Article

Deep Learning-Based Human Activity Recognition Using Dilated CNN and LSTM on Video Sequences of Various Actions Dataset

Bakht Alam Khan and Jin-Woo Jung *

Posted Date: 15 April 2025

doi: 10.20944/preprints202504.1120.v1

Keywords: human activity recognition; Dilated CNN; LSTM; UCF 50 dataset; spatial-temporal information

Article

Deep Learning-Based Human Activity Recognition Using Dilated CNN and LSTM on Video Sequences of Various Actions Dataset

Bakht Alam Khan and Jin-Woo Jung *

Department of Computer Science and Engineering, Dongguk University, Seoul 04620, Republic of Korea; bakhtalam4687@gmail.com

* Correspondence: jwjung@dongguk.edu

Abstract: Human Activity Recognition (HAR) plays a critical role across various fields, including surveillance, healthcare, and robotics, by enabling systems to interpret and respond to human behaviors. In this research, we present an innovative method for HAR that leverages the strengths of Dilated Convolutional Neural Networks (CNNs) integrated with Long Short-Term Memory (LSTM) networks. The proposed architecture achieves an impressive accuracy of 94.9%, surpassing the conventional CNN-LSTM approach, which achieves 93.7% accuracy on the challenging UCF 50 dataset. The use of dilated CNNs significantly enhances the model's ability to capture extensive spatial-temporal features by expanding the receptive field, thus enabling the recognition of intricate human activities. This approach effectively preserves fine-grained details without increasing computational costs. The inclusion of LSTM layers further strengthens the model's performance by capturing temporal dependencies, allowing for a deeper understanding of action sequences over time. To validate the robustness of our model, we assessed its generalization capabilities on an unseen YouTube video, demonstrating its adaptability to real-world applications. The superior performance and flexibility of our approach suggests its potential to advance HAR applications in areas like surveillance, human-computer interaction, and healthcare monitoring.

Keywords: human activity recognition; Dilated CNN; LSTM; UCF 50 dataset; spatial-temporal information

1. Introduction

Human Activity Recognition (HAR) aims to identify and classify the physical activities performed by an individual or a group of individuals, depending on the specific application context. This recognition process involves understanding and interpreting various movements and actions, which can range from simple daily tasks to more complex activities. For instance, activities such as walking, running, jumping, or sitting can be performed by a single person, each involving distinct patterns of body motion and posture changes. These tasks require the system to capture variations in body movements to accurately detect and classify the activities being performed. HAR systems are designed to extract meaningful information from these movements, making it possible to monitor, analyze, and respond to human behavior in diverse scenarios. This capability is particularly valuable in fields such as healthcare, where monitoring patient mobility can be crucial, or in surveillance, where detecting suspicious actions can enhance security. By accurately recognizing and distinguishing these activities, HAR technologies contribute significantly to improving the responsiveness and efficiency of automated systems in various domains [1,2]. Our model follows a similar approach to Hassan et al. (2024), who utilized a Deep BiLSTM model with transfer-learning-based feature extraction for dynamic human activity recognition [25]. Certain activities are carried out through the movement of specific body parts, such as making gestures with one's hands [3,4]. In some instances, activities involve interacting with objects, such as preparing meals in the kitchen [5,6]. The process of recognizing human activities using deep learning architectures, particularly Convolutional Neural Networks (CNNs), involves multiple critical stages that form a cohesive and complex system. A general framework for a CNN-based activity recognition. The initial phase focuses on the selection and integration of appropriate sensing devices tailored to the specific recognition tasks. Following this, data acquisition is conducted, where edge devices are utilized to capture data from various input sources. This data is then transmitted to a centralized server using communication technologies like Wi-Fi or Bluetooth.

Edge computing plays a crucial role in this architecture by bringing computational and storage capabilities closer to the data source, thus facilitating efficient and real-time processing. This approach ensures that sensors deployed for data collection can transmit information directly to edge servers, which are capable of processing the data promptly. By leveraging edge computing, the system achieves low-latency responses and enhanced performance, making it particularly suitable for applications requiring real-time activity recognition. The authors in [10] conducted an in-depth review of HAR frameworks that specifically utilize accelerometer data. This analysis included discussions on various parameters, such as sampling rates, window sizes, and overlap percentages, which are critical for processing time-series data. The review also shed light on feature extraction techniques and highlighted key factors that influence the effectiveness of HAR systems. Cornacchia et al. [11] emphasized the use of wearable sensors in HAR systems, covering a wide range of sensor types, including pressure sensors, accelerometers, gyroscopes, depth sensors, and hybrid modalities. They classified recent HAR studies based on the sensor data processing techniques employed, particularly those leveraging machine learning algorithms.

Additionally, Beddiar et al. [12] provided a comprehensive survey on the latest advancements in HAR, focusing on significant features like the types of activities recognized, input data formats, validation methods, targeted body parts, and camera viewpoints used in data collection. Their review involved a comparative analysis of state-of-the-art methods based on the diversity of activities they could detect. Furthermore, they offered a detailed overview of vision-based datasets, which are crucial for advancing HAR research by providing standardized benchmarks for model evaluation. Collectively, these studies highlight the rapid advancements in HAR technologies, particularly in leveraging wearable sensors and machine learning techniques. They emphasize the need for continued innovation in sensor fusion, data processing, and feature extraction to develop robust systems capable of real-time and accurate activity recognition across diverse environments.

Moreover, incorporating edge devices reduces the dependency on cloud infrastructure by enabling localized data processing, which is vital for maintaining data privacy and reducing network congestion. Thus, the entire system is designed to handle large-scale, continuous data streams efficiently, which is essential for complex tasks like human activity recognition in dynamic environments.

In recent years, deep learning (DL) algorithms have gained significant traction due to their ability to automatically extract features from complex datasets, including visual or image data sequential time-series data . This indicates the need for manual feature engineering, which is often time-consuming and domain-specific, making DL models highly efficient and adaptable across diverse applications. As a result, DL methods have been widely adopted in areas such as computer vision, natural language processing, and predictive analytics, where high-dimensional data is prevalent. The flexibility and robustness of DL models in capturing intricate patterns have made them a preferred choice for tasks requiring accurate, data-driven insights [13,14].

Convolutional Neural Networks (CNNs) are known to effectively learn hierarchical features, progressing from low-level to high-level representations. Researchers have observed that the features extracted by CNNs tend to outperform traditional handcrafted ones. In recent years, substantial efforts have been dedicated to designing neural networks that can effectively capture spatiotemporal characteristics for human activity recognition. Many studies leverage deep learning approaches in this area, as they enable automated extraction and learning of hierarchical features crucial for behavior analysis. This has led to the development of various systems demonstrating promising outcomes in human activity recognition. Deep Learning (DL) techniques have garnered significant interest due to their impressive performance across diverse domains. It is, therefore, unsurprising that DL-based models have seen a surge in applications for tasks such as identification, prediction, and intention recognition. In particular, Recurrent Neural Networks (RNNs) have achieved notable success in behavior analysis, with the Long Short-Term Memory (LSTM) architecture being the most prevalent. LSