Exploring the Impact of Rendering Method and Motion Quality on Model Performance when Using Multi-view Synthetic Data for Action Recognition

Stanislav Panev*, Emily Kim*, Sai Abhishek Si Namburu1, Desislava Nikolova2, Celso de Melo3, Fernando De la Torre1, Jessica Hodgins1
1Carnegie Mellon University, 2Technical University of Sofia, 3Army Research Laboratory
{spanev, ekim2}@andrew.cmu.edu, snamburu@alumni.cmu.edu, dnikolova@tu-sofia.bg, celso.m.demelo.civ@army.mil, {ftorre, jkh}@andrew.cmu.edu

Abstract

This paper explores the use of synthetic data in a human action recognition (HAR) task to avoid the challenges of obtaining and labeling real-world datasets. We introduce a new dataset suite comprising five datasets, eleven common human activities, three synchronized camera views (aerial and ground) in three outdoor environments, and three visual domains (real and two synthetic). For the synthetic data, two rendering methods (standard computer graphics and neural rendering) and two sources of human motions (motion capture and video-based motion reconstruction) were employed. We evaluated each dataset type by training popular activity recognition models and comparing the performance on the real test data. Our results show that synthetic data achieve slightly lower accuracy (4–8%) than real data. On the other hand, a model pre-trained on synthetic data and fine-tuned on limited real data surpasses the performance of either domain alone. Standard computer graphics (CG)-rendered data delivers better performance than the data generated from the neural-based rendering method. The results suggest that the quality of the human motions in the training data also affects the test results: motion capture delivers higher test accuracy. Additionally, a model trained on CG aerial view synthetic data exhibits greater robustness against camera viewpoint changes than one trained on real data. See the project page: http://humansensinglab.github.io/REMAG/

1. Introduction

Human action recognition (HAR) from videos is crucial for numerous applications [37]. For instance, video-based human action tracking has proven useful for surveillance and detecting abnormal behaviors [36]. It has also been used for sports, training, and physical therapy [17, 28, 52].

Robust learning for HAR models relies heavily on diverse, large-scale training data. Collecting data is laborious, time-consuming, and error-prone. To solve these issues, re-
searchers have used synthetic data for training [6, 7, 51]. A significant advantage of synthetic data is that it can be scaled quickly, labels can be added effortlessly, and the data can be diversified while keeping their photometric and geometric qualities consistent for data augmentation.

We collected a dataset with video recordings of eleven activity categories captured in the wild with three cameras—one orbiting small UAV and two fixed ground cameras. We also recorded indoor motion capture data and RGB videos of the same activities and reconstructed the human motions in those videos using VIBE [22]. We utilized these two sources of motion data (mocap and VIBE) to create synthetic datasets by employing two different rendering techniques: a 3D computer graphics (CG) engine Blender and a neural human motion imitation generative model Liquid Warping GAN (LWG) [29]. Figure 1 presents samples of the waving gesture across the three camera views from the real and synthetic modalities.

These datasets constitute REMAG—REndering, Motion, Aerial, and Ground view analysis data suite for the HAR task. With it, we sought to answer the following questions:

1. Does the choice of the ML model make a significant difference in performance?
2. Does the synthetic data rendering method affect the model performance?
3. Does improving the quality of the motion in the synthetic training data improve the performance?
4. Can we combine synthetic training data with a limited amount of real training data to improve performance?
5. Can the models trained on one camera view transfer to a novel camera view?

We ran an extensive set of experiments and compared the performance of models trained on different modalities of the data. We used three off-the-shelf activity recognition models—X3D [11] (a single video stream), SlowFast [12] (dual video streams), and MViT [10] (transmitter-based).

The contributions of our paper are as follows:

1) A new real-world video dataset for human action recognition has been collected and annotated, including three synchronized camera views, one aerial dynamic and two ground static views, and eleven everyday activities;

2) Four synthetic counterparts of the real dataset were created by combining two different rendering approaches (CG and neural-based) with two human motion sources (motion capture data and video-extracted motions);

3) An extensive set of experiments has been conducted to answer the five research questions listed above using off-the-shelf activity recognition models. We analyzed how the different rendering techniques, the motion sources, and transferring from one camera view to another affect the models’ performance. To the best of our knowledge, we are the first to perform such an analysis systematically. Our datasets will be made public.

2. Related Work

Research in video action recognition has progressed quickly in the past decade. Several review papers covered in great detail different types of models and datasets for deep video action recognition [25, 47]. Here, we briefly discuss the most popular public training synthetic HAR datasets, the existing real HAR benchmarks, neural renderers for HAR data, and video-based activity recognition deep learning models.

2.1. Synthetic HAR Datasets

Many synthetic video datasets have been created for training HAR deep-learning models in recent years. Synthetic data has already proven to be a helpful solution when large amounts of annotated data should be produced quickly. In Tables 1 and 2, we compare some of the most recent synthetic and real HAR datasets.

The PHAV dataset [8] was created through a procedural workflow based on human action videos parametric generative model. A unique feature of it is the so-called kite camera dynamics model, which resembles the view from a person following the actor. The characters were animated by composite actions assembled from a library of atomic motions. It contains motion capture data or manually designed motions. The Game Action Dataset (GAD) dataset [46] is a relatively small dataset comprised of recordings of gaming sessions (GTA5 and FIFA) performed by human players. This dataset is the first derived from gaming environments and includes synchronized ground and aerial views. The game generates character motions that are conditioned on players’ commands. Verol et al. used 3D human motion estimation models, such as HMMR [20] and VIBE [22], to reconstruct the human body mesh and its motions from a single view RGB videos to create the SURREACT [50] dataset. The body mesh is based on the SMPL [31] statistical model and is further augmented with randomized cloth textures, lighting conditions, and body shapes for better diversity. SynADL [18] is a synthetic dataset focused on detecting elders’ activities of daily living (ADL). The 3D human characters were created by scanning 15 participants with a Kinect sensor. Motion capture data were recorded from the same 15 participants and used to animate the characters. Kim et al. combined PHAV, SURREACT, and SynADL to create a new dataset called SynAPT [51]. The goal of this dataset was to pre-train a model which would be transferred to a different downstream task involving completely new categories. RoCoG [6] and RoCoG-v2 [40] are two datasets designed for human-robot interaction based on seven gestures from the US Army Field Manual [16]. RoCoG-v2 presents static ground and aerial views, unlike RoCoG, which contains only a static ground view. Moreover, RoCoG contains only manually-designed motions, whereas RoCoG-v2 introduces motion capture data.
Table 1. Synthetic HAR datasets comparison sorted by the year of publication. The underlined datasets include real counterparts (Table 2).

<table>
<thead>
<tr>
<th>Name</th>
<th>Year</th>
<th>Num. Act.</th>
<th>Motion Source(^a)</th>
<th>Total Frms</th>
<th>Num. Seq.</th>
<th>FPS(^q)</th>
<th>Rendering</th>
<th>Human Models</th>
<th>Camera Views(^c)</th>
<th>Virt. Env.(^d)</th>
</tr>
</thead>
<tbody>
<tr>
<td>PHAV [8]</td>
<td>2017</td>
<td>15</td>
<td>Ma., Ga., Vid., MC</td>
<td>39.9k</td>
<td>150</td>
<td>Unity</td>
<td>CG</td>
<td>R15(^f)</td>
<td>7</td>
<td>In., Ga.</td>
</tr>
<tr>
<td>RoCoG [6]</td>
<td>2020</td>
<td>8</td>
<td></td>
<td>17.6M</td>
<td>110k</td>
<td>Unity</td>
<td>CG</td>
<td>4 HQ(^g)</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>GAD [46]</td>
<td>2021</td>
<td>7</td>
<td></td>
<td>0.3M</td>
<td>1.4k</td>
<td>GTAY FIFA</td>
<td>CG</td>
<td>n/a</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SURREACT [50]</td>
<td>2021</td>
<td>100</td>
<td></td>
<td>9.3M</td>
<td>109k</td>
<td>Cycles(^h)</td>
<td>CG</td>
<td>SMPL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SynADL [18]</td>
<td>2021</td>
<td>55</td>
<td></td>
<td>135.0M</td>
<td>462k</td>
<td>U4</td>
<td>CG</td>
<td>15 Kinect(^k)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SynAPF [51]</td>
<td>2022</td>
<td>150</td>
<td></td>
<td>150.0k</td>
<td></td>
<td>Unity</td>
<td>CG</td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SynCG-MC (ours)</td>
<td>2023</td>
<td>7</td>
<td></td>
<td>19.6M</td>
<td>107k</td>
<td>Eevee(^l)</td>
<td>CG</td>
<td>12</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SynCG-RGB (ours)</td>
<td>2023</td>
<td>11</td>
<td></td>
<td>3.14M</td>
<td>25.4k</td>
<td>Eevee(^l)</td>
<td>CG</td>
<td>3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SynLWG-MC (ours)</td>
<td>2023</td>
<td>11</td>
<td></td>
<td>31.2M</td>
<td>25.4k</td>
<td>Eevee(^l)</td>
<td>CG</td>
<td>3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SynLWG-RGB (ours)</td>
<td>2023</td>
<td>5</td>
<td></td>
<td>5.0M</td>
<td>6.1k</td>
<td>Eevee(^l)</td>
<td>CG</td>
<td>3</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

\(^a\) Four motion sources: manually designed motions (Ma.), game engine motions (Ga.), video-extracted (Vid.), and motion capture data (MC)

\(^b\) The numbers depict the number of subjects included in the data

\(^c\) View types: Static (St.), Dynamic (Dy.), Aerial (Ae.), Ground view (Gr.)

\(^d\) Virtual Environments: Indoor (In.), Outdoor (Out.)

\(^e\) 15-muscle ragdoll model

\(^f\) Game character

\(^g\) Blender Cycles render engine

\(^h\) 25-joint models made by scanning human participants with a Kinect sensor

\(^i\) A combination of PHAV, SURREACT, and SynADL datasets

\(^j\) Blender Eevee render engine

\(^k\) Liquid Warping GAN [29]

\(^l\) Neural Renderer

\(^m\) SMPL meshes were textured using 32 real human avatars

\(^n\) High-quality human assets from public repositories

\(^o\) Mean number of frames per sequence

\(^p\) Models have 25 motion capture-animated joints, high-resolution meshes and textures

Table 2. Real HAR datasets comparison sorted by the year of publication.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>UCF-ARG [13]</td>
<td>2010</td>
<td>10</td>
<td>12</td>
<td>1.4k</td>
<td>3</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>HMDB51 [24]</td>
<td>2011</td>
<td>51</td>
<td></td>
<td>6.8k</td>
<td>1</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>UCF-101 [44]</td>
<td>2012</td>
<td>101</td>
<td></td>
<td>13.3k</td>
<td>1</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Kinetics-400 [21]</td>
<td>2017</td>
<td>400</td>
<td></td>
<td>306.2k</td>
<td>1</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Charades-Ego [42]</td>
<td>2018</td>
<td>157</td>
<td>112</td>
<td>8.0k</td>
<td>2</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Kinetics-600 [2]</td>
<td>2018</td>
<td>600</td>
<td></td>
<td>495.4k</td>
<td>1</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Kinetics-700 [3]</td>
<td>2019</td>
<td>700</td>
<td></td>
<td>650.3k</td>
<td>1</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Kinetics-700 2020 [43]</td>
<td>2020</td>
<td>700</td>
<td></td>
<td>647.9k</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NTU-RGB+D 120 [27]</td>
<td>2020</td>
<td>120</td>
<td>106</td>
<td>114.5k</td>
<td>3</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>RoCoG [6]</td>
<td>2020</td>
<td>8</td>
<td>14</td>
<td>1.5k</td>
<td>1</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HOMAGE [58]</td>
<td>2021</td>
<td>75</td>
<td>27</td>
<td>5.7k</td>
<td>2</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UAV-Human [26]</td>
<td>2021</td>
<td>155</td>
<td>119</td>
<td>22.5k</td>
<td>1</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>VAD [46]</td>
<td>2023</td>
<td>8</td>
<td></td>
<td>0.4k</td>
<td>1</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RoCoG-v2 [40]</td>
<td>2023</td>
<td>7</td>
<td>10</td>
<td>0.5k</td>
<td>2</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Real (ours)</td>
<td>2023</td>
<td>11</td>
<td>24</td>
<td>1.5k</td>
<td>3</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

We introduce four unique synthetic datasets: *SynCG-MC*, *SynCG-RGB*, *SynLWG-MC*, and *SynLWG-RGB*, generated using a combination of a motion source (motion capture or video-based motions) and a rendering technique (*CG* and *neural rendering*). As evident from Table 1 *SynLWG-MC* and *SynLWG-RGB* are the only HAR datasets based on neurally-generated videos, all other datasets are created using different kinds of CG renderers. To the best of our knowledge, we are the first to generate such HAR datasets and compare them with the standard CG engines. Our synthetic datasets were created using 32 realistic rigged 3D human characters generated using 3D real people. They were animated using a 26-participant motion capture library we created. Furthermore, very few aerial synthetic HAR datasets exist, which is also apparent from Table 1. The two other datasets are GAD and RoCoG-v2. Still, neither contains nor provides analysis on a more complete set of camera views as we do, namely, synchronized dynamic aerial and static ground views. Moreover, the test results in [51] highlight the need for more synthetic aerial HAR datasets. The authors use a combination of three synthetic datasets (PHAV, SURREACT, and SynADL) without aerial view data to pre-train their models. UAV Human [26], the only aerial dataset out of the six real test datasets, delivers the lowest accuracy.

### 2.2. Real HAR Datasets

We also provide a real dataset containing the same three camera views and action categories as the four synthetic variants mentioned in the previous section. This dataset provides a performance baseline (train and test on real data only), a real test set, and a fine-tuning dataset. More details for creating it are provided in Section 3.2. Table 2 briefly compares our real and other currently existing HAR datasets.

The *UCF-ARG* [13] dataset varied the scope of the camera angles by providing videos from aerial and rooftop cameras. The aerial view was captured by a camera attached to a balloon. *HMDB51* [24] was one of the first datasets with an increased number of action categories, providing 51 action categories from 6766 videos. *UCF-101* [44] dataset introduced a larger dataset with 101 activity classes subdivided...
into five categories. Introducing massive action datasets from the Kinetics family (Kinetics-400 [21], Kinetics-600 [2], Kinetics-700 [3], and Kinetics-700-2020 [43]) accelerated the progress in the field by providing several hundred activity categories and close to 1000 videos per category. Charades-Ego [42] and HOMAGE [38] are datasets of videos of daily human activities recorded from first- and third-person perspectives. The NTU-120 dataset also introduced multiple-view datasets for action recognition with 120 classes from 110,000 clips of video, depth map sequences, 3D skeletal data, and infrared videos for each sample. UAH-Human [26] is a dataset of human activities captured by a flying UAV in multiple urban and rural districts in daytime and nighttime. It is the largest real HAR dataset which contains an aerial view. YAD [46] (YouTube-Aerial Dataset) is a small HAR dataset containing aerial videos from YouTube. It includes large and fast camera motions as well as variable shooting altitudes. RoCoG [6] RoCoG-v2 [40] also contain real data of the same activity categories as their synthetic variants. RoCoG-v2 contains aerial and ground views, whereas RoCoG provides only a static ground view.

