Article

Low-Cost Embedded System Using Convolutional Neural Networks-Based Spatiotemporal Feature Map for Real-Time Human Action Recognition

Jinsoo Kim and Jeongho Cho*

Department of Electrical Engineering, Soonchunhyang University, Asan 31538, Korea; js.kim@sch.ac.kr
* Correspondence: jcho@sch.ac.kr; Tel.: +82-41-530-4960

Abstract: The field of research related to video data has difficulty in extracting not only spatial but also temporal features and human action recognition (HAR) is a representative field of research that applies convolutional neural network (CNN) to video data. The performance for action recognition has improved, but owing to the complexity of the model, some still limitations to operation in real-time persist. Therefore, a lightweight CNN-based single-stream HAR model that can operate in real-time is proposed. The proposed model extracts spatial feature maps by applying CNN to the images that develop the video and uses the frame change rate of sequential images as time information. Spatial feature maps are weighted-averaged by frame change, transformed into spatiotemporal features, and input into multilayer perceptrons, which have a relatively lower complexity than other HAR models; thus, our method has high utility in a single embedded system connected to CCTV. The results of evaluating action recognition accuracy and data processing speed through challenging action recognition benchmark UCF-101 showed higher action recognition accuracy than the HAR model using long short-term memory with a small amount of video frames and confirmed the real-time operational possibility through fast data processing speed. In addition, the performance of the proposed weighted mean-based HAR model was verified by testing it in Jetson NANO to confirm the possibility of using it in low-cost GPU-based embedded systems.

Keywords: CNN; human action recognition; spatiotemporal feature; embedded system; real-time

1. Introduction

Human action recognition (HAR) in video is one of the most challenging tasks in the field of computer vision as it requires simultaneous consideration of spatial and temporal representations of motion [1]. Unlike image classification [2] and object detection [3], which utilize spatial expression extracted by convolutional neural network (CNN) based on an image, HAR recognizes action through spatiotemporal feature extracted from time-varying motions, as well as the appearance of the person extracted from the video, which is a series of images [4]. During the early stage, research [5,6] that recognized actions through a two-dimensional CNN, which learns existing spatial expressions, was performed; however, action recognition was difficult as there were limitations in terms of learning the conditions and temporal features, such as the scale and pose of the human appearance, the similarity of movement, and the change of camera’s point of view [7]. Therefore, CNN-based models that learn motion representations through spatiotemporal features are being proposed [8–10], because action in video is recognized based on extracted spatiotemporal features by identifying the connectivity of movements that change over time. The models that recognize action by learning time-varying motion patterns based on CNN have achieved significant performance improvements in the field of HAR [11,12]. The latest CNN-based models for HAR learn time-varying motion representations via an extended CNN structure that combines CNNs applied to static images with networks that extract temporal features [13]. Typically, there are 3D CNNs that simultaneously extract...
spatiotemporal features, which represent human appearance and motion representations through filters, two-stream CNN, which combines spatial CNN network and temporal CNN network, and convolution recurrent neural network (CRNN), which combines CNN and recurrent neural networks (RNNs) [14].

3D CNN is a structure that applies a convolution layer consisting of 3D kernels to a 3D image stack generated by stacking sequential images over time, simultaneously learning the spatial and temporal features of the input data through a 3D kernel [15]. Therefore, it has the advantage of recognizing action by directly generating hierarchical representations of spatiotemporal information through motion information encoded from the sequential image, and there is a disadvantage [16] that computationally-heavy 3D CNN-based models have high computation and memory cost.

Two-stream CNN recognizes action through a model that fuses spatial CNN network, which extracts appearance information—a spatial feature of motion—from images, and temporal CNN network, which extracts temporal features from motion vectors changing over time, such as optical flow, in parallel structures [17]. Such parallel structures overcome the limitations that conventional CNNs are difficult to learn temporal representations and efficiently fuse spatial features and temporal features extracted from optical flows through late fusion. However, due to the fusion of two CNN networks into a parallel structure, it has a limitation [18] that it is difficult to apply to real-time HAR systems with a long data processing time compared to single-stream CNNs.

CRNNs are single-stream CNN structures and recognize action by identifying context for time-varying motions through RNNs after extracting spatial feature maps from sequential image stacks to CNNs [19]. Long short-term memory (LSTM) is used as a network to analyze time information, and a model that fuses CNN and LSTM is called ConvLSTM. ConvLSTM is an LSTM that learns time-varying motion patterns from images, so it has the advantage of identifying the connectivity of spatial features that change over time and of extracting temporal features well. However, ConvLSTM has the disadvantage that the more images that are extracted from the video, the more the complexity of the model increases.

The CNN-based HAR models that were mentioned can be used in various fields that require actual HAR through the action recognition performance that improved compared with existing techniques, but owing to the complexity of the model and long data processing time, there are still limitations to be operated in real-time in low-cost embedded systems [20]. Recently, HAR has been applied to integrated monitoring systems, which detect emergency situations with CCTVs. Most of integrated monitoring systems receive multiple CCTV images through network-based digital video communication and extract monitoring information [21] as shown in Figure 1. The video data transmission process is used to recognize behavior in a server PC equipped with a GPU that has strong computational efficiency owing to the complexity of the HAR model. Therefore, HAR models have a limitation in recognizing behavior in real-time, being installed in a single embedded system directly connected to the CCTV.

![Figure 1](image_url)

Figure 1. An example of a wireless communication network for recognizing behavior through CCTV using a complex HAR model.
In particular, it is essential for the system to recognize human action quickly in urgent situations related to the safety, such as violence and theft on the streets, and in emergencies, such as detecting urgent situations or abnormal behaviors of the elderly and single-person households [22]. When the action is recognized through the HAR model in the embedded system where the camera is installed, time and cost in the process of transmitting high-capacity video data can be reduced [23].

Therefore, in this paper, we propose a weighted mean-based single-stream CNN model that recognizes action faster than conventional models. As the proposed method has a simpler structure than the existing CNN-based models with high complexity, recognizing behavior in a single embedded system connected to CCTV without transmitting image data through network communication is possible. The main idea is to build a lightweight CNN-based HAR model that can be applied to low-cost embedded systems with low-end GPUs to apply HAR to CCTV-based surveillance fields and to recognize instantaneous motions in emergency situations at high speed. The proposed system extracts spatial and temporal features by weight and averaged the change rate of frames according to time on a spatial feature map extracted by CNN. The weighted mean is calculated sequentially by the change rate of each frame extracted according to any time interval from the video and the change rate of frame at the corresponding time is weighted in the process of averaging feature maps, so the spatial and temporal features are created in the contexts of the integrated time-varying motion representation. Spatial feature maps at a specific time where the amount of motion change is high are weighted more than the point of time when the amount of change is low and the context of the motion representation is created; spatial representations are extracted efficiently at the point of time affecting the recognition performance of systems. The weighted mean also recognizes action by processing data at a rapid rate through a lighter structure than conventional CNN-based HAR models. 3D CNNs extract temporal features using spatiotemporal filters, two-stream CNNs utilize optical flows, and ConvLSTMs use LSTMs to extract temporal features. Such models take a long time in the calculation process with an extended CNN structure combining networks that extracts temporal features in the existing two-dimensional CNN structure. However, the proposed model recognizes action by inputting a one-dimensional spatiotemporal feature vector into FC layers, where the vector is generated by spatial feature maps and frame change rate using weighted mean. Therefore, video data are processed at high speed while maintaining the existing two-dimensional CNN structure.

2. Related Works

In ImageNet Challenge 2012 [24], a deep learning-based CNN with superior performance than the algorithm that was to be used in the existing computer vision has been proposed, and in recent studies, the deep learning-based CNN shows high level of usability for HARS. HAR models applied by CNN automatically extract and learn motion features from video and utilize the motion features extracted from different modalities of data to enhance the performance for action recognition according to modality of data perspective; action recognition models are largely divided into depth-, skeleton-, and vision-based HAR [25].

2.1. Depth-Based Human Action Recognition

Depth- and skeleton-based HARS [26,27] recognize actions using change in motion representation of the depth map acquired through the depth sensor. The depth map comprising RGB-D video has the spatiotemporal structure. Changes in the depth information over time are extracted to the spatiotemporal features of motion [28]. In addition, depth maps clearly separate people from backgrounds to represent appearance information, so they can be used for meaningful feature extraction for action recognition. Zhanga et al. [29] propose orientation histogram features of 3D normal vectors to extend the features of histogram oriented gradient (HOG) extracted from the depth map to a spatiotemporal depth structure and to represent appearance information of a three-
dimensional depth structure of the spatiotemporal depth. Authors in [30] construct super-normal feature vectors based on depth map sequences to represent motion representations for action recognition. Feature vectors are generated by applying spatial average pooling and temporal maximum pooling to time-varying depth maps, and the evaluation results on various benchmark datasets show robustness about scale change.

The HAR models using depth information have shown high action recognition performance, but have limitations, applying only to a limited range and specific environment. Generally used depth sensors include stereo cameras using triangulation techniques, time of flight (TOF) cameras, and structured-light cameras. Depth sensors using stereo cameras are inexpensive, but the process of calculating depth information is complicated, making it difficult to acquire accurate depth information. Depth sensors based on TOF cameras and structured-light cameras have limitations of difficulty to be applied outdoors, which are heavily influenced by light. In addition, the depth sensors represent the information included in the measurable distance as data, which is difficult to apply to outdoor surveillance fields. Some sensors, such as light detection and ranging, are robust to lighting and measure a relatively wide range of depth but are expensive and not suitable for use in video-based surveillance systems.

2.2. Skeleton-Based Human Action Recognition

Skeleton-based HAR [31,32] recognizes action through joint points extracted via CNN-based pose estimation algorithms from depth maps or RGB images. The location of a person’s joint points represented by the time axis extracted from the video is used as the feature vector, which is connected according to the body structure of the person, and adjacent points have an important correlation with each other.

Warchol et al. [33] proposed a CNN-based HAR model that automatically learns the spatial and temporal features of data based on joint points. To reduce the redundancy of data and preserve spatiotemporal features, key frames are extracted using interframe difference methods, and joint points are generated through the open-pose [34] and then CNN is applied. Recently, more research related to skeleton-based HAR has been proposed than depth-based HAR, but since the process of extracting joint points with pose estimation algorithms is preceded, there is a disadvantage that the accuracy of joint points varies depending on sensor performance. The whole recognition performance of systems can be degraded if joint points containing noise are obtained by sensor performance, or if they are affected by external environmental factors such as lighting and occlusion [35].

2.3. Vision-Based Human Action Recognition

Before a deep learning-based CNN was applied to HAR, the conventional method recognized actions based on hand-crafted features for some human-performed actions in a simple background. Hand-crafted features include spatiotemporal interest points (STIPs) [36], 3-dimensional HOG [37], 3-Dimensional Scale Invariant Feature Transform (3D-SIFT) [38], which use various feature encoding schemes such as histograms or pyramids. In [39], the geometric properties of space-time volume according to human movement are extracted with action sketches, which stacked body outlines on the time axis according to direction, speed, and shape. These low-level features were entered into machine learning-based classification algorithms, such as SVM, decision tree, and K-nearest neighbor, and used in HARS.

However, the deep learning-based CNN is proposed and research on various CNN-based methods is being conducted to build HAR models; typically, there are methods such as 3D CNN [40], two-stream CNN [41], and ConvLSTM [42]. Recently, a study of building HAR models by fusing the above methods and a method of entering depth maps and joint points into an algorithm of vision-based HAR models have also been proposed. Karpathy et al. [43] propose a spatiotemporal LSTM (Spatial LSTM) model for 3-dimensional HARs that extend the RNN into the spatiotemporal domain to analyze the hidden features of motion representations. In addition, in [44], a study was conducted to classify an action in
spatial and temporal encoding information of depth map sequences by applying 3D CNNs to encode motion patterns with spatiotemporal features in the depth map sequences. The advantages and disadvantages of these three HAR modalities are summarized in Table 1.

| Modality | Advantage | Disadvantage |
|----------|-----------|--------------|
| Depth    | Clearly separate foreground and background | Vulnerable to light |
| Skeleton | Extract correlations from joint points connected by body structure | Data uncertainty |
| Vision   | Includes various visual information with high resolution | Data capacity |

3. Proposed Methodology

3.1. System Overview

The proposed embedded system-based HAR model uses a spatial feature map extracted using CNN through the weighted mean and the temporal feature extracted via frame change rate, and then, generates spatiotemporal features. The generated spatiotemporal features are entered into a multilayer perceptron (MLP), which is lighter than the existing networks that were used in the HAR models, outputting action classes and schematizing the block diagram of the entire system, as shown in Figure 2. Image sequences, which make up a video, entered into a model, are converted into N frame stacks ($FS^N$) with random intervals and extracted via CNN into feature map ($FM^N_k$) with spatial features ($K$ means the length of a feature map that is flattened in one dimension). Here, CNN is used as a feature extractor to extract a spatial feature map, and the fully connected layer connected to the end of the CNN is removed. $FS^N$ extracts frame difference ($FD^N$) of each frame constituting $FS^N$ as a temporal feature, since the sequential frames within a video contain time information with sequentially stacked data. Finally, to fuse spatial and temporal features, $FM^N_k$ is weighted and averaged to $FD^N$ according to the interval of $FS^N$ converted from video to generate feature vectors ($FV^N_k$) with spatiotemporal features and input them into MLP. The MLP, which is independent from CNN, outputs predefined action classes and schematized the process of recognizing action by receiving video data, which is shown in Figure 3.

![Figure 2. Block diagram of the proposed weighted mean-based human action recognition (HAR) model.](image-url)
3.2. Extract Spatial Features

Videos are sequential data, in which frames representing spatial representations are listed according to time information, and the sequence of frames is the time information required to recognize the context of the action contained in the video. The video from the UCF-101 dataset processes frames at 25 fps. When all frame sequences per unit time are entered into the model, the processing time according to the computational cost increases and has a lot of redundant data. Considering processing complexity according to the data, the proposed system extracts N frames at random intervals and converts the image sequence of input video into $FS^N$.

$FS^N$ is inputted to the CNN structure VGGNet [45], which is pretrained with ImageNet dataset and is extracted to feature maps. Moreover, the resolution of the images in $FS^N$ is converted to (224, 224, 3) according to the kernel size of VGGNet. VGGNet is a CNN structure that has deeply stacked convolutional layers through (3, 3) filters, with fewer parameters than convolutional layers, which were shallowly stacked through a larger sized filter. Therefore, although VGGNet is a deep layer structure, it is a CNN structure that has been proven to extract feature maps at high speed from input data and proven to extract significant features, which reduce classification errors as the layer deepens. In addition, VGGNet has the size (7, 7, 512) of a feature map generated from the convolutional layer prior to the FC layer, which is smaller than other CNN structures, such as ResNet and Inception. The proposed system is light weighted using VGGNet, which extracts relatively small size feature maps to build HAR models that process data at high speed. MobileNet, a CNN structure that could be used in embedded systems with relatively low performance...
GPUs and memory, was also proposed. However, the size of the final feature map was (7, 7, 1024), which was not suitable for the purpose of the system to be built in this paper, which requires faster data processing by outputting a larger feature map than the VGGNet used in this paper.

3.3. Integrated Spatial and Temporal Features Based on Weighted Mean

$FS^N$ is a collection of image data including time information sequentially listed according to any interval, and the model extracts $FD^N$, which is the frame change rate of each image included in $FS^N$, as a temporal feature. $N$ means the number of frames that constitute $FS^N$, and since they are extracted according to the same time interval, the frame change rate, $FD^N$, that changes according to the time interval can be used as a temporal feature of motion representation. $FD^N$ is calculated as the average of the frame change sequentially calculated according to the image sequence of $FS^N$ and it is transformed to one-dimensional temporal feature vectors. As compared to optical flow applied to two-stream CNNs, LSTM applied to single-stream CNNs is simpler to operate and it is extracted at high speed. The frame change rate calculated by the frame change of the images constituting $FS^N$ additionally calculates the frame change for the whole frame of the video and the first frame of $FS^N$ because it has the length of $N - 1$. Therefore, $FD^N$ has the same length as the data length $N$, indicating the time information of $FS^N$ extracted according to any interval in the video, and $FD^N$ in $N$ means the frame change rate of $FM^{N-1}_K$ and $FM^K_N$. $FM^K_N$ has a size of $(N, K)$ over the entire time because $K$ one-dimensional vectors are extracted at each time. The data in each row $(N)$ of $FM^K_N$ mean spatial features extracted from images at a specific time and are sorted in a column direction over time. In the proposed system, $FV_K$ with spatiotemporal features is generated by weighted mean of $FD^N$, which represents the degree of change of each frame over time in the spatial feature $FM^K_N$ sorted according to time information, as shown in Equation (1).

$$FV_K = \left[ \frac{\sum_{i=1}^{N} FM^1_K \times FD^i}{\sum_{i=1}^{N} FD^i}, \frac{\sum_{i=1}^{N} FM^2_K \times FD^i}{\sum_{i=1}^{N} FD^i}, \ldots, \frac{\sum_{i=1}^{N} FM^{K-1}_K \times FD^i}{\sum_{i=1}^{N} FD^i}, \frac{\sum_{i=1}^{N} FM^K_K \times FD^i}{\sum_{i=1}^{N} FD^i} \right]$$

(1)

The weighted mean is used to convert the spatial feature map, $FM^K_N$, listed over time into a spatiotemporal feature during the entire time, and the elements of each feature map are averaged by weighting $FD^N$ at the corresponding time according to size $K$ of spatial feature maps. $FD^N$ is a temporal feature extracted from spatial information represented by the motion information of an object and background information, which changes according to the camera’s point of view, so it can be matched with spatial features simultaneously when $FM^K_N$ is extracted. According to Equation (1), the system divides the sum of the result multiplying the $k$th element of each feature map by $FD^N$ at the corresponding time and the sum of $FD^N$ to generate spatiotemporal features $FV$. The elements of feature maps are generated as spatiotemporal features for action recognition through weighted mean calculations considering frame change rate during the entire time. Figure 4 is a result of schematizing the operation process when $K = 1$ and generates $FV_K$ by repeating the operation $K$ times according to the size of the spatial feature map. The weighted mean is weighted to the feature point of video when the motion or background information of an object changes greatly according to the time information of $FD^N$, thereby enabling the context of the spatial feature that changes throughout the whole time. Finally, $FV_K$ generated through the weighted mean is inputted to the FC layer of the MLP structure, and the MLP outputs the class of the predefined videos.
Finally, the proposed HAR model recognizes an action by inputting the spatiotemporal feature vector, $FV_K$, generated by the weighted mean into the MLP. The MLP receiving the $FV_K$ has three hidden layers, the input layer consists of 25,088 nodes, and the output layer consists of 101 nodes, which is the number of predefined classes. Each hidden layer consists of 2048, 1024, and 512 nodes. The batch size set in the input layer of the model is 64, and to prevent overfitting during the learning process, dropout is applied at a ratio of 0.5 in the first hidden layer. The categorical cross-entropy, which is a loss function to classify multiple classes according to the purpose of the system, was selected and adaptive moment estimation (ADAM) was applied to the model.

4. Experimental Results
4.1. Experimental Setup
The performance evaluation results of the proposed weighted mean-based action recognition model are described. The learning and testing of the model proceed with HAR benchmark dataset UCF-101. UCF-101 consists of 13,220 video clips and 101 actionable classes taken on YouTube. The actionable categories are divided into (1) human–object interaction, (2) body-motion only, (3) human–human interaction, (4) playing musical instruments, and (5) sports. Video is a relatively challenging dataset, filmed at various illuminations, poses of people, and viewpoints of cameras. According to the train–test list of provided datasets, 9.5K datasets are divided into learning and 3.8K datasets are divided into tests, and 20% of the learning datasets are used as validation datasets. In addition, the train–test list consists of three scenarios with random order of video data, so the average of three performance evaluations was selected as the final result.

The proposed system was trained at the workstation, and performance evaluations were conducted at the workstation and NVIDIA Jetson NANO. At the workstation, tests were conducted to compare and evaluate the accuracy of the proposed method with the existing HAR models, and at Jetson NANO, the tests were performed to evaluate usability based on throughput of low-cost embedded systems. The model of the proposed system is
built in the Python environment, Keras, and the system environment of the workstation consists of Intel i9-10900X CPU, NVIDIA Titan RTX (24 GB) GPU, and 128 GB of main memory. The system environment of Jetson NANO consists of Quad-core ARM A57 CPU, 128-core Maxwell GPU, and 4 GB of main memory, as shown in Figure 5.

Figure 5. The proposed Jetson NANO-based HAR system appearance.

4.2. Performance Evaluation

The main idea proposed in this paper is to recognize action at a rapid rate through the spatiotemporal feature vector \((FV_k)\) generated by weighted mean of the spatial feature map \((FM_k^N)\) extracted from the frame stack \((FS^N)\) consisting of sequential images with the change rate \((FD^N)\) of each frame representing time information. The performance evaluation is conducted by comparing the action recognition accuracy and complexity of the HAR model using the proposed weighted mean and the existing HAR models. The comparison results of performance evaluation are shown in Table 2, and results are obtained using workstation. Models used for comparison of performance evaluation received RGB stream exactly as the system was built in this paper, and complexity of model and the average accuracy were compared according to the three random scenarios of the train–test split list provided by UCF-101. Equation (2) represents the complexity of each model, and the definition of each factor is defined in Table 3. If the network of the model is 3D CNN, time-axis operation is added to the convolution operation. The recognition accuracy of 3D CNN and single-stream CNN was 84.8%, 85.2%, and 88.1%, respectively, and the recognition accuracy of the model using the LSTM-based method was 90.8% and 91.21%. The LSTM-based action recognition model shows higher accuracy than 3D CNN and single-stream CNN because sequential data are received and weighted to identify connectivity. However, the proposed model based on single-stream CNN recognized action with 2.48% higher accuracy than deep LSTM. This study proved that the proposed method using the change rate of sequential frames as time information can effectively identify the connectivity of motion for action recognition.

\[
N_{CNN} = \sum_{conv} 2C_{k-1}C_k^{(kernel)}N_k + \sum_{pool} C_kN_k\left(N_k^{(pool)} - 1\right) + 2(C_kN_kN_d + N_dN_{out})
\]

\[
N_{LSTM} = L \cdot \sum_{k=1}^K (8(N_{k-1} + N_k)N_k + 4N_k) + 2N_kN_{out}
\]

\[
N_{BILSTM} = L \cdot \sum_{k=1}^K (8(N_{k-1} + N_k)N_k + 4N_k) + 2N_kN_{out}
\]
Table 2. Comparison of average accuracy of the proposed method for UCF-101 with other methods.

| Model           | Network            | Accuracy (%) | Complexity          |
|-----------------|--------------------|--------------|---------------------|
| Multi-scale CNN | 3D CNN             | 84.80        | $N_{CNN}$           |
| C3D [47]        | 3D CNN             | 85.20        | $N_{CNN}$           |
| MSD [48]        | 3D ConvLSTM        | 90.80        | $N_{CNN}+N_{LSTM}$  |
| FCN [49]        | Single-stream CNN  | 88.10        | $N_{CNN}$           |
| Deep LSTM [50]  | 2D ConvLSTM        | 91.21        | $N_{CNN}+N_{BiLSTM}$|
| Ours            | Single-stream CNN  | 93.69        | $N_{CNN}$           |

Table 3. Notation of equations indicating model complexity.

| Symbol | Definition                          |
|--------|-------------------------------------|
| $K$    | Number of hidden layers             |
| $N_k/N_{out}$ | Feature vector length of the $k$th hidden layer/output size |
| $N_k^{(kernel)}$ | Kernel size of the $k$th hidden layer |
| $N_k^{(pool)}$ | Pooling factor in the $k$th hidden layer |
| $C_k$  | Number of filters in the $k$th hidden layer |
| conv/ pool | Set of indices of convolutional/pooling layers |
| $L$    | Input sequence length               |

In addition, additional experiments for performance evaluation were conducted to verify the applicability of the action recognition model on embedded systems equipped with low-cost GPUs. Experiments were conducted by means of “Element-wise Mean,” “Weighted Mean,” “ConvLSTM,” and “Bi-ConvLSTM.” The “Element-wise Mean” means an operation that averages $F_{out}^N$ outputted from CNN according to the number of $N$ without weighting time information.

The frame number ($N$) of $FS^N$ extracted according to a random interval is increased in five intervals from 10 to 40, the action recognition accuracy was measured, and testing time times were compared according to the frame number ($N$) of $FS^N$, and then testing time was measured by the average time it takes to output the action class from each video of the test data set. The action recognition accuracy and testing time of three methods according to $N$ number of $FS^N$ extracted from the video data according to any interval are shown as a graph in Figure 6 and results are obtained using Jetson NANO.

![Figure 6](image_url)

Figure 6. Accuracy and testing time according to frame number of $FS^N$ extracted at a random interval: (a) accuracy and (b) testing time.
First, the comparative evaluation was conducted with the case of applying “Element-wise Mean” to determine whether the weighted mean effectively extracts meaningful spatiotemporal features to recognize action by identifying connectivity of time-varying spatial features. Performance evaluation results show that the proposed method for weighted $FD^N$, in all $N$ measures higher accuracy than “Element-wise Mean” and the weighted mean generates spatiotemporal features to recognize action efficiently. When the spatial and temporal features are fused through unweighted “Element-wise Mean” of $FD^N$, the spatiotemporal features are generated by relying solely on the number of $N$, which is the interval of $FS^N$. Since videos are sequential data with time-varying frames, even in the case of considering only the interval of $FS^N$, the connectivity of the time-varying spatial features can be identified. The motion and background information of an object have nonlinear features rather than constant changes such as the interval of $FS^N$, so if frames depend only on extracted intervals, changes in various environmental conditions cannot be considered. Various environmental conditions include changes in camera’s viewpoints and influences of external environmental factors such as lighting and obstacles, according to data acquisition environment, and accordingly, the scale and pose of people’s appearances change and motion representation changes over time. Therefore, if spatiotemporal features are generated through “Element-wise Mean,” the spatiotemporal features cannot be efficiently generated because spatial representation changes according to the above conditions cannot be identified at a specific time. However, considering the change rate of frames according to the interval extracted through the weighted mean, since big changes in spatial representation are weighed with time information at a specific time and spatiotemporal features are generated, the action recognition performance was improved by considering the motion and background information of objects changing over time effectively.

In addition, as a result of the performance comparison with weighted mean and LSTM-based models, “ConvLSTM” and “Bi-ConvLSTM”, when extracting more than 30 frames, LSTM-based models predict action with high recognition accuracy of 2–3% or more. However, when 20–30 frames are extracted, we can see that the proposed weighted mean-based model predicts action with high recognition accuracy of 20% or more. In particular, when 20 frames are extracted, the proposed weighted mean-based model and LSTM-based model showed the greatest performance difference, and the action prediction accuracy and testing time for videos in each method when $N = 20$ are shown in Table 4 where the tests were conducted in Jetson NANO.

An LSTM prediction network has been proposed to solve the problem of long input data length and disappearing influence on initial inputs in RNNs, which sequentially receive and process time series data. It controls the amount of information transmitted from hidden layers to input gates and oblivion gates. When extracting a large number of frames from video, the amount of data increases and the extracted frame interval shortens, so it is advantageous to identify time-varying spatial features. Therefore, when extracting more than 30 frames, LSTM-based models could recognize action with higher accuracy than weighted mean-based models. In addition, “Bi-ConvLSTM” utilizes information when sequentially listed $FM^N$ is inputted in the reverse direction; “Bi-ConvLSTM” has higher accuracy than “ConvLSTM” when extracting frames number ($N$) increases. This shows that the LSTM-based model shows higher performance as the large number of frames is extracted from the videos.
Table 4. Comparison of action prediction accuracy using four methods, when \( N = 20\).

| Method                | GT: ApplyEyeMakeup | GT: LongJump | GT: Kayaking |
|-----------------------|--------------------|--------------|--------------|
|                       | Accuracy score (%) | Testing time (s) | Accuracy score (%) | Testing time (s) | Accuracy score (%) | Testing time (s) |
| Element-wise Mean     | 65.23              | 0.28         | 68.72        | 0.34           | 67.34             | 0.31           |
| Weighted Mean         | 88.64              | 0.59         | 89.02        | 0.63           | 88.46             | 0.61           |
| ConvLSTM              | 69.25              | 0.89         | 71.87        | 1.05           | 68.53             | 1.26           |
| Bi-ConvLSTM           | 71.24              | 0.91         | 72.53        | 1.08           | 67.71             | 1.38           |

However, as the results of comparing data testing time increase, the proposed weighted mean-based HAR model showed faster data processing speed than LSTM-based models when extracting more than 30 frames. This is because the proposed model recognizes action using MLP, which is lighter than LSTM as a prediction network. LSTM receives time series data sequentially listed over time and controls the amount of information stored while repeating the operation of hidden layers by the length of data. Therefore, when extracting many frames from videos, LSTM has a structure that receives a relatively long time of data input, and the data processing speed is slowed. The data processing speeds of “ConvLSTM” and “Bi-ConvLSTM” in Figure 5 show that the data processing speed is slower as \( N \) increases. However, MLP predicts action at a faster rate than LSTM by conducting forward propagation operations regardless of data length. LSTM receives data sequentially to identify the temporal features of spatial feature maps, which are inputted, and repeats the computation of hidden layers as the frame number \( (N) \). However, the proposed model receives the spatiotemporal features generated by the weighted mean of
$F M^N_k$ to $F D^N$ representing time information at a time as data of the entire time. Jetson NANO, with low-cost GPUs, can recognize action at a faster speed than LSTM. In addition, the weighted mean-based model recognizes action with a high accuracy of 30% or more even when extracting ~10 frames less than LSTM-based models. This shows that the proposed method, which generates spatiotemporal features through weighted mean efficiently, extracts motion representations for action recognition. Therefore, the HAR model can be constructed by extracting relatively few frames and the instantaneous action can be recognized quickly.

5. Conclusions

In this paper, we proposed a weighted mean-based spatiotemporal feature extraction technique to build a CNN-based HAR model that recognizes action by processing video data at high speed. Previously, an action was recognized by analyzing motion patterns based on time information through models with complex structures such as 3D CNN, two-stream CNNs using optical flows, and ConvLSTMs. However, the proposed method recognizes an action by extracting frame changes, which are calculated at high speed with temporal features. The temporal feature is used to generate the spatiotemporal features during the entire time by weighing the spatial features extracted from CNNs and through the weighted mean spatial information at the point where motion changes significantly. The generated spatiotemporal features are used to recognize action by entering into MLPs with lower complexity than the prediction model used in the existing HAR, and the proposed model is verified via experiments to recognize action by processing data at a faster speed than the existing CNN-based HAR models. In addition, the efficiency of extracting spatiotemporal features was verified using the frame change rate as time information through higher action recognition performance compared to general average technique. Performance evaluation results according to the number of frames extracted from videos showed that action was recognized with high accuracy even while extracting fewer frames than the HAR model using LSTM. Finally, to assess real-time possibility in embedded systems of low-cost GPUs, the results of performance evaluation in Jetson NANO also show that data are processed at high speed. It was possible to verify the system’s utilization value to recognize instantaneous action in an emergency situation.

Nevertheless, the proposed model still has a weakness in that it is vulnerable to rapid changes of background or obstacles because it recognizes the action by using the changes in image frames over time. Changes in frames caused by background or obstacles, not actions of a person to be recognized, can act as noise and deteriorate action recognition performance. In particular, it is very difficult to recognize an action in a camera that is constantly moving, not a CCTV, which is a fixed type of camera we have adopted, because the frame is continuously changing. Addressing this issue is the direction of research we are pursuing in the future.

Author Contributions: All authors took part in the discussion of the work described in this paper. Conceptualization, J.K.; Data curation, J.K.; Investigation, J.K.; Project administration, J.C.; Writing—original draft, J.C.; Writing—review & editing, J.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by a National Research Foundation of Korea (NRF) grant funded by the Korean government (MOE) (No. 2018R1D1A3B07041729) and the Soonchunhyang University Research Fund.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Acknowledgments: The authors thank the editor and anonymous reviewers for their helpful comments and valuable suggestions.
Conflicts of Interest: The authors declare that they have no competing interests.

References

1. Qiu, Z.; Yao, T.; Ngo, C.W.; Tian, X.; Mei, T. Learning spatio-temporal representation with local and global diffusion. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–21 June 2019; pp. 12056–12065.

2. Rawat, W.; Wang, Z. Deep convolutional neural networks for image classification: A comprehensive review. Neural Comput. 2017, 29, 2352–2449. [CrossRef]

3. Zhao, Z.-Q.; Zheng, P.; Xu, S.-T.; Wu, X. Object detection with deep learning: A review. IEEE Trans. Neural Netw. Learn. Syst. 2019, 30, 3212–3232. [CrossRef] [PubMed]

4. Wu, Q.; Hu, F.; Zhu, A.; Wang, Z.; Bao, Y. Learning spatial-temporal features via a pose-flow relational model for action recognition. AIP Adv. 2020, 10, 075208. [CrossRef]

5. Liu, A.-A.; Xu, N.; Nie, W.-Z.; Su, Y.-T.; Wong, Y.; Kankanhalli, M. Benchmarking a multimodal and multiview and interactive dataset for human action recognition. IEEE Trans. Cybern. 2016, 47, 1781–1794. [CrossRef] [PubMed]

6. Gao, Z.; Zhang, Y.; Zhang, H.; Xue, Y.B.; Xu, G.P. Multi-dimensional human action recognition model based on image set and group sparsity. Neurocomputing 2016, 215, 138–149. [CrossRef]

7. Wang, L.; Xiong, Y.; Wang, Z.; Qiao, Y.; Lin, D.; Tang, X.; Van Gool, L. Temporal segment networks for action recognition in videos. IEEE Trans. Pattern Anal. Mach. Intell. 2019, 41, 2740–2755. [CrossRef] [PubMed]

8. Leong, M.C.; Prasad, D.K.; Lee, Y.T.; Lin, F. Semi-CNN architecture for effective spatio-temporal learning in action recognition. Appl. Sci. 2020, 10, 557. [CrossRef]

9. Li, S.; Zhao, Z.; Su, F. A spatio-temporal hybrid network for action recognition. In Proceedings of the IEEE Visual Communications and Image Processing (VCIP), Sydney, Australia, 1–4 December 2019; pp. 1–4.

10. Varol, G.; Laptev, I.; Schmid, C. Long-term temporal convolutions for action recognition. IEEE Trans. Pattern Anal. Mach. Intell. 2018, 40, 1510–1517. [CrossRef]

11. Ben-Ari, R.; Shpigel, M.; Azulai, O.; Barzelay, U.; Rotman, D. TAEN: Temporal aware embedding network for few-shot action recognition. arXiv 2020, arXiv:2004.10141.

12. Wang, H.; Song, Z.; Li, W.; Wang, P. A hybrid network for large-scale action recognition from RGB and depth modalities. Sensors 2020, 20, 3305. [CrossRef]

13. Rodriguez-Moreno, I.; Martinez-Otzeta, J.M.; Sierra, B.; Rodriguez, I.; Jauregi, E. Video activity recognition: State-of-the-Art. Sensors 2019, 19, 3160. [CrossRef]

14. Carreira, J.; Zisserman, A. Quo Vadis, action recognition? A new model and the kinetics dataset. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 4724–4733.

15. Diba, A.; Fayyaz, M.; Sharma, V.; Karami, A.H.; Arzani, M.; Yousefzadeh, R.; Gool, L.V. Temporal 3D ConvNets: New architecture and transfer learning for video classification. arXiv 2017, arXiv:1711.08200.

16. Tran, D.; Ray, I.; Shou, Z.; Chang, S.F.; Paluri, M. ConvNet architecture search for spatiotemporal feature learning. arXiv 2017, arXiv:1708.05038.

17. Simonyan, K.; Zisserman, A. Two-stream convolutional networks for action recognition in videos. arXiv 2014, arXiv:1406.2199.

18. Zhao, Y.; Man, K.L.; Smith, J.; Siddique, K.; Guan, S.-U. Improved two-stream model for human action recognition. EURASIP J. Image Video Process. 2020, 2020, 1–9. [CrossRef]

19. Majd, M.; Saafari-Bakhsh, R. A motion-aware ConvLSTM network for action recognition. Appl. Intell. 2019, 49, 2515–2521. [CrossRef]

20. Lee, J.; Ahn, B. Real-time human action recognition with a low-cost RGB camera and mobile robot platform. Sensors 2020, 20, 2886. [CrossRef]

21. Shidik, G.F.; Noersasongko, E.; Nugraha, A.; Andono, P.N.; Jumanto, J.; Kusuma, E.J. A systematic review of intelligence video surveillance: Trends, techniques, frameworks, and datasets. IEEE Access 2019, 7, 170457–170473. [CrossRef]

22. Fakhimeh, R.; Sareh, S.; Upcrofit, B.; Michael, M. Action recognition: From static datasets to moving robots. In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Marina Bay Sands, Singapore, 29 May–3 June 2017; pp. 1–9. [CrossRef]

23. Sreenu, G.; Durai, M.A.S. Intelligent video surveillance: A review through deep learning techniques for crowd analysis. J. Big Data 2019, 6, 48. [CrossRef]

24. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet classification with deep convolutional neural networks. Adv. Neural Inf. Process. Syst. 2012, 25, 1097–1105. [CrossRef]

25. Zhang, H.-B.; Zhang, Y.-X.; Zhong, B.; Lei, Q.; Yang, L.; Du, J.-X.; Chen, D.-S. A comprehensive survey of vision-based human action recognition methods. Sensors 2019, 19, 1005. [CrossRef] [PubMed]

26. Chen, C.; Liu, K.; Kehtarnavaz, N. Real-time human action recognition based on depth motion maps. J. Real-Time Image Process. 2016, 12, 155–163. [CrossRef]

27. Zhang, J.; Liu, W.; Ogunbona, P.O.; Wanga, P.; Tang, C. RGB-D-based action recognition datasets: A survey. Pattern Recognit. 2016, 60, 86–105. [CrossRef]

28. Yang, X.; Tian, Y. Effective 3D action recognition using EigenJoints. J. Vis. Commun. Image Represent. 2014, 25, 2–11. [CrossRef]
29. Oreifej, O.; Liu, Z. HON4D: Histogram of oriented 4D normals for activity recognition from depth sequences. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Portland, OR, USA, 23–28 June 2013; pp. 716–723.

30. Yang, X.; Tian, Y.L. Super normal vector for activity recognition using depth sequences. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, USA, 24–27 June 2014; pp. 804–811.

31. Warcho, D.; Kapuściński, T. Human action recognition using bone pair descriptor and distance descriptor. Symmetry 2014, 12, 1580. [CrossRef]

32. Muralikrishna, S.N.; Muniyal, B.; Acharya, U.D.; Holla, R. Enhanced human action recognition using fusion of skeletal joint dynamics and structural features. J. Robot. 2020, 2020, 3096858. [CrossRef]

33. Yang, Y.; Cai, Z.; Yu, Y.D.; Wu, T.; Lin, L. Human action recognition based on skeleton and convolutional neural network. In Proceedings of the Photonics & Electromagnetics Research Symposium Symposium (PIERS-Symposium), Xiamen, China, 17–20 December 2019; pp. 1109–1112.

34. Cao, Z.; Hidalgo, G.; Simon, T.; Wei, S.E.; Sheikh, Y. OpenPose: Realtime multi-person 2D pose estimation using part affinity fields. IEEE Trans. Pattern Anal. Mach. Intell. 2019, 43, 172–186. [CrossRef]

35. Chaaraoui, A.A.; Padilla-Lopez, J.R.; Florez-Revuelta, F. Fusion of skeletal and silhouette-based features for human action recognition with RGB-D devices. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Portland, OR, USA, 23–28 June 2013; pp. 91–97.

36. Laptev, I. On Space-Time Interest Points. Int. J. Comput. Vis. 2005, 64, 107–123. [CrossRef]

37. Klaser, A.; Marszałek, M.; Schmid, C. A spatio-temporal descriptor based on 3D-gradients. In BMVC 2008-19th British Machine Vision Conference 2008; British Machine Vision Association: Durham, UK, 2008; p. 275.

38. Scovanner, P.; Ali, S. A 3-dimensional sift descriptor and its application to action recognition. In Proceedings of the 15th ACM International Conference on Multimedia: Augsburg, Germany, 24–29 September 2007; pp. 357–360.

39. Yilmaz, A.; Shah, M. Actions sketch: A novel action representation. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR05), San Diego, CA, USA, 20–25 June 2005; pp. 984–989.

40. Ji, S.; Xu, W.; Yang, M.; Yu, K. 3D Convolutional neural networks for human action recognition. IEEE Trans. Pattern Anal. Mach. Intell. 2013, 35, 221–231. [CrossRef]

41. Karpathy, A.; Toderici, G.; Shetty, S.; Leung, T.; Sukthankar, R.; Fei-Fei, L. Large-scale video classification with convolutional neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, USA, 23–28 June 2014; pp. 1725–1732.

42. Si, C.; Chen, W.; Wang, W.; Wang, L.; Tan, T. An attention enhanced graph convolutional LSTM network for skeleton-based action recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–21 June 2019; pp. 1227–1236.

43. Liu, J.; Shahroudy, A.; Xu, D.; Wan, G. Spatio-temporal LSTM with trust gates for 3D human action recognition. In Proceedings of the European Conference on Computer Vision (ECCV), Amsterdam, The Netherlands, 8–16 October 2016; pp. 816–833.

44. Sanchez-Caballero, A.; López-Díz, S.; Fuentes-Jiménez, D.; Losada-Gutiérrez, C.; Marrón-Romera, M.; Casillas-Perez, D.; Sarkar, M.I. 3DFCNN: Real-time action recognition using 3D deep neural networks with raw depth information. arXiv 2020, arXiv:2006.07743.

45. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. arXiv 2014, arXiv:1409.1556.

46. Hara, K.; Kataoka, H.; Satoh, Y. Towards good practice for action recognition with spatiotemporal 3D convolutions. In Proceedings of the 24th International Conference on Pattern Recognition (ICPR), Beijing, China, 20–24 August 2018; pp. 2516–2521.

47. Tran, D.; Bourdev, L.; Fergus, R.; Torresani, L.; Paluri, M. Learning spatiotemporal features with 3D convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 4489–4497.

48. Li, Q.; Qiu, Z.; Yao, T.; Mei, T.; Rui, Y.; Luo, J. Action recognition by learning deep multi-granular spatio-temporal video representation. In Proceedings of the ACM on International Conference on Multimedia Retrieval, Melbourne, Australia, 6–9 June 2016; pp. 159–166.

49. Sun, L.; Jia, K.; Yeung, D.Y.; Shi, B.E. Human action recognition using factorized spatio-temporal convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 4957–4605.

50. Ullah, A.; Ahmad, J.; Muhammad, K.; Sajjad, M.; Baik, S. Action recognition in video sequences using deep bi-directional LSTM with CNN features. IEEE Access 2017, 6, 1155–1166. [CrossRef]