiv, Jul. 12, 2019. Accessed: Sep. 09, 2022. [Online]. Available: http://arxiv.org/abs/1907.05597

[18] Y. Jain, C. I. Tang, C. Min, F. Kawsar, and A. Mathur, "ColloSSL: Collaborative Self-Supervised Learning for Human Activity Recognition," *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, vol. 6, no. 1, pp. 1–28, Mar. 2022, doi: 10.1145/3517246.

[19] M. T. H. Tonmoy, S. Mahmud, A. K. M. M. Rahman, M. A. Amin, and A. A. Ali, "Hierarchical Self Attention Based Autoencoder for Open-Set Human Activity Recognition." arXiv, Mar. 07, 2021. Accessed: Jul. 12, 2022. [Online]. Available: http://arxiv.org/abs/2103.04279

[20] J. Sung, I. Lenz, and A. Saxena, "Deep multimodal embedding: Manipulating novel objects with point-clouds, language and trajectories," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, Singapore, Singapore, May 2017, pp. 2794–2801. doi: 10.1109/ICRA.2017.7989325.

[21] L. van der Maaten and G. Hinton, "Visualizing Data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2605, 2008.

[22] H. Mahdi, S. A. Akgun, S. Saleh, and K. Dautenhahn, "A survey on the design and evolution of social robots — Past, present and future," *Robot. Auton. Syst.*, vol. 156, p. 104193, Oct. 2022, doi: 10.1016/j.robot.2022.104193.

[23] J. Fasola and M. J. Mataric, "Using Socially Assistive Human–Robot Interaction to Motivate Physical Exercise for Older Adults," *Proc. IEEE*, vol. 100, no. 8, pp. 2512–2526, Aug. 2012, doi: 10.1109/JPROC.2012.2200539.

[24] S. F. R. Alves, M. Shao, and G. Nejat, "A Socially Assistive Robot to Facilitate and Assess Exercise Goals," in *Proceedings of the IEEE International Conference of Robotics and Automation Workshop on Mobile Robot Assistants for the Elderly*, Montreal, QC, Canada, 2019, p. 5.

[25] A. Lally, "Natural Language Processing With Prolog in the IBM Watson System," *Assoc. Log. Program. ALP Newsl.*, vol. 9, p. 4, 2011.

[26] N. Villaroman, D. Rowe, and B. Swan, "Teaching natural user interaction using OpenNI and the Microsoft Kinect sensor," in *Proceedings of the 2011 conference on Information technology education - SIGITE '11*, West Point, New York, USA, 2011, p. 227. doi: 10.1145/2047594.2047654.

[27] C. Feichtenhofer, "X3D: Expanding Architectures for Efficient Video Recognition." arXiv, Apr. 09, 2020. Accessed: Jul. 12, 2022. [Online]. Available: http://arxiv.org/abs/2004.04730

[28] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition." arXiv, Dec. 10, 2015. Accessed: Aug. 19, 2022. [Online]. Available: http://arxiv.org/abs/1512.03385

[29] S. Ioffe and C. Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift." arXiv, Mar. 02, 2015. Accessed: Aug. 19, 2022. [Online]. Available: http://arxiv.org/abs/1502.03167

[30] A. F. Agarap, "Deep Learning using Rectified Linear Units (ReLU)." arXiv, Feb. 07, 2019. Accessed: Sep. 09, 2022. [Online]. Available: http://arxiv.org/abs/1803.08375

[31] T. N. Kipf and M. Welling, "Semi-Supervised Classification with Graph Convolutional Networks." arXiv, Feb. 22, 2017. Accessed: Aug. 18, 2022. [Online]. Available: http://arxiv.org/abs/1609.02907

[32] A. Vaswani *et al.*, "Attention Is All You Need." arXiv, Dec. 05, 2017. Accessed: Aug. 04, 2022. [Online]. Available: http://arxiv.org/abs/1706.03762

[33] G. Jocher, "ultralytics/yolov5: v6.1 - TensorRT, TensorFlow Edge TPU and OpenVINO Export and Inference." Zenodo, Feb. 22, 2022. [Online]. Available: doi:10.5281/zenodo.6222936

[34] C.-Y. Wang, H.-Y. M. Liao, I.-H. Yeh, Y.-H. Wu, P.-Y. Chen, and J.-W. Hsieh, "CSPNet: A New Backbone that can Enhance Learning Capability of CNN." arXiv, Nov. 26, 2019. Accessed: Aug. 18, 2022. [Online]. Available: http://arxiv.org/abs/1911.11929

[35] J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger." arXiv, Dec. 25, 2016. Accessed: Sep. 15, 2022. [Online]. Available: http://arxiv.org/abs/1612.08242

[36] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path Aggregation Network for Instance Segmentation." arXiv, Sep. 18, 2018. Accessed: Aug. 18, 2022. [Online]. Available: http://arxiv.org/abs/1803.01534

[37] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition," vol. 8691, 2014, pp. 346–361. doi: 10.1007/978-3-319-10578-9_23.

[38] T.-Y. Lin *et al.*, "Microsoft COCO: Common Objects in Context," in *Computer Vision – ECCV 2014*, vol. 8693, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds. Cham: Springer International Publishing, 2014, pp. 740–755. doi: 10.1007/978-3-319-10602-1_48.

[39] B. Xu, N. Wang, T. Chen, and M. Li, "Empirical Evaluation of Rectified Activations in Convolutional Network." arXiv, Nov. 27, 2015. Accessed: Sep. 12, 2022. [Online]. Available: http://arxiv.org/abs/1505.00853

[40] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A Simple Way to Prevent Neural Networks from Overfitting," *J. Mach. Learn. Res.*, vol. 15.1, pp. 1929–1958, 2014.

[41] J. Carreira and A. Zisserman, “Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, Jul. 2017, pp. 4724–4733. doi: 10.1109/CVPR.2017.502.
[42] J. Jang, D. Kim, C. Park, M. Jang, J. Lee, and J. Kim, “ETRI-Activity3D: A Large-Scale RGB-D Dataset for Robots to Recognize Daily Activities of the Elderly,” in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Las Vegas, NV, USA, Oct. 2020, pp. 10990–10997. doi: 10.1109/IROS45743.2020.9341160.
[43] S. Das *et al.*, “Toyota Smarthome: Real-World Activities of Daily Living,” *IEEE Int. Conf. Comput. Vis. ICCV*, p. 10, Oct. 2019.
[44] Z. Zhang and M. Sabuncu, “Generalized Cross Entropy Loss for Training Deep Neural Networks with Noisy Labels,” *Adv. Neural Inf. Process. Syst.*, vol. 31, p. 11, 2018.
[45] D. P. Kingma and J. Ba, “Adam: A Method for Stochastic Optimization.” arXiv, Jan. 29, 2017. Accessed: Sep. 09, 2022. [Online]. Available: http://arxiv.org/abs/1412.6980
[46] X. Shu, J. Yang, R. Yan, and Y. Song, “Expansion-Squeeze-Excitation Fusion Network for Elderly Activity Recognition,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 8, pp. 5281–5292, Aug. 2022, doi: 10.1109/TCSVT.2022.3142771.
[47] P.-E. Danielsson, “Euclidean distance mapping,” *Comput. Graph. Image Process.*, vol. 14, no. 3, pp. 227–248, Nov. 1980, doi: 10.1016/0146-664X(80)90054-4.