2.3. HAR Data Neural Renderers

Models like Everybody Dance Now [5] and Liquid Warping GAN (LWG) [29] generate videos by transferring the body pose from a video of a person performing an activity to a new person from another video or an image. They can generate action videos with different appearances depending on the input visual source image. However, these models rely on body pose estimation from monocular videos or images, which is inherently less accurate than motion capture. We integrated MoSh [30], a model for generating 3D human SMPL meshes from a motion capture marker set, into the LWG architecture. Thus, we extended its capabilities to generate videos from motion capture data.

In recent years, NLP models have demonstrated the capability to generate different data types from text prompts. Models like TEMOS [34], MotionCLIP [48], MDM [49], T2M-GPT [55], and MotionGPT [19] can generate realistic human motions by taking activity descriptions (text-to-motion). Other models like Stable Diffusion [41] and DALL-E [39] can generate images from text (text-to-image). Ma et al. [32] also made one of the first attempts for a pose-guided text-to-video generation. However, it is hard to incorporate such models into our pipeline because they do not offer fine-grained control over the generated output and establish precise motion matching across different synthetic sets—an important aspect of our analysis.

2.4. Deep Video Action Recognition

An expansive set of activity recognition models has been developed in recent years [35]. This section reviews some of the most popular and accessible ones used in our analysis.

Architectures like I3D [4] and X3D [11] were built on image classification networks applied to videos. I3D does that by inflating the filters and pooling kernels to 3D, while X3D progressively expands on a tiny 2D image classification model with multiple axes (space, time, width, height). The performances of I3D and X3D are 73% and 77%, respectively when tested on Kinetics [9]. The SlowFast [12] architecture was built upon a different approach, combining the slow and fast pathways, which take small and large frame rates of the video, respectively, to identify gestures happening at shorter and longer time frames. Its performance on Kinetics was 77%. MVIT [10] is founded upon multi-scale transformers to create a multi-scale pyramid of features to learn coarse to complex features. So far, MVIT has shown the best performance on Kinetics—83% [9].

3. Our Datasets

This section describes the creation of our dataset suite consisting of a real-world and four synthetic video datasets.

3.1. Activity Classes

Our datasets contain eleven daily human activity categories (Figure 2), where five are Gestures and six involve Object-handling. Every frame that does not fit into any of these categories was labeled as Idle. The set of activities was intentionally constructed to contain confounding pairs to increase the difficulty of the recognition task. For instance, Carrying a shovel on the shoulder resembles Carrying a bat on the shoulder; some instances of Holding out a flashlight may be confused with Talking on the phone; Waving “Hello or Goodbye” is similar to Shaking fist.

3.2. Real-World Capture

This dataset contains video recordings of human participants using multiple cameras during daylight hours in three outdoor environments: a grass field, a parking lot, and a tennis court. The data collection spanned eight months (November 2021 to July 2022). Thus, different seasons were captured. All subjects consented with an approved Institutional Review Board (IRB) for video or motion capture.

In total, 24 subjects (18 male and 6 female) participated in the video collection process. Most of them were captured with three cameras recording simultaneously in 4K (3840 px × 2160 px) resolution at 30 fps. One camera was a remotely-controlled small UAV (DJI Mavic 2 Zoom), which circled over the participant at a constant speed as shown in Figure 3. This camera constitutes the aerial view. The target radius of the circular UAV trajectory $R_{\text{traj}}$ is 15 m. The flying altitude $H_{\text{traj}}$ was maintained within the 11 m – 13 m range. These two trajectory parameters were selected such

1One and a half participant sessions lack the second ground camera.
that the camera’s tilt angle $\theta_{\text{tilt}}$ would vary within the range $30^\circ – 45^\circ$. The mean height of the participant in the image frame was kept to about 200 px by adjusting the lens focal length. The subject performed every activity twice, once for each flying direction of the UAV (clockwise and counter-clockwise). The other two cameras were stationary (mounted on tripods) and elevated about 1.3 m above the ground. They constitute the ground view. The three views can be seen in Figure 1.

To synchronize the three video streams, we asked each participant to clap their hands above their head at the beginning of each session. The temporal offsets of the streams were determined manually in post-processing using the clapping actions. The annotation software ELAN [53] was used to synchronize and label the activity being performed.

The real-world dataset contains a total of 1538 video sequences. For all videos, we detected the subjects in each frame using Faster R-CNN [54] and cropped around the center of the bounding box. The frames were resized to 224 px $\times$ 224 px. The videos have a frame rate of 30 fps. The total frame count of the dataset is 1 844 253 ($\sim$17.08 h). The mean video sequence length is 39.97 s.

3.3. Synthetic Data Generation

Two generation methods were employed: a standard computer graphics (CG) pipeline and a deep generative neural model called Liquid Warping GAN (LWG) [29]. To animate the human characters, we used two sources of motion: a motion capture (MC) dataset we collected and motions extracted from real video sequences (RGB). We utilize a simple naming convention to differentiate between the four synthetic datasets—Syn (Renderer) - (MotionSource): SynCG-MC, SynCG-RGB, SynLWG-MC, SynLWG-RGB. Below, we provide more details on how each one was created.

**Fully CG-based synthetic data** We used Blender as the main 3D scene development and rendering environment (Eevee engine). MotionBuilder [1] was used as motion data editing and re-targeting software. We used 32 commercially-available rigged and skinned human characters [14] (Figure 4) divided into two gender groups—16 male and 16 female. We collected motion capture data from 26 human actors$^2$ (19 male, 7 female) performing the

$^2$For more information, refer to the supplementary material.
same eleven activities from Figure 2 and re-targeted it to the characters from their gender group. Therefore, each subject’s data were used to animate 16 characters. A 3D graphics artist designed three virtual environments (visualizations can be found in the supplementary material) resembling the ones from the real data capture. The ground plane meshes were reconstructed from the shooting locations through photogrammetry [45]. Various vegetation (trees and grass) and structure (buildings and fences) assets were added manually. We randomized the camera location, the lighting conditions, and the characters’ clothing colors to introduce more diversity in the final synthetic images. The colors were sampled from the real videos.

Each of the three camera views of the SynCG-MC dataset is represented by 8648 video sequences that contain 10,453,420 frames—equivalent to almost 100 h of data. In total, this synthetic variant contains 31 million frames, equivalent to about 290 h of data, roughly 17 times more than the real dataset.

While collecting the motion capture data, we also captured RGB videos from 15 subjects. With this data, we generated another variation of the data rendered from the CG pipeline, SynCG-RGB. We used VIBE [22] to fit the SMPL [31] parametric model for each person in the videos and produce 3D meshes and skeletons. They were used to animate the synthetic avatars, similar to SynCG-MC. The rendering pipeline follows that of SynCG-MC. LWG-generated Data A major drawback of using the traditional CG pipeline is that it requires time and manual effort to design virtual scenes and render them. To solve this problem, we used a neural-based rendering method to generate videos of the same gestures. Because the model has been pre-trained on priors to generate video of human motions, it reduces the processing for generating a single video sequence.

In particular, we used LWG with Attention [29] to generate the second set of synthetic videos. LWG is a unified framework for human image synthesis, including human motion imitation, appearance transfer, and novel view synthesis, with a 3D body mesh recovery module that utilizes SMPL [31] to disentangle the pose and shape.

We chose LWG because it allowed us to easily incorporate motion capture data a second source of human motion data. The original pipeline relies on a pose reconstruction model from 2D videos called SPIN [23]. We used the indoor videos we captured while collecting the motion capture data to generate a synthetic HAR dataset with this non-traditional, neural-based method (SynLWG-RGB). This approach resulted in lower motion quality compared to that of the motion capture. We also generated a counterpart of the SynLWG-RGB dataset by modifying the original LWG pipeline by replacing the motions extracted from SPIN with SMPL fit on motion capture and CG source images (SynLWG-MC). We used the SMPL parameters fitted on raw marker data from motion capture using MoSh [30]. SynLWG-MC had a better quality of motion (less jittery) compared to SynLWG-RGB.

For both SynLWG datasets, we used front and back rendered images of the 3D human character to generate the input source images (Figure 4). For each sequence, we used the same background, avatar, and motion capture sequence that was used in the corresponding CG sequence. All frames were resized to $224 \times 224$ px and were generated at 30 fps.

4. Experiments

We used three deep video activity recognition models in our experiments: SlowFast [12], MViT [10], and X3D [11]. All models are part of the PyTorch [33] implementation of SlowFast on GitHub [9].

We trained all models for 300 epochs and used top-1 classification accuracy to compare their performance. We used Kinetics-400 [21] pre-trained network weights. We used an SGD optimizer for Slowfast and X3D, and an AdamW optimizer for MViT with a cosine learning schedule of learning rate decay starting with 0.001 following [10–12]. We trained on 4 GPUs (batch size 7/GPU).

During training, we employed three 50% probability augmentation methods: color-jittering (brightness, hue, contrast, saturation), random Gaussian blur, and sharpness adjustment. We randomly select 64-frame clips from videos, count frame labels, and assign the label with the highest count to the clip. All training samples contain at least 80% of frames with the same label.

During testing, we evaluated all 64-frame clips from the videos and predicted their labels. Training procedures vary depending on the dataset. For real data, we exclude test subjects. For lab-captured datasets, we train on all data and test on the real data for three subject groups.

4.1. Analysis

This section presents the experiments we conducted using our dataset suite. They were designed to answer the following questions:

**Does the choice of the ML model make a significant difference in performance?** We set a performance baseline by training three activity recognition models (SlowFast, MViT, X3D) on real-world data. Each model is trained on eleven activity classes from 21 subjects, excluding the test subjects. Three models were trained for each camera view, each with a distinct test group.

The results from this experiment are shown in Figure 5. On average, SlowFast outperforms MViT and X3D for both...

---

3 The lighting conditions of each scene were determined by randomly selecting one out of 13 HDRi environmental spherical panoramas downloaded from Poly Haven [15].
camera views most of the time. We believe that this result is because MVIT and X3D are designed for high spatio-temporal resolution samples. Our input sample size of 64 frames with 224 px × 224 px was too small to take advantage of this functionality. Based on these initial results, we used only SlowFast for the remainder of the experiments.

We also observe that the ground view always performs better than the aerial one (Figure 5). We believe this is because the aerial view is recorded with a moving camera, making it more challenging to learn the task-specific spatio-temporal features. The ground cameras provide two views and, therefore, twice as much data. The overhead camera angle may also degrade performance.

**Does the synthetic data rendering technique affect the model performance?** A unique feature of our dataset suite is that it is generated using two different rendering techniques. We compared the performance of SlowFast trained on three datasets, Real, SynCG-MC and SynLWG-MC for eleven activities. The models were tested on three real test groups, excluding overlaps with the training data.

The results in Figure 6 show that the models trained on real data consistently outperform those trained on synthetic data by 4-8%. SynCG-MC data performs strictly better than SynLWG-MC, suggesting the distribution gap between the data generated by the standard CG-based method for rendering and the real data is smaller than the one between the LWG approach and the real data.

**Does improving the quality of the motion in the synthetic training data improve the performance?** Another property of our dataset suite is that we used two different motion sources to generate the synthetic dataset. Motion capture systems collect the skeletal data in 3D, whereas the motion extracted from the RGB video extrapolates the 3D information from the 2D data. Therefore, the synthetic data generated using motion capture data has superior quality. We compared the performance of SlowFast trained on Real, SynCG-MC, SynCG-RGB, SynLWG-MC, and SynLWG-RGB. Because SynCG-RGB has only 15 subjects, we also limited the other datasets to 15 subjects. We evaluated the models for the five gesture-only classes.

Figure 7 shows that the model trained on real data performs the best for both views. For ground view data, the motion source does not significantly influence model performance in both CG and LWG-rendered videos. In aerial data, motion capture-generated datasets outperform video-extracted motion datasets, likely because ground cameras focus on lateral axes, aligning with pose reconstruction methods. In contrast, aerial views encompass diverse angles, making the quality of the motion more important.

**Can we combine synthetic training data with a limited amount of real training data to improve performance?** To assess this hypothesis, we created two subsets from the real dataset that include 5- and 10-subject data. Then, we used them to train/fine-tune three models—a model untrained on any of our datasets and two pre-trained on SynCG-MC and SynLWG-MC. Our results (Figure 8) indicate that pre-training a model on synthetic data and subsequently fine-tuning it on a small amount of real data outper-
forms a model trained only on the exact small amounts of real data. That is valid for both camera views. Furthermore, the models pre-trained on SynCG-MC consistently outperform the ones pre-trained on SynLWG-MC. The improvement is considerably more subtle for the ground view when the larger (10 subjects) real subset is used. The overall limited diversity of the static ground view data most probably causes that effect, which underlines the importance of the synthetic HAR datasets for improving the performance on more dynamic scenes, such as the aerial view.

Can the models trained on one camera view transfer to a novel one? In this experiment, we evaluated the robustness of the model trained on our datasets to the change of camera viewpoints. We did that by cross-view evaluation. We ran two sets of experiments, one with the five gesture-only datasets and the other with all eleven behaviors for Real, SynCG-MC and SynLWG-MC datasets excluding any overlaps with the test set.

The results in Figure 9 show that the model trained on the aerial view is more robust to view change than the model trained on the ground view for all types of training data. We hypothesize that two factors are responsible for this difference in performance. First, while the ground cameras only see the subject from the front, drone cameras see the subject from all sides. This more varied viewpoint allows the model trained on aerial data to transfer to the ground data more easily than in the opposite direction. Second, the difference in the sizes of the ground and the aerial datasets play a role: because the ground view data is twice as large as the aerial view data, it is harder for the model trained on ground view data to transfer to the aerial view.

5. Conclusion

In this paper, we introduced a new HAR dataset suite containing real and synthetic data captured from ground and dynamic aerial camera perspectives. Synthetic data was generated using two rendering methods: standard computer graphics (CG) and neural-based rendering (LWG). We evaluated synthetic data performance against baseline models trained on real-world data, yielding the following findings: (1) Training data rendering method matters (CG outperforms LWG). (2) Motion quality is more crucial than rendering quality for model performance. (3) Fine-tuning models with a smaller batch of real data after pre-training on synthetic data improves performance, sometimes surpassing models trained solely on the full real dataset. (4) Synthetic data training with diversity enhances model robustness to changes in camera view compared to real data training. In summary, the quality of synthetic data is vital for bridging the domain gap with real-world data, and even small amounts of real-world data can boost performance through fine-tuning. Future experiments should explore broader activities, alternative synthetic data forms, and the impact of fine-tuning and data mixing ratios.

6. Acknowledgement

We thank Melanie Danver and Justin Macey for their valuable help in creating the datasets. This work has been funded in part by the DEVCOM Army Research Lab. Desislava Nikolova did her work while visiting Carnegie Mellon University, sponsored by the Bulgarian-American Commission for Educational Exchange Fulbright.
References


[34] Mathis Petrovich, Michael J. Black, and Gül Varol. TEMOS: Generating diverse human motions from textual descriptions. In *European Conference on Computer Vision (ECCV)*, 2022. 4


[40] 3Dflow s.r.l. 3D Flow Zephyr. *https://www.3dflow.net/3d-flow-zephyr-photogrammetry-software/*, Accessed: 2023-11-03. 6


