IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version Dec. 31, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.DOI

ElderSim: A Synthetic Data Generation Platform for Human Action Recognition in Eldercare Applications

HOCHUL HWANG[1], CHEONGJAE JANG[1], GEONWOO PARK[1], JUNGHYUN CHO[1], AND IG-JAE KIM[1,2].

[1]Artificial Intelligence & Robotics Institute, Korea Institute of Science and Technology (KIST), Seoul 02792, Republic of Korea
[2]Division of Nano and Information Technology, KIST School, University of Science and Technology (UST), Seoul 02792, Republic of Korea

Corresponding author: Ig-Jae Kim (e-mail: drjay@kist.re.kr).

This work was supported by the Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. 2017-0-00162, Development of Human-care Robot Technology for Aging Society).

ABSTRACT To train deep learning models for vision-based action recognition of elders' daily activities, we need large-scale activity datasets acquired under various daily living environments and conditions. However, most public datasets used in human action recognition either differ from or have limited coverage of elders' activities in many aspects, making it challenging to recognize elders' daily activities well by only utilizing existing datasets. Recently, such limitations of available datasets have actively been compensated by generating synthetic data from realistic simulation environments and using those data to train deep learning models. In this paper, based on these ideas we develop ElderSim, an action simulation platform that can generate synthetic data on elders' daily activities. For 55 kinds of frequent daily activities of the elders, ElderSim generates realistic motions of synthetic characters with various adjustable data-generating options and provides different output modalities including RGB videos, two- and three-dimensional skeleton trajectories. We then generate KIST SynADL, a large-scale synthetic dataset of elders' activities of daily living, from ElderSim and use the data in addition to real datasets to train three state-of-the-art human action recognition models. From the experiments following several newly proposed scenarios that assume different real and synthetic dataset configurations for training, we observe a noticeable performance improvement by augmenting our synthetic data. We also offer guidance with insights for the effective utilization of synthetic data to help recognize elders' daily activities.

INDEX TERMS Classification algorithms, computer graphics, computer simulation, computer vision, supervised learning

I. INTRODUCTION

The need and importance of vision-based human action recognition (HAR) are growing in a wide range of eldercare services [1], including care robots [2], [3], smart surveillance [4], and health monitoring [5], [6]. Recently, the performance of vision-based HAR has been dramatically improved by deep learning methods [7]–[17], which require large-scale training datasets for accurate action recognition [7], [19]–[24] as mentioned in [18]. Accordingly, to train deep learning models to recognize elders' activities of daily living (ADL), we need large-scale datasets that contain activities acquired under various environments and conditions that we encounter in daily life.

However, most public datasets used in HAR, including the NTU RGB+D dataset [21], which is frequently used as a benchmark, either differ from or have limited coverage of elders' daily activities in many aspects. Even if they have a large number of samples with various action classes, only a few action classes of such datasets match elders' ADL. Moreover, they are usually acquired from laboratory environments that deviate from the places of daily living. The way or speed of actions may also differ from the elders' since they mostly consist of relatively young subjects' actions. These differences can induce inaccurate action recognition results when a model trained on such datasets is tested on data of elders' ADL [24].

Recently, some datasets of elders' ADL have been publicly available [24], [25]. However, due to the limited data acqui-

sition conditions, they often lack diversity in aspects such as background, camera viewpoint, and lighting condition. The low variations in a dataset can cause overfitting of deep learning models, especially for RGB-based HAR methods that are sensitive to the conditions above. An overfitted model will not generalize well and result in low recognition accuracy when applied to data obtained under conditions significantly different from the training datasets.

A naive approach to this problem is to build a dataset that reflects all the conditions that arise in diverse real-world household environments. However, given the diversity in data acquisition conditions such as camera view, lighting, background, and the type of actions, it is expensive and laborious to acquire such a dataset due to the combinatorial explosion [26], i.e., the number of data can increase exponentially. There also exist other difficulties in acquiring real data. For example, viewpoints are often restricted due to spatial limitations such as small-sized bathrooms or complex indoor environments. Personal privacy issues and physical limitations of the elders also make it more challenging to obtain a large-scale training dataset of good quality.

To compensate for the limitations of available datasets and the difficulties of acquiring real data, recent studies endeavor to generate automatically-labeled synthetic data from virtual environments [27]–[29], [51]–[53] and further use those data to train deep learning models and enhance action recognition performance [29], [52]. In such virtual environments, we can freely adjust aspects such as backgrounds, subjects (or synthetic characters), camera viewpoints, and lighting conditions. Therefore, it becomes possible to customize the dataset that contains a large number of realistic data as needed. If such synthetic data are appropriately utilized for training deep learning models, we can expect those data to help fill the holes that reside in real-world datasets, e.g., the limited coverage in camera viewpoints and lighting conditions or the severe gap from the target data in subjects' ages and backgrounds.

In this paper, based on the above ideas we develop ElderSim, an action simulation platform that can generate synthetic data on elders' daily activities. We visualize the daily living environment and the characters of ElderSim to be as close as possible to those of the real-world using a recent three-dimensional rendering and modeling software. Targeting the actual application to eldercare services, we model movements for 55 kinds of frequent daily activities of the elders and offer variability in data acquiring options such as camera viewpoints and lighting conditions that change over time, to name a few. To summarize, ElderSim generates realistic daily living activities of synthetic characters with several adjustable data-generating options and provides different output modalities including RGB videos, two-dimensional (2D) and three-dimensional (3D) skeleton trajectories to increase applicability further. As an illustrating dataset generated from ElderSim, we release KIST SynADL, a large-scale simulated synthetic dataset of elders' activities.

We use KIST SynADL in addition to real datasets to train state-of-the-art HAR models, and validate the effectiveness of augmenting synthetic data. Unlike previous data augmentation studies focusing primarily on some limited benchmark datasets and experimental scenarios, we propose several new scenarios to examine various aspects that arise from recognizing elders' ADL. Specifically, in addition to cross-view and cross-subject train/test splits, widely considered in the literature, we newly introduce cross-lighting, cross-age, and cross-dataset splits that assume the real and synthetic training datasets of different configurations. Here, the last two settings are focused more on the application to the elders. We also examine synthetic data augmentation for each of the three data modalities provided, namely RGB video, 2D and 3D skeleton. From the extensive experiments held with three action recognition models on four different real-world datasets, we show that augmenting our synthetic data for training increases recognition performance for most of the considered methods in various settings. We also offer some guidance with insights on utilizing synthetic data to help recognize elders' daily activities effectively. These points are of great importance since they can be easily combined with additional improvements in both deep learning models and objective functions for training to gain enhanced action recognition performance. The main pipeline of our work is described in Fig. 1.

In summary, the contributions in this paper are three-fold:

1) We propose a novel action simulation platform (ElderSim) that generates realistic motions of elders' activities of daily living based on user-adjustable data generation parameters.
2) With ElderSim, we generate a large-scale synthetic dataset (KIST SynADL) that covers 55 activity classes performed by synthetic elder characters in diverse backgrounds, viewpoints, and light conditions. ElderSim and KIST SynADL are publicly available in https://ai4robot.github.io/ElderSim.
3) We leverage KIST SynADL to train three human action recognition methods on several experimental splits using four real-world activity datasets and present remarkable performance improvement in action recognition. In addition, we provide some guidance with insights for the effective usage of synthetic data in action recognition.

The paper is organized as follows. Section II presents related works, and we elaborate on ElderSim along with the synthetic dataset generated from ElderSim in Section III. Section IV presents action recognition experiments augmenting our synthetic data. We conclude in Section V.

II. RELATED WORK

In this section, we introduce several HAR methods with various human activity datasets utilized in the literature. We also mention previous studies that exploit synthetic data to improve action recognition.

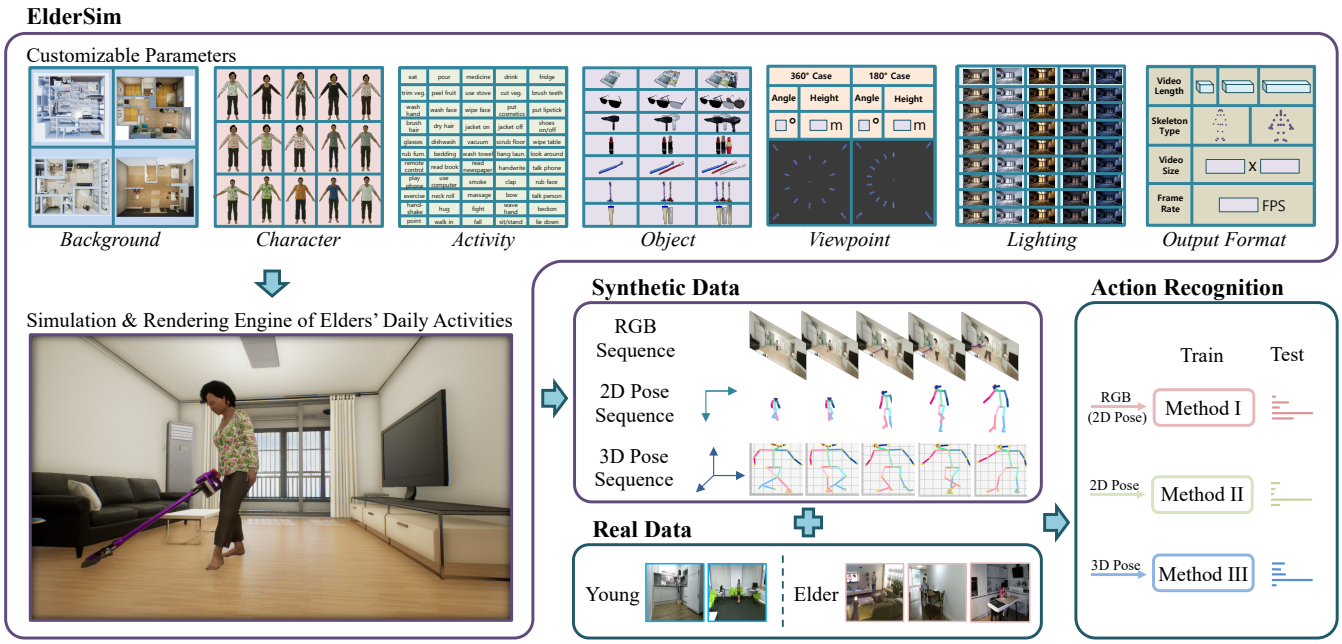


FIGURE 1. The synthetic data generation process of ElderSim and the main pipeline of our proposed work. ElderSim generates synthetic RGB video, 2D, and 3D skeleton data based on the data-generating options that are customized by the user. Here, we experimentally augment synthetic data on real ones and train three different action recognition methods (Method I: Glimpse [11], Method II: ST-GCN [12], Method III: VA-CNN [13]) to scrutinize the effects of using our generated data.

A. HUMAN ACTION RECOGNITION

Considering the temporal dimension along with the spatial dimension is essential for video understanding. Conventionally, these features were extracted by hand-crafted descriptors, including Histogram of Oriented Gradients (HOG) [30], Motion Boundary Histogram (MBH) [31], and Histograms of Optical Flow (HOF) [32], which are followed by a classifier such as Support Vector Machines (SVMs) for classification. Other methods used dense trajectories that track densely-sampled feature points and extract appearance and motion information with the previously mentioned descriptors along the trajectories [33], [34].

With the development of powerful computation hardware and large-scale activity datasets, deep learning methods achieved profound performance in action recognition. As being one of the convolutional neural network (CNN)-based methods [7]–[9], [35], [36], [35] computes spatial information from a still frame and samples temporal motion from multiple-frame dense optical flow. The information is then inserted into a two-stream network that consists of a spatial and temporal CNN for better training. [36] first introduced 3D convolution for action recognition, which extracts features from both the spatial and temporal dimensions with a 3D kernel. [7] tested varying architectures on their own dataset, Sports-1M, showing that the Slow Fusion model outperforms other structures with different connectivity in time and also proposed an architectural method using lowered-resolution inputs to speed up training without any performance loss. [8] built a C3D (Convolutional 3D) network with 3D kernels and empirically showed the optimal kernel size and network architecture for improved action recognition performance in large-scale video datasets. Unlike [8], [9] inflated the filters and pooling kernels of deep 2D image classification models (e.g., Inception-v1 with batch normalization [37]) into 3D to gain from the advantages of ImageNet pre-training. They marked up performance with an additional optical-flow stream which is trained separately and tested with averaged-predictions. Without the support of multimodal inputs, [10] computes a single modality of raw RGB to a two-stream design by passing each pathway with a different frame rate. Spatial meanings are captured through one stream with a higher frame rate while the temporal information is learned through the other stream with a lower frame rate. The architecture in [11] learns to predict the attention windows in the feature space; extracted from a global model. A set of recurrent architectures are used to track the unstructured windows and classify actions from RGB video inputs.

Skeleton-based HAR methods [12]–[16] have been studied to avoid various interference of RGB appearance while using simpler data that are coordinates of several joints and their derived forms. These methods mainly obtain the human skeletal structure utilizing depth sensors [38] or human pose estimation algorithms [39], [40]. [16] uses the main LSTM network with a spatial attention module and a temporal attention module that holds different attention levels to select discriminative joint inputs and frame outputs, respectively. The three networks are jointly trained for optimization as an end-to-end training method. More recently, initiated by [12], graph convolutional network-based action recognition studies tried to understand the skeletal information as a graph and extract features using CNNs. [14] represented the tree-based natural human body structure as a directed acyclic

graph for a better interpretation. They also adopt the two-stream method by feeding a graph that contains information of joints and bones to one network and another graph that contains the motion of joints and deformation of bones to the other network. [15] used temporally different-sized kernels instead of fixed ones as in [12] and added an additional spatial graph convolution layer branch to form a parallel structure. They further improved performance by using six modalities of input features including relative positions of joints and bones. Other multimodal fusion methods enhance performance by handling data of different domains, such as RGB and 3D skeleton [17].

B. REAL-WORLD ACTIVITY DATASETS

Diverse real-world human activity datasets have been publicly available with the emerging importance of robust human action recognition. Initial human activity datasets were relatively small-scale, having a small number of subjects and activity categories, until the early 2010s [41]–[44]. KTH [41], being one of the earliest databases, contains a single RGB modality of six action categories with simple motions such as *walk*, *run*, and *clap*. Depth map information was firstly provided with RGB in MSR Action3D [42], which focused on game console interaction-based motions, including *draw circle*, *forward kick*, *tennis swing*. As an extension of [42], MSR Daily Activity3D [43] covers living room daily activities, most containing human-object interaction captured by a Kinect sensor. RGBD HuDaAct [44] also deals with 12 daily activities of 30 students with RGB and depth modalities maintaining under 1,200 samples. Most of the early datasets contain only a few thousands of video samples and under 20 class categories, which allowed studies for hand-crafted methods without the support of deep learning. Large-scale activity datasets emerged along with the advance of data-hungry action recognition methods [7], [19]–[24]. The initial version of Kinetics [19], obtained from YouTube, included 400 activity classes having more than 300K samples in total. The dataset is now extended to contain 700 classes with approximately 650K video clips. PKU-MMD [20] provides untrimmed daily activity video sequences in four modalities of RGB, depth, infrared (IR), and skeleton for the research field of action detection. A more recent multimodal dataset, MMAct [23], was released with seven modalities: RGB, skeleton, acceleration, and other sensor signals. NTU RGB+D [21] and its updated version [22] are extensively used as a benchmark dataset in recent human action recognition literature. They respectively captured 60 and 120 action categories, including daily actions, medical situations, and human-human interactions. Both versions are provided with RGB, depth, 3D skeletons, and IR data, having almost 115K samples for the updated version.

Some datasets are acquired under more varied settings to better reflect the conditions that are likely to occur in real-world applications. Multi-view human activity datasets were obtained from various camera viewpoints by simultaneously using several cameras or changing the camera viewpoints in a different trial [21], [22], [45], [46]. UESTC [46] considered human-robot interaction (HRI) applications for action recognition from arbitrary viewpoints. The dataset includes eight fixed viewpoints with arbitrary viewpoints sampled from the entire 360° horizontal directions. There also exist other datasets that target action recognition for specific applications, such as eldercare [24], [25]. ETRI-Activity3D [24] captured elders' activities of daily living (ADL) from several viewpoints considering mobile robots' heights for care robotic services. Toyota Smarthome [25] is another dataset on elders' ADL that possesses severe class imbalance and intraclass variation by capturing unscripted daily activity videos of the elderly.

C. SYNTHETIC DATA EXPLOITATION

To provide abundant training data for deep learning methods to avoid overfitting, some studies focused on utilizing synthetic data. Synthetic data generation is considered cost-effective and customizable since users can manipulate data reflecting one's needs without any additional data capturing middleware or subjects. Some studies generate synthetic data using generative adversarial networks (GANs) [47], [48] or composite methods based on existing real data [49], [50]. Another group of studies uses computer graphics and game engine techniques to simulate data and exploit them for deep learning tasks [27]–[29], [51], [52].

[48] uses two adversarial generative networks to train instance-level pairwise cross-view connection knowledge and performs robust action recognition with additional training data generated for deficient views. [49] composites realistic images to overcome the laborious manual labeling process for the 3D skeleton, depth, and motion. The human motion is extracted based on motion capture (MoCap) recordings, randomized textures, viewpoints, and lighting conditions are added on top of a static real-world background image to generate data; such data are applied to human depth estimation and human part segmentation tasks. The subsequent study [50] extracts 3D human dynamics using a 3D human shape estimation method and synthesizes other randomized components to render complementary training data to improve the action recognition from unseen viewpoints.

Here, we focus on game engine techniques to synthesize data without any reference RGB data and generate realistic videos by considering various factors, including the context of the background, physics, and object interaction. Various game engine-based data generation studies were conducted in fields where data acquisition in various environments is highly expensive, such as autonomous systems [51], [52] and robotics [27], [28]. For human action recognition, Souza et al. [29] initially generated abundant synthetic training data under a variety of conditions with the Unity® game engine and enhanced action recognition performance by training with a mixture of real-world and generated data. However, most activity categories are not indoor activities of daily living hence not applicable to train models for eldercare applications. [53] introduced a simulation platform, developed in Unreal

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3051842, IEEE Access

Engine 4® (UE4), to procedurally produce photorealistic synthetic videos of household activities in various modalities, but fails to provide details of the synthetic data augmentation effect in action recognition. [52] developed a simulation framework to automatically generate annotated training data from the Unity® game engine. They show outstanding action recognition accuracy in classifying five activities by training a shallow skeleton-based action recognition algorithm with their generated data. In this paper, we further explore the benefits of training synthetic data based on three state-of-the-art deep action recognition algorithms (fed with different data modalities containing RGB, 2D, and 3D skeleton) to classify 55 action classes.

III. ELDERSIM DEVELOPMENT

We now elaborate on how our elders' activity simulation platform, denoted as ElderSim, has been developed in detail. In the development, we focus on the following two aspects: 1) to visualize the virtual environment as close as possible to the real-world and 2) to reflect various situations that can be captured in actual applications. To fulfill our first aim, we utilize a real-time photorealistic rendering platform Unreal Engine 4® (UE4) and a three-dimensional (3D) computer animation and modeling software called Autodesk Maya® (Maya). Using the two software, we construct the simulation environment of elders' daily living that resembles the real household backgrounds. We then model appearances and movements of synthetic characters based on the motion capture (MoCap) data obtained from the elders. To achieve the second objective, we consider 55 activity classes that sufficiently include the most frequent ADL of the elders. We also make it available to customize various camera viewpoints and lighting conditions, regarding care robot and smart surveillance applications. The following sections explain further development details and distinctive features of ElderSim. We then introduce KIST SynADL, a large-scale synthetic dataset generated from ElderSim.

A. BACKGROUND

To provide realistic simulation backgrounds for elders' daily living in ElderSim, we have modeled four residential houses based on their indoor measurements and photographs. House models can be added if necessary. When implementing the house models in ElderSim, the household background has become visually more realistic by using physics-based materials and the Post-Process Volume function in UE4. Each of the four house models contains four areas (living room, bedroom, kitchen, and bathroom) as shown in Fig. 2. In each area, we only simulate activities that are plausible to be performed (e.g., *wash face* is simulated only in the bathroom while *play with a mobile phone* is simulated in all four areas).

B. CHARACTER

We have modeled synthetic characters that imitate thirteen elder subjects (seven females and six males with average age and standard deviation as 69.92 and 3.36, respectively)

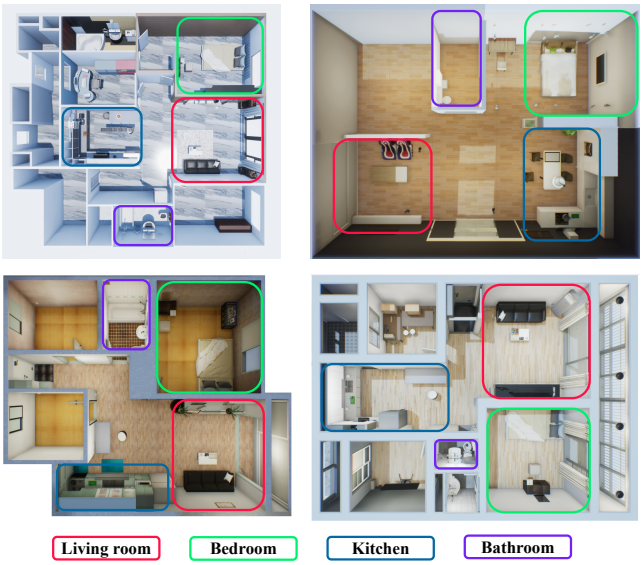


FIGURE 2. The top view of four different residential household backgrounds implemented in ElderSim. Each household consists of four areas where daily activities are frequently occurred.

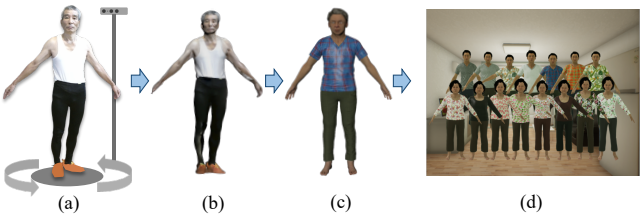


FIGURE 3. Body shapes are captured from a depth sensor in the real-world (a). Depth information is used to model the synthetic character (b) and appropriate textures are applied (c) to reflect the various appearances (d).

and two relatively young subjects (a female and a male) in ElderSim. These subjects have been recruited to sufficiently represent a variety of body shapes and appearances. Their body shapes have been captured from Kinect depth sensors and utilized to design the body shape of synthetic characters in Maya. The faces of characters have been randomly created due to legal issues on portrait rights. In addition, different age-appropriate clothes have been applied to each character to enhance their appearance diversity. As a result, ElderSim can generate action data from fifteen synthetic characters possessing individual face, body shape, and appearance (see Fig. 3). The number of synthetic characters can be increased by implementing body shape transformation techniques in computer graphics, which are left for future work.

C. MOTION

Following [24], we provide motions for 55 activity classes considered to be the most frequent ADL of the elders in ElderSim. To generate realistic motions for these activities, we utilize MoCap data obtained from the subjects recruited in Section III-B. Sixteen digital MoCap cameras have captured the subject motions using 40 markers attached to the subjects' body. When acquiring data, there have not been any specific

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3051842, IEEE Access

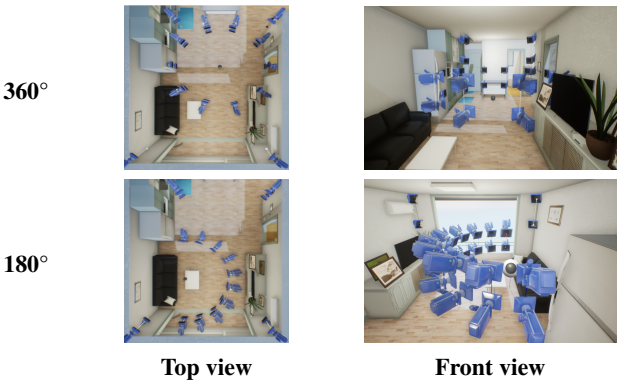


FIGURE 4. An example of the virtual camera setups based on the available viewpoints. Top and front views are chosen for a better interpretation. Here, 20 cameras represent robot-viewpoints and eight cameras on each corner of the room represent surveillance-viewpoints. The user can easily manage the number of viewpoints by adjusting the angle interval and a set of heights.

FIGURE 5. Representative lighting conditions modeled in ElderSim (clockwise from top left: *dawn*, *noon*, *night*, and *sunset*). Controllable lighting conditions appropriately reflect the real-world with the usage of indoor light sources and the Post-Process Volume effect.

instructions for subjects to perform ADL to increase realism and diversity of motions. The obtained MoCap data have been rigged in Maya, i.e., skeletal templates and their movements that best fit the data are constructed. From the rigged data, the motions for synthetic characters are generated by adjusting the template's kinematic parameters to those of each character and playing the constructed movements. To provide motion data in 3D skeleton modality, we define skeleton joints by attaching the sockets of UE4 to each character's 25 joints following two types of joint format labels used in OpenPose [40] and Kinect v2. Two-dimensional (2D) joint motions are obtained by projecting those in 3D to the image plane of each camera viewpoint using a transformation function in UE4.

D. VIEWPOINT

Camera viewpoints in ElderSim contain robot- and surveillance-viewpoints, considering eldercare applications. Robot-viewpoints simulate video acquisition from care robots, and corresponding cameras are located on a circle to surround a target character with the circle radius appropriately defined from the range of the character's motion. The cameras have equal spacing on the circle with an angle interval ϕ and located at heights specified by a set of h height values $\delta = \{\delta_1, \delta_2, ..., \delta_h\}$, where δ_i denotes the i-th height value. The angle interval ϕ and the set of height values δ are set to be user-adjustable parameters. Given these parameter values, the number of viewpoints (V_{circle}) can be expressed as

$$V_{circle} = h \times floor(360^\circ/\phi). \quad (1)$$

Such a circular camera layout may not be available occasionally due to obstacles in some backgrounds (e.g., when the character is sitting on a sofa, a wall behind the sofa hinders the rear robot-viewpoints); we then form viewpoints to cover a semicircle instead of a circle (see Fig. 4). In this semicircular camera layout, the number of viewpoints is given as

$$V_{semicircle} = h \times (floor(180^\circ/\phi) + 1). \quad (2)$$

To implement these camera layouts for robot-viewpoints in ElderSim, we define UE4 splines that contain multiple cameras vertically and position these splines according to the parameter settings. Meanwhile, surveillance-viewpoints simulate video acquisition from surveillance cameras such as closed-circuit televisions (CCTV). They are located at the height of $1.5\,\mathrm{m}$ and $2.2\,\mathrm{m}$ in four corners of each area to reflect the realistic camera installation, hence resulting in eight surveillance-viewpoints.

E. LIGHTING

Lighting conditions in ElderSim are affected by both sunlight and indoor light sources modeled in UE4. To simulate the effect of sunlight over time, we utilize the SkySphere Blueprint function of UE4 and provide an adjustable time parameter in 100 levels to vary sunlight. Indoor light sources are placed according to lighting layouts of actual houses considered in Section III-A and controlled to resemble our daily life better, e.g., turned off during the daytime and turned on during the evening as shown in Fig. 5. Rendering effects, which are significantly affected by lighting conditions, become finer by applying the Post-Process Volume effect of UE4.

F. OBJECT

Among the 55 activity classes considered in ElderSim, 35 activities contain human-object interaction. We model objects that are required to simulate these activities in UE4. The types of objects range from a single rigid body (for 28 classes) such as a cup to articulated objects such as a vacuum cleaner (for the *vacuum the floor* class) or even deformable objects such as a jacket (for the *take off jacket* class). All the objects are modeled in three different ways to increase diversity. When objects are used in ElderSim, they are attached to the contacting body parts' mesh and move along with the body parts to look natural.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3051842, IEEE Access

FIGURE 6. A snapshot of ElderSim showing the user interface to generate action data (a *male elder* character performing *hang out laundry*). The intuitive user interface allows the user to generate customized data by providing several data-generating options.

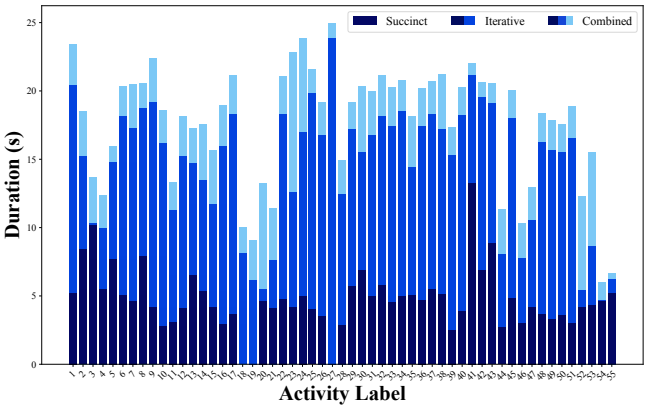


FIGURE 7. We provide three duration types in ElderSim to enhance applicability. The average video duration for each activity class provided in ElderSim is represented in different colors based on the duration types. Actions with a *Succinct* duration type contain a single trial of an action. *Iterative* duration-typed actions are performed repeatedly, but in a different way. Trivial motions are added to the *Iterative* duration type to form a *Combined* duration.

G. USER INTERFACE

An intuitive graphical user interface (GUI) is provided in ElderSim to select data-generating options as needed. The user can easily choose the desired subset of activities, characters, and backgrounds from the provided sets in order (see Fig. 6). The camera viewpoints can then be selected among the robot- and surveillance-viewpoints, while preferable robot-viewpoints are adjusted by an angle interval ϕ and a set of heights δ as mentioned in Section III-D. The lighting conditions are determined by choosing a subset of the hundred time-levels to vary sunlight, from 0 to 1. For the activities containing object interactions, the user can choose whether to use an object and which object model to use. In addition, for the activities containing repetitive motion, we provide three different types of motion duration to include one iteration (succinct), multiple iterations (iterative), and sequential movements (combined); the average motion duration for each activity class in ElderSim is illustrated in Fig. 7. Once the data-generating options are determined, the data are automatically generated and recorded according to all possible combinations of options in ElderSim. ElderSim provides adjustable video resolutions and frame rates of up to 1920×1080 and 60 frames per second (FPS), respectively. Furthermore, three kinds of output data modalities are provided: RGB video, 2D, and 3D skeleton. For skeleton data, we provide both OpenPose- and Kinect v2-based skeletal formats. Parallel processing allows faster data generation.

H. KIST SYNADL

Based on the developmental features of ElderSim, we generate KIST SynADL, a large-scale synthetic dataset of elders' daily activities considering care robot and smart surveillance applications. All 55 activities, 15 characters, and four backgrounds modeled in ElderSim are utilized to generate KIST SynADL. We further customize parameters for camera viewpoints as follows. To provide robot-viewpoints, we set the angle interval and height parameters (introduced in Section III-D) to $\{\phi = 36°, \delta = (0.7\,\mathrm{m}, 1.2\,\mathrm{m})\}$ for semicircular camera layout respectively, where the height parameters are set based on several real-world care robots. These parameter settings result in ten horizontal camera locations, each having two different heights, thus providing 20 robot-viewpoints. Including eight more surveillance-viewpoints introduced in Section III-D, KIST SynADL contains 28 camera viewpoints. For the lighting conditions, we divide a day into five parts by setting the time parameter to 0 (*dawn*), 0.25 (*noon*), 0.5 (*evening*), 0.75 (*sunset*), and 1 (*night*), with *noon* being the default lighting condition.

In the case of activities involving human-object interactions, we utilize only one kind of object model for each activity to generate data. RGB videos in the KIST SynADL dataset are recorded with a 640×360 resolution at 20 FPS, and corresponding 2D and 3D skeleton data are saved in both OpenPose- and Kinect v2-based formats. As a result, KIST SynADL provides 462k RGB videos, 2D, and 3D skeleton data, covering 55 action classes, 28 camera viewpoints, 15 characters, five lighting conditions, and four backgrounds.

IV. EXPERIMENTS

In this section, we experimentally validate and discuss the effect of augmenting our synthetic data, KIST SynADL (KIST), to train the models to recognize elders' ADL. We begin by introducing four real-world datasets for the experiments and address how insufficient the existing public dataset (NTU RGB+D 120) is to cover the elders' ADL. We then describe three state-of-the-art HAR methods used in the experiments as well as several experimental scenarios to examine the various aspects arising from the recognition of the elders' ADL. Within each experimental scenario, we investigate how our synthetic data can help recognize elders' daily activities and offer some guidance and insights for effective synthetic data utilization.

A. DATASETS

We now introduce real datasets used in the experiments and explain how their activity classes are selected to match the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3051842, IEEE Access

FIGURE 8. Sample RGB snapshots and 2D skeleton coordinates of the datasets used in the experiments. The border color of each sample indicates both data type and age group (purple: synthetic-elderly data, pink: real-elderly data, cyan: real-young data).

elders' ADL. Samples of the datasets are visualized in Fig. 8.

1) ETRI-Activity3D Dataset

As introduced in Section II, the ETRI-Activity3D (ETRI) [24] dataset has been recently released for care robot applications to recognize the ADL of the elders'. This dataset contains 55 activity classes performed by a hundred subjects (composed of 50 elder and 50 young subjects) and captured from eight robot-viewpoints (installed in four locations each having two different heights) under constant lighting conditions. Since the dataset contains a large number of data with 112,564 samples, we utilize the ETRI dataset as our primary real-world dataset to train and evaluate three HAR models on elders' ADL. However, due to the limited lighting conditions, the model trained only on this dataset may fail to perform well on other lighting conditions that might appear in the real-world. Further details are addressed in the cross-lighting split defined in Section IV-B.

2) KIST Living Lab Activity Dataset

KIST Living Lab Activity (LIVA) dataset is a newly acquired real-world elder activity dataset in this work to include different lighting conditions. It is composed of 23 activities performed by ten elder subjects (average age of 67.6) in a laboratory designed to imitate a household environment. Here, the selected 23 activity classes correspond to the intersection of those in ETRI and NTU RGB+D 120 datasets as shown in Table 1. 5,520 video sequences are captured from four to eight camera viewpoints considering care robot and smart surveillance applications under three lighting conditions (*bright*, *dim*, and *dark*). We mainly utilize the LIVA dataset as a test set in the cross-lighting split.

3) NTU RGB+D 120 Dataset

The NTU RGB+D 120 [22] dataset consists of 120 kinds of activities performed by relatively young subjects in a laboratory environment. This dataset includes multiple trials of actions captured from five camera viewpoints installed at a constant height and under a constant lighting condition. Among 120 activities of [22], we construct the NTU dataset by selecting only 25 activities with 23,436 samples that match the 55 frequent ADL considered in the ETRI dataset. Such activities of the NTU dataset are mapped to the 23 activities of the ETRI dataset as shown in Table 1. Even though [22] is one of the most widely investigated datasets in the HAR literature due to its large scale and diverse activity classes, there is still a severe gap in the subjects' ages, backgrounds, and activity classes compared to the elders' ADL held in households. Therefore models trained on this dataset may not generalize well to the elders' ADL; we discuss such a point in the cross-dataset split with further details in Section IV-B.

4) Toyota Smarthome Dataset

The Toyota Smarthome [25] dataset consists of 31 kinds of activities performed by eighteen elder subjects in an apartment. This dataset is collected from Kinect v1 at seven surveillance viewpoints under a constant lighting condition.

TABLE 1. Combined activity categories of the KIST, ETRI, LIVA, NTU, and TOYS datasets for the cross-dataset and cross-lighting splits. In some cases, more than two classes from NTU and TOYS datasets are merged to match a single class label in other datasets.

	Dataset			
Combined Label	**KIST (Ours), ETRI [24]**	**LIVA (Ours)**	**NTU [22]**	**TOYS [25]**
1. eat	eat food with a fork	eat food with a fork	eat meal	eat (at table+snack)
2. drink	drink water	drink water	drink water	drink (from cup+from bottle +from can+from glass)
3. brush teeth	brush teeth	brush teeth	brush teeth	-
4. wash hands	wash hands	wash hands	rub two hands	-
5. brush hair	brush hair	brush hair	brush hair	-
6. wear clothes	put on jacket	wear jacket	put on jacket	-
7. take off clothes	take off jacket	take off jacket	take off jacket	-
8. put on/take off shoes	put on/take off shoes	put on/take off shoes	put on a shoe+take off a shoe	-
9. put on/take off glasses	put on/take off glasses	put on/take off glasses	put on glasses+take off glasses	-
10. read	read a book	read a book	read	read a book
11. write	handwrite	write	write	-
12. phone call	talk on the phone	phone call	phone call	use telephone
13. play with phone	play with a mobile phone	play with phone	play with phone/tablet	use tablet
14. use computer	use a computer	use a laptop	type on a keyboard	use laptop
15. clap	clap	clap	clap	-
16. rub face	rub face with hands	rub face with hands	wipe face	-
17. bow	take a bow	bow	nod head/bow	-
18. handshake	handshake	handshake	shake hands	-
19. hug	hug each other	hug	hug	-
20. fight	fight each other	fight	punch/slap	-
21. hand wave	wave a hand	wave hand	hand wave	-
22. point finger	point with a finger	point a finger	point to something	-
23. fall down	fallen on the floor	fall down	fall down	-

Among 31 activity classes, we select ten activities included in the NTU dataset as shown in Table 1 to form the TOYS dataset. Due to a relatively smaller number of action classes and smaller dataset size than ETRI, we find that this dataset alone may not be sufficient to be used as a training dataset to recognize elders' ADL. We utilize the combined six activities of the TOYS dataset as a test set in the cross-dataset split.

B. EXPERIMENTAL SCENARIOS

We now introduce several experimental scenarios considered in the experiments. We begin by explaining cross-subject and cross-view splits which are widely considered in the literature, and then introduce newly suggested cross-age, cross-dataset, and cross-lighting splits that assume the real and synthetic training datasets of different configurations.

In the **cross-subject** or **cross-view** splits, it is assumed that the real-world training dataset has limitations on available camera subjects or viewpoints. We train models using data acquired from only a part of the available subjects or viewpoints of a dataset and leave the remaining data as the test set. In the cross-subject split, we train the models on data for 24 subjects of the ETRI dataset and test on the other 76 subjects. In our cross-view split, we train on data for the two viewpoints (the seventh and eighth viewpoints of [24]) of the ETRI dataset and test on the other six viewpoints.

As an extension of the cross-subject split, we assume the situation in which the training data are only limited to one age group while having the other age group for evaluation. In such **cross-age** splits, we divide the ETRI dataset into two subject groups of different ages, 50 younger subjects with the average age of 23.6 (ETRI_Y) and 50 elder subjects with the average age of 77.1 (ETRI_E). We then train the models on ETRI_Y and test on ETRI_E and vice versa. From such a split, we investigate if there are some differences according to the age groups in the recognition performance as well as the effect of utilizing our synthetic data.

We further assume an extreme scenario of training a model to recognize ADL of the elders, while available data are far from those in several aspects. For example, data may be obtained only from young subjects in laboratory environments (e.g., the NTU dataset). In the **cross-dataset** split assuming this scenario, we train models on the NTU dataset and test on the ETRI, TOYS, and LIVA datasets, all of which correspond to the dataset on elder's ADL. We then examine whether the models trained on the NTU dataset can be generalized well to the others and see if augmenting our synthetic data during training can help the generalization. Since the class composition of the datasets does not completely match each other, we define 23 combined classes based on the ETRI dataset and consider only 25 activity classes out of 120 for the NTU dataset and ten out of 31 for the TOYS dataset to match the classes as explained in Section IV-A (see Table 1).

On the other hand, considering recognition models applied in real household environments, we do not know the lighting condition in advance in which the recognition should be performed (or from which the test data would be obtained). In our newly proposed **cross-lighting** split, we assume that the recognition model is trained from data of limited lighting conditions and tested on other lighting conditions. To simulate this scenario, we train models on the ETRI dataset, which is acquired in a constant lighting condition, and test them on the LIVA dataset which contains various lighting conditions.

TABLE 2. The experimental scenarios and the factors that differ between training and test datasets for each split.

Variation Factor	Dataset	Subject	View	Age	Lighting
Cross-Subject	✗	✓	✗	✗	✗
Cross-View	✗	✗	✓	✗	✗
Cross-Age	✗	✓	✗	✓	✗
Cross-Dataset	✓	✓	✓	✓	✗
Cross-Lighting	✓	✓	✓	✗	✓

We then discuss the effect of training our synthetic KIST SynADL dataset for each scenario. From the baseline models trained without the KIST SynADL dataset, we investigate how action recognition performance varies when our synthetic dataset is augmented in the model training process. For the two widely used cross-subject and cross-view scenarios, we also examine the performance of fine-tuning the models pre-trained on the KIST SynADL dataset. Furthermore, we vary the composition of the data used among the KIST SynADL dataset according to the scenarios. For example, since the ETRI dataset does not contain surveillance-viewpoints and diverse lighting conditions, we may use only robot-viewpoints and a default lighting condition of the KIST SynADL dataset when it is known to be tested on the ETRI dataset. For the later experiments, we use the abbreviation KIST and KIST_5 for the KIST SynADL dataset containing only the default lighting condition and all of the five lighting conditions, respectively. The experimental scenarios performed in this work is listed in Table 2 with the variation factors that differ between training and test datasets.

C. TRAINING DETAILS OF HAR METHODS

This section provides training details for the HAR methods utilized in the experiments, namely Glimpse Clouds (Glimpse) [11], Spatial Temporal Graph Convolution Network (ST-GCN) [12], and View Adaptive Convolutional Neural Network (VA-CNN) [13].

Glimpse [11] is an RGB-based model that uses a visual attention module over the spatio-temporal cube to generate a cloud of glimpse windows. These windows are then soft-assigned to a set of gated recurrent units (GRUs) [54] that track the windows and process classification. A loss function to appropriately locate the windows is added to the original cross-entropy loss. Here, we follow [11] and utilize the 2D skeleton data corresponding to the RGB data to encourage the training process with another loss term that helps the model to perform pose regression. The Adam optimizer is used in training with an initial learning rate of 1e-4. Training the whole model took 13 hours for ten epochs with a minibatch size of 32 using a single NVIDIA Tesla V100 PCIe GPU. During test time, only RGB data resized to a 256×256 resolution is used as an input. We sample eight frames from a video sequence as in [50] and extract three windows per frame as inputs for the recurrent units.

ST-GCN [12] represents 2D or 3D skeleton joint trajectories as a graph that connects nearby joints in a single frame and identical joints between consecutive frames. It then applies spatial temporal graph convolution on the constructed graph and captures the interaction between nearby joint groups and the temporal motions to facilitate action recognition. In this experiment, we apply ST-GCN on the 2D skeleton data. The 2D skeleton data of the real datasets are estimated from RGB videos using OpenPose, and pixel coordinates with the estimation confidence values of each joint are used as an input. We use the stochastic gradient descent to train ST-GCN models with batch size 64 for 50 epochs. The learning rates start at 0.1 and are reduced by 10 in epochs 20, 30, and 40. Moreover, when synthetic data is augmented in training, we split the last fully connected layer of the model so that the real and synthetic data can pass through different classifiers (except for the cross-dataset split). In this way the model empirically shows slightly better performance.

VA-CNN [14] represents 3D skeleton joint trajectories as a planar image by mapping joint index and time axes to height and width axes respectively, and recognizes the image using a convolutional neural network (CNN). The most distinctive feature of the method is that it adapts the input data view to enhance the recognition performance with a view adaptation subnetwork. The subnetwork used to adapt the view is also modeled using a CNN, and the whole model is trained in an end-to-end fashion. We utilize Adam optimizer to train VA-CNN with batch size 64 for 30 epochs. The learning rate starts at 1e-4 and reduces by 10 for every ten epochs. We use the Kinect v2-based format for the KIST SynADL dataset to match real datasets.

For the experiments augmenting the KIST SynADL dataset to train ST-GCN and VA-CNN, we balance mini-batches to contain an equal amount of real and synthetic data. Since the sizes of datasets differ, we randomly upsample the dataset of a smaller amount (usually the real-world data) to match the size. Twenty viewpoints of the KIST SynADL dataset were utilized for both methods, while only eight viewpoints were used for the Glimpse method to ensure reasonable training time.

D. EXPERIMENTAL RESULTS

We now report the results of the experiments performed according to the above settings. In the experiments, we trained three recognition algorithms for the proposed experimental splits and report the average video sequence-level top-1 classification accuracy for the five test trials as the action recognition score. For the results obtained from augmenting the KIST SynADL dataset, we designate the change in the recognition score from that obtained without augmentation in the parenthesis next to the score.

1) Cross-Subject

In the cross-subject split, 24 subjects (26,612 samples) from the ETRI dataset are sampled for the training set and eval-

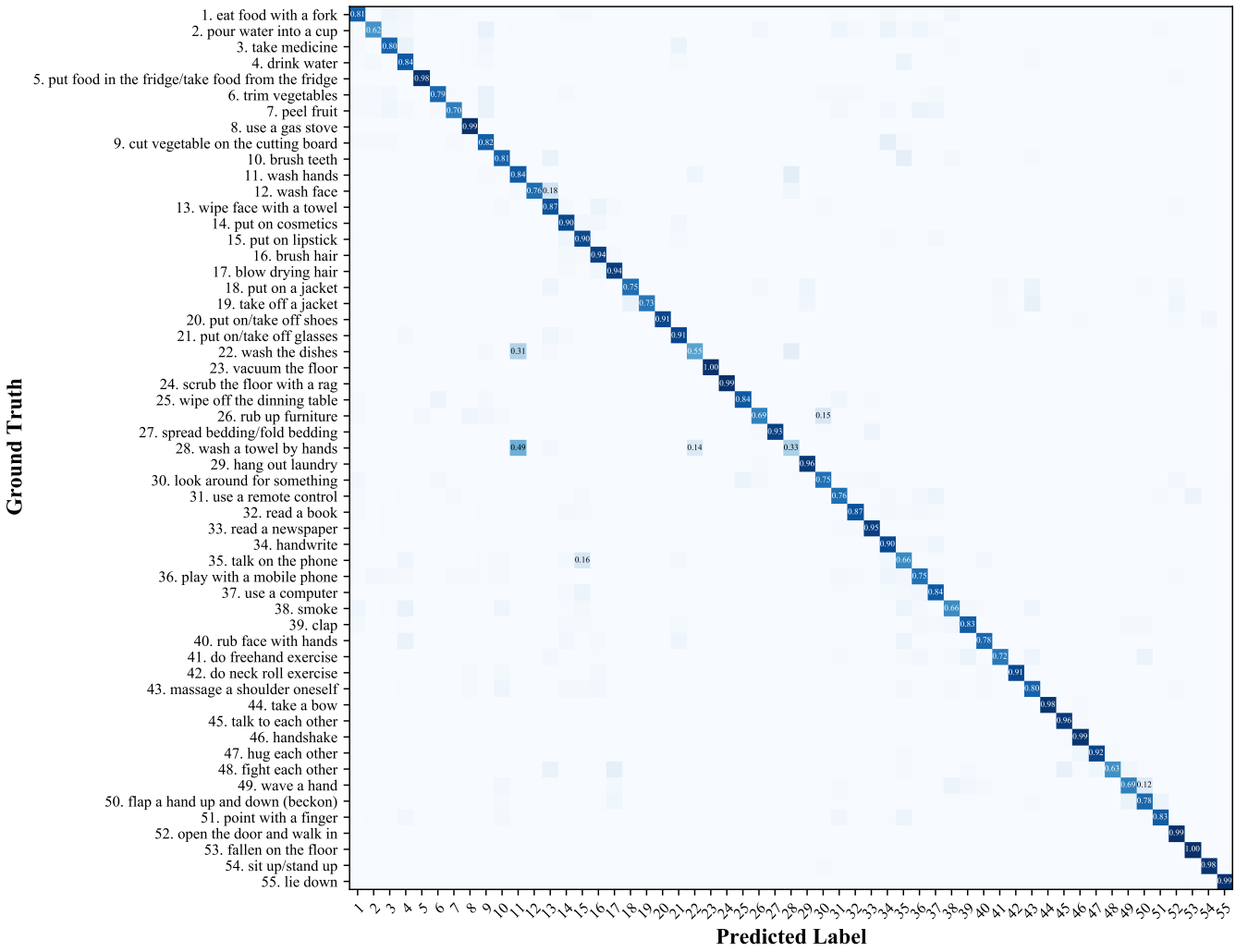


FIGURE 9. Normalized confusion matrix of the Glimpse method trained by augmenting synthetic data in the cross-subject split. Only the values over 0.1 are displayed for better visualization.

TABLE 3. Accuracy comparison for the cross-subject split.

Setting		Top-1 Accuracy (%)		
Train	Test	Glimpse [11]	ST-GCN [12]	VA-CNN [13]
ETRI	ETRI	80.22	83.36	81.98
ETRI+KIST	ETRI	**83.53 (+3.31)**	**84.04 (+0.68)**	**82.20 (+0.22)**

TABLE 4. Accuracy comparison for the cross-view split.

Setting		Top-1 Accuracy (%)		
Train	Test	Glimpse [11]	ST-GCN [12]	VA-CNN [13]
ETRI	ETRI	79.97	77.88	79.72
ETRI+KIST	ETRI	**81.59 (+1.62)**	**80.84 (+2.96)**	**80.00 (+0.28)**

uated by the remaining 76 subjects (85,912 samples) as explained in Section IV-B.

By augmenting synthetic data (26,400 samples for Glimpse and 66,000 samples for ST-GCN and VA-CNN) in training, each method's performance increases by 3.31, 0.68, and 0.22 percent points as described in Table 3. While Glimpse shows the largest improvement, the absolute classification accuracy score is still lower than ST-GCN, showing confusion within some classes (e.g., data from *wash a towel by hands* class was frequently misclassified as *wash hands* class) as illustrated in Fig. 9.

One should note that ST-GCN outperforms other methods in this cross-subject split of the ETRI dataset, while it performs worse than other considered methods on the NTU RGB+D [21] cross-subject split in the corresponding literature [11]–[13].

2) Cross-View

For the cross-view split, the data obtained from two camera viewpoints are used for training (26,757 samples) while the remaining six viewpoints (85,807 samples) are used for evaluation of the trained models. The cross-view results also demonstrate the benefit of training additional synthetic data for better performance as shown in Table 4.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3051842, IEEE Access

TABLE 5. Accuracy comparison between the synthetic data augmentation training and fine-tuning for the cross-subject (C-S) and cross-view (C-V) splits.

Setting			Top-1 Accuracy (%)		
Split	Train	Test	Glimpse [11]	ST-GCN [12]	VA-CNN [13]
C-S	ETRI	ETRI	80.22	83.36	81.98
	ETRI+KIST	ETRI	83.53	**84.04**	**82.20**
	KIST→ETRI	ETRI	**83.57**	83.47	75.07
C-V	ETRI	ETRI	79.97	77.88	79.72
	ETRI+KIST	ETRI	81.59	**80.84**	**80.00**
	KIST→ETRI	ETRI	**83.77**	80.38	71.60

While ST-GCN shows the largest increase, Glimpse outperforms other methods, unlike the results of the baseline studies [11]–[13]. This observation implies that the accuracy rank among the three methods evaluated on a benchmark dataset is not entirely consistent with that evaluated on other datasets (e.g., ETRI).

3) Comparing Synthetic Data Augmentation Training and Fine-Tuning

We further examine the action recognition performance of fine-tuning the three models pre-trained with synthetic data for the two widely used cross-subject (C-S) and cross-view (C-V) splits. In pre-training, the models' weights are trained only with synthetic data hoping that some meaningful features for the recognition are learned from the large amount of data. We then fine-tune the model on real-world data to allow the model to better adapt to the unseen real-world data. We compare the recognition performance of this fine-tuning approach with the previous synthetic data augmentation training in Table 5. The notation KIST→ETRI in Table 5 denotes the fine-tuning approach that pre-trains the models with the synthetic data (KIST) and then fine-tunes the models on the real-world data (ETRI). For the Glimpse method, the fine-tuning approach surpasses the data augmentation approach by 2.18 percent point for the cross-view split. For the ST-GCN method, the fine-tuning approach shows superior performance to the model trained without synthetic data, but shows inferior performance to the model trained by augmenting synthetic data. For the VA-CNN method, the performance from the fine-tuning approach is rather reduced compared with the model trained without synthetic data. One possible explanation for this result is that pre-training only with the synthetic data may have caused overfitting for the VA-CNN method since the synthetic 3D skeleton data have been essentially generated from a relatively small set of MoCap data and may possess limited variations in the data on its own. Since the results from the fine-tuning approach are relatively inconsistent among different algorithms than the synthetic data augmentation results, we focus on the synthetic data augmentation approach for the following experimental splits.

TABLE 6. Accuracy comparison for the cross-age split.

Setting		Top-1 Accuracy (%)		
Train	Test	Glimpse [11]	ST-GCN [12]	VA-CNN [13]
ETRI_E	ETRI_Y	**74.96**	77.52	77.52
ETRI_E+KIST	ETRI_Y	73.90 (-1.06)	**78.12 (+0.60)**	**78.00 (+0.48)**
ETRI_Y	ETRI_E	75.35	79.32	78.06
ETRI_Y+KIST	ETRI_E	**77.74 (+2.41)**	**80.38 (+1.06)**	**78.18 (+0.12)**

4) Cross-Age

In the cross-age split, we construct the training and test data from the ETRI dataset by splitting the data according to the subjects' age as explained in Section IV-B, and examine if augmenting our KIST SynADL dataset in training affects the recognition performance differently according to the age group. The action recognition performances for the cross-age split experiments are shown in Table 6.

Similarly to the previous results, synthetic data augmentation enhances recognition performance in most of the cases. By focusing on the performance change (the values placed in parentheses) induced from augmenting synthetic data, we observe that our synthetic data affect the recognition performance in a somewhat age-specific way, i.e., the augmentation seems more beneficial for the models trained on ETRI_Y (and tested on ETRI_E) rather than those trained on ETRI_E (and tested on ETRI_Y). This effect is the most evident for the Glimpse method, which has the highest performance gain among the considered methods for the models trained on ETRI_Y and even shows a performance decrease for the models trained on ETRI_E. Another interesting point to note is that, as can be observed in Table 6, the actions in ETRI_Y seem to be more challenging to classify than the actions in ETRI_E when the models are trained on the other data. This tendency agrees with the observation that the actions of the young subjects usually have larger motion differentials and shorter durations than the motions performed by the elders hence contain a wider variety [24]. Comparing the three HAR methods, ST-GCN outperforms other methods in the cross-age split.

5) Cross-Dataset

From the cross-dataset split, as explained in Section IV-B, we examine whether a model trained on the NTU dataset (data from the young subjects in a laboratory background) can be generalized well to the others (the ETRI, TOYS, and LIVA datasets obtained from the elder subjects in daily-living environments) as well as the effect of augmenting our synthetic data during training. Tables 7 and 8 show the results from the cross-dataset split tested on the ETRI dataset and the TOYS and LIVA datasets, respectively.

The recognition performances for the cross-dataset split are lower than the results obtained from the former splits, in which the (real) training and test data come from a common

[1]Since the lighting condition does not affect the skeleton data, the results obtained from augmenting KIST and KIST_5 for ST-GCN are identical.

TABLE 7. Accuracy comparison for the cross-dataset split tested on the ETRI dataset.

Setting		Top-1 Accuracy (%)		
Train	Test	Glimpse [11]	ST-GCN [12]	VA-CNN [13]
NTU	ETRI	39.99	46.92	43.00
NTU+KIST	ETRI	**54.79 (+14.80)**	**49.76 (+2.84)**	**46.32 (+3.32)**
NTU	ETRI_E	38.61	45.66	41.30
NTU+KIST	ETRI_E	**55.00 (+16.39)**	**48.46 (+2.80)**	**45.00 (+3.70)**
NTU	ETRI_Y	41.18	48.08	44.58
NTU+KIST	ETRI_Y	**54.62 (+13.44)**	**50.92 (+2.84)**	**47.48 (+2.90)**

TABLE 8. Accuracy comparison for the cross-dataset split tested on the TOYS and LIVA dataset.

Setting		Top-1 Accuracy (%)	
Train	Test	Glimpse [11]	ST-GCN [12]
NTU	TOYS	16.06	29.58
NTU+KIST	TOYS	**35.87 (+19.81)**	**30.91 (+1.33)**
NTU	LIVA	35.94	36.68
NTU+ETRI	LIVA	**46.93 (+10.99)**	**47.16 (+10.48)**
NTU+KIST	LIVA	**52.41 (+16.47)**	**39.62 (+2.94)**
NTU+KIST_5	LIVA	**59.56 (+23.62)**	**39.62 (+2.94)**[1]

dataset. These results imply that, for the eldercare services, it may not be sufficient to utilize deep models trained only on the NTU dataset, despite its large-scale. When synthetic data are augmented in training, we observe a firm performance increase for all the considered HAR methods for the cross-data split. The improvement gap is in general larger than the previous splits, with a remarkable boost for the Glimpse method (even over 13 percent point when tested on the ETRI dataset). Such a considerable increase in the Glimpse method may be partially because meaningful background information contained in RGB videos, which might be helpful to distinguish which activities are performed, is provided to the model from our synthetic data. In contrast, the NTU dataset alone might not provide much information on the backgrounds due to its limited laboratory setting. In Table 7, it is also interesting to observe that the recognition performance tested on ETRI_Y is higher than that on ETRI_E; this may result from the fact that the NTU dataset contains actions of relatively young subjects.

When testing the TOYS and LIVA dataset, we exclude the VA-CNN model since the datasets do not provide skeletal data of the identical structure to the training (NTU) dataset. In Table 8, the action recognition performance on TOYS is lower; this may be partially due to the challenging features of the dataset, such as viewpoints resembling surveillance cameras, as described in [25]. Still, we observe a large performance improvement for the Glimpse method similarly to the results tested on the ETRI dataset.

For the LIVA dataset, we additionally compare the performance of augmenting the (real) ETRI dataset during training, which improves the performance for both Glimpse and ST-GCN methods. Surprisingly, better recognition performance is achieved by augmenting the synthetic KIST dataset with only a single lighting condition rather than the ETRI dataset for the Glimpse method. Even further gain is obtained when using the KIST_5 dataset, which includes all the five lighting conditions. From these results, we can figure out that there exist cases in which synthetic data could be more supportive than real data when augmented in training. Based on the fact that the KIST SynADL dataset contains a household-like background modeled based on where the LIVA dataset is acquired, we surmise that the synthetic data could effectively help the model to generalize well on the LIVA dataset.

TABLE 9. Accuracy comparison between the VA-CNN method and the baseline model trained by augmenting synthetic data (odd rows for the former and even rows for the latter).

Setting				Top-1 Accuracy (%)
Split	Train	Test	VA Module	VA-CNN [13]
Cross-Subject	ETRI	ETRI	✓	81.98
	ETRI+KIST	ETRI	✗	**82.26 (+0.28)**
Cross-View	ETRI	ETRI	✓	**79.72 (+0.04)**
	ETRI+KIST	ETRI	✗	79.68
Cross-Age	ETRI_E	ETRI_Y	✓	**77.52 (+0.10)**
	ETRI_E+KIST	ETRI_Y	✗	77.42
	ETRI_Y	ETRI_E	✓	**78.06 (+0.18)**
	ETRI_Y+KIST	ETRI_E	✗	77.88
Cross-Dataset	NTU	ETRI	✓	43.00
	NTU+KIST	ETRI	✗	**44.94 (+1.94)**
	NTU	ETRI_E	✓	41.30
	NTU+KIST	ETRI_E	✗	**43.70 (+2.40)**
	NTU	ETRI_Y	✓	44.58
	NTU+KIST	ETRI_Y	✗	**46.10 (+1.52)**

6) Comparing VA-CNN and a Simpler Model Trained by Augmenting Synthetic Data

Using the dataset splits considered so far, we now propose a simple experiment to compare the effect of synthetic data augmentation to that of improving HAR neural network models. For ease of comparison, we adopt the VA-CNN model, in which the improvement of the neural network model is represented by implementing the view adaptation subnetwork (or VA module) [13]. Specifically, we compare the recognition performance of the baseline model, i.e., the VA-CNN without the VA module, trained by augmenting synthetic data and the VA-CNN method (the improved model from the baseline). According to the results in Table 9, the recognition performances from both settings are comparable to each other. In the cross-dataset split, augmenting synthetic data is superior to the model improvement. These results indicate that effective utilization of synthetic data during training can be a viable option for better HAR performance, as increasing the complexity of a neural network architecture.

7) Cross-Lighting

In the cross-lighting split, which has not been considered much in the HAR literature, we examine if augmenting our synthetic data can actually help recognize videos acquired under various lighting conditions. Here we only consider the

TABLE 10. Accuracy comparison for the cross-lighting split.

Setting		Top-1 Accuracy (%)
Train	Test	Glimpse [11]
ETRI	LIVA	35.60
ETRI+KIST$_5$	LIVA	**72.39 (+36.79)**
ETRI	LIVA$_{bright}$	40.72
ETRI+KIST$_5$	LIVA$_{bright}$	**75.85 (+35.13)**
ETRI	LIVA$_{dim}$	35.88
ETRI+KIST$_5$	LIVA$_{dim}$	**74.23 (+38.35)**
ETRI	LIVA$_{dark}$	29.61
ETRI+KIST$_5$	LIVA$_{dark}$	**66.85 (+37.24)**

Glimpse method since the lighting matters only for the RGB videos. We train the models on the ETRI dataset acquired under a limited lighting condition (*bright*) and test the models on the LIVA dataset. We further divide the LIVA dataset into LIVA$_{bright}$, LIVA$_{dim}$, and LIVA$_{dark}$, by collecting the data acquired under *bright*, *dim*, and *dark* lighting conditions, respectively; we then test the trained models on each of the datasets.

As shown in Table 10, the recognition performance drops as the lighting condition of the test data gets darker and becomes farther from that of the training data. The recognition performance is remarkably increased (even over 35 percent point) for all the test data by augmenting our KIST$_5$ dataset, which contains five different lighting conditions. These results strongly support that augmenting synthetic data of various lighting conditions can help the models to generalize to the lighting conditions that are not included in real training data.

V. CONCLUSION

Considering eldercare applications, obtaining data of elders' activities of daily living is necessary, but challenging. We take advantage of modern realistic rendering and visualization techniques to develop a platform named ElderSim and simulate a variety of daily activities of the elderly. Based on ElderSim, we generate a large-scale synthetic dataset of elders' activities, KIST SynADL dataset, considering possible applications for care robots and smart surveillance systems. We then demonstrate the effectiveness of augmenting the KIST SynADL dataset in training from extensive experiments involving three state-of-the-art HAR methods as well as four different real-world human activity datasets. We show noticeable improvements of action recognition performance by augmenting our synthetic data. We also offer guidance and insights for the effective utilization of our synthetic data in human action recognition.

In the future, we plan to enlarge the subject diversity by changing the body shape of the elderly and applying the corresponding motion styles to actions. Furthermore, we will extend our ElderSim platform by employing additional features of UnrealCV [55] to apply to more various problems in computer vision and robotics. Designing a domain-adaptive learning framework for HAR to further utilize our synthetic data would be another intriguing area of future research.

ACKNOWLEDGMENT

The authors thank Jaewang Lee for his supportive work with realistic 3D modeling of implemented features in ElderSim. The authors would also like to show their gratitude to Youngjoong Kwon for her initial work with the simulation development and Haetsal Lee for his comments that improved the quality of the manuscript.

REFERENCES

[1] A. Abou Allaban, M.Wang, and T. Padır, "A systematic review of robotics research in support of in-home care for older adults," *Information*, vol. 11, no. 2, p. 75, Feb. 2020.

[2] R. Khosla and M. T. Chu, "Embodying care in Matilda: an affective communication robot for emotional wellbeing of older people in Australian residential care facilities," *ACM Transactions on Management Information Systems (TMIS)*, vol. 4, no. 4, pp. 1-33, Dec. 2013.

[3] A.K. Pandey and R. Gelin. "A mass-produced sociable humanoid robot: Pepper: The first machine of its kind." *IEEE Robotics & Automation Magazine 25*, no. 3, pp. 40-48, Jul. 2018.

[4] M. Yu *et al.*, "A posture recognition-based fall detection system for monitoring an elderly person in a smart home environment." *IEEE transactions on information technology in biomedicine 16*, no. 6, pp. 1274-1286, Aug. 2012.

[5] M. Al-Khafajiy *et al.*, "Remote health monitoring of elderly through wearable sensors." *Multimedia Tools and Applications 78*, no. 17, pp. 24681-24706, Sep. 2019.

[6] L. Yu *et al.*, "Personalized health monitoring system of elderly wellness at the community level in Hong Kong." *IEEE Access 6*, pp. 35558-35567, Jun. 2018.

[7] A. Karpathy *et al.*, "Large-scale video classification with convolutional neural networks," In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1725-1732, 2014.

[8] D. Tran *et al.*, "Learning spatiotemporal features with 3d convolutional networks," In *Proceedings of the IEEE international conference on computer vision (ICCV)*, pp. 4489-4497, 2015.

[9] J. Carreira and A. Zisserman, "Quo vadis, action recognition? a new model and the kinetics dataset," In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6299-6308, 2017.

[10] C. Feichtenhofer *et al.*, "Slowfast networks for video recognition," In *proceedings of the IEEE Conference on Computer Vision (ICCV)*, pp. 6202-6211, 2019.

[11] F. Baradel *et al.*, "Glimpse clouds: Human activity recognition from unstructured feature points," In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 469-478, 2018.

[12] S. Yan, Y. Xiong, and D. Lin, "Spatial temporal graph convolutional networks for skeleton-based action recognition," *arXiv preprint arXiv:1801.07455,* Jan. 2018.

[13] P. Zhang *et al.*, "View adaptive neural networks for high performance skeleton-based human action recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, no. 8, pp. 1963-1978, Jan. 2019.

[14] L. Shi *et al.*, "Skeleton-based action recognition with directed graph neural networks," In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7912-7921, 2019.

[15] F. Li *et al.*, "Multi-stream and Enhanced Spatial-temporal Graph Convolution Network for Skeleton-based Action Recognition," *IEEE Access*, May. 2020.

[16] S. Song *et al.*, "An End-to-End Spatio-Temporal Attention Model for Human Action Recognition from Skeleton Data," *arXiv preprint arXiv:1611.06067*, Nov. 2016.

[17] G. Liu *et al.*, "Action Recognition Based on 3D Skeleton and RGB Frame Fusion," In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 258-264, Nov. 2019.

[18] K. Hara, H. Kataoka, and Y. Satoh, "Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet?," In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6546-6555, 2018.

[19] W. Kay *et al.*, "The kinetics human action video dataset," *arXiv preprint arXiv:1705.06950*, 2017.

[20] C. Liu *et al.*, "Pku-mmd: A large scale benchmark for continuous multi-modal human action understanding," *arXiv preprint arXiv:1703.07475*, 2017.

[21] A. Shahroudy *et al.*, "Ntu rgb+ d: A large scale dataset for 3d human activity analysis," In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pp. 1010-1019, 2016.

[22] J. Liu *et al.*, "Ntu rgb+ d 120: A large-scale benchmark for 3d human activity understanding," *IEEE transactions on pattern analysis and machine intelligence*, May. 2019.

[23] Q. Kong *et al.*, "MMAct: A Large-Scale Dataset for Cross Modal Human Action Understanding," In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 8658-8667, 2019.

[24] J. Jang *et al.*, "ETRIActivity3D: a large-scale rgb-d dataset for robots to recognize daily activities of the elderly," *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct. 2020.

[25] S. Das *et al.*, "Toyota smarthome: Real-world activities of daily living," In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 833-842, 2019.

[26] A. L. Yuille and C. Liu, "Deep Nets: What have they ever done for Vision?," *arXiv preprint arXiv:1805.04025*, 2018.

[27] X. Puig *et al.*, "Virtualhome: Simulating household activities via programs," In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8494-8502, 2018.

[28] P. Martinez-Gonzalez *et al.*, "UnrealROX: an extremely photorealistic virtual reality environment for robotics simulations and synthetic data generation," *arXiv preprint arXiv:1810.06936*, 2018.

[29] C. R. de Souza, *et al.*, "Procedural generation of videos to train deep action recognition networks," In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4757-4767, 2017.

[30] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, vol. 1, pp. 886-893, Jun. 2005.

[31] N. Dalal, B. Triggs, and C. Schmid, "Human detection using oriented histograms of flow and appearance," In *European conference on computer vision (ECCV)*, pp. 428-441, May. 2006.

[32] I. Laptev *et al.*, "Learning realistic human actions from movies," In *2008 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1-8, Jun. 2008.

[33] H. Wang *et al.*, "Action recognition by dense trajectories," In *CVPR*, pp. 3169-3176, Jun. 2011.

[34] H. Wang and C. Schmid, "Action recognition with improved trajectories," In *Proceedings of the IEEE international conference on computer vision*, pp. 3551-3558, 2013.

[35] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," In *Advances in neural information processing systems*, pp. 568-576, 2014.

[36] S. Ji *et al.*, "3D convolutional neural networks for human action recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 1, pp. 221-231, 2012.

[37] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.

[38] Zhang, Z, "Microsoft kinect sensor and its effect," *IEEE multimedia*, vo. 19, no. 2, pp. 4-10, 2012.

[39] J. Shotton *et al.*, "Real-time human pose recognition in parts from single depth images," In *CVPR 2011*, pp. 1297-1304, Jun. 2011.

[40] Z. Cao *et al.*, "Realtime multi-person 2d pose estimation using part affinity fields," In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pp. 7291-7299, 2017.

[41] C. Schuldt, I. Laptev, and B. Caputo, "Recognizing human actions: a local SVM approach," In *Proceedings of the 17th International Conference on Pattern Recognition (ICPR)*, vol. 3, pp. 32-36, Aug. 2004.

[42] W. Li, Z. Zhang, and Z. Liu, "Action recognition based on a bag of 3d points," In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops*, pp. 9-14, Jun. 2010.

[43] J. Wang *et al.*, "Mining actionlet ensemble for action recognition with depth cameras," In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1290-1297, Jun. 2012.

[44] B. Ni, G. Wang, and P. Moulin, "Rgbd-hudaact: A color-depth video database for human daily activity recognition," In *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, pp. 1147-1153, Nov. 2011.

[45] J. Wang *et al.*, "Cross-view action modeling, learning and recognition," In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2649-2656, 2014.

[46] Y. Ji *et al.*, "A large-scale RGB-D database for Arbitrary-view Human Action Recognition," In *Proceedings of the 26th ACM international Conference on Multimedia*, pp. 1510-1518, Oct. 2018.

[47] M. Khodabandeh *et al.*, "Diy human action dataset generation," In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 1448-1458, 2018.

[48] L. Wang *et al.*, "Generative multi-view human action recognition," In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 6212-6221, 2019.

[49] G. Varol *et al.*, "Learning from synthetic humans," In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 109-117, 2017.

[50] G. Varol *et al.*, "Synthetic Humans for Action Recognition from Unseen Viewpoints," *arXiv preprint arXiv:1912.04070*, 2019.

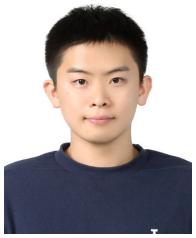
[51] G. Ros *et al.*, "The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes," In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pp. 3234-3243, 2016.

[52] D. Ludl, T. Gulde, and C. Curio, "Enhancing Data-Driven Algorithms for Human Pose Estimation and Action Recognition Through Simulation," *IEEE Transactions on Intelligent Transportation Systems*, 2020.

[53] W. Deneke *et al.*, "Towards a Simulation Platform for Generation of Synthetic Videos for Human Activity Recognition," In *2018 International Conference on Computational Science and Computational Intelligence (CSCI)*, pp. 1234-1237, Dec. 2018.

[54] K. Cho *et al.*, "On the properties of neural machine translation: Encoder-decoder approaches," *arXiv preprint arXiv:1409.1259*, 2014.

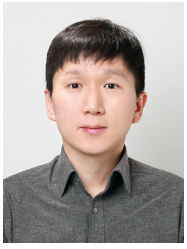
[55] W. Qiu *et al.*, "Unrealcv: Virtual worlds for computer vision," In *Proceedings of the 25th ACM international conference on Multimedia*, pp. 1221-1224, 2017.



HOCHUL HWANG (MEMBER, IEEE) received the B.S. degree in robot engineering from Hanyang University ERICA Campus, Ansan, South Korea, in 2019. He has been working as an Intern Researcher with the Center for Artificial Intelligence, Korea Institute of Science and Technology, Seoul, South Korea, since 2019. His research interest includes machine learning and robotics.

CHEONGJAE JANG received the B.S. and Ph.D. in mechanical and aerospace engineering from Seoul National University in 2012 and 2019, respectively. He is a Research Assistant Professor in the A.I. institute at Hanyang University. His research interests are in machine learning, non-Euclidean data analysis, and robotics. Before joining in Hanyang University, he worked at the Korea Institute of Science and Technology (KIST) as a post-doctoral researcher, from 2019 to 2020.

GEONWOO PARK received the B.S. degree in game engineering from Korea Polytechnic University, Siheung, South Korea, in 2018. From 2018 to 2020, he was a Intern Researcher with the Center for Artificial Intelligence, Korea Institute of Science and Technology, Seoul, South Korea. He has been working as a freelancer with the identical group, since 2020. His research interest includes game programming, virtual tool development, machine learning.



JUNGHYUN CHO (MEMBER, IEEE) received the B.S. degree in industrial design and the M.S. degree in applied mathematics from KAIST, Daejeon, South Korea, in 2002 and 2004, respectively, and the Ph.D. degree in computer graphics from Seoul National University, Seoul, South Korea, in 2013. He has been a Senior Research Scientist with the Center for Artificial Intelligence, Korea Institute of Science and Technology, Seoul, South Korea, since 2014. His research interests include computer graphics, computer vision, and deep learning, especially for domain adaptation.

IG-JAE KIM received the B.S. and M.S. degrees in EE from Yonsei University, Seoul, South Korea, in 1998 and 1996, respectively, and the Ph.D. degree in EECS from Seoul National University, Seoul, in 2009. He was with the Massachusetts Institute of Technology (MIT) Media Laboratory, as a Postdoctoral Researcher, from 2009 to 2010. He is currently the Director-General of Artificial Intelligence & Robotics (AIR) Institute, Korea Institute of Science and Technology (KIST), Seoul. He is also an Associate Professor with the Korea University of Science and Technology. He has published over 90 fully referred papers in the international journal and conferences, including ACM Transaction on Graphics, the IEEE Transaction on Visualization and Computer Graphics, Pattern Recognition, SIGGRAPH, Eurographics, and so on. His research interests include pattern recognition, computer vision and graphics, deep learning, and computational photography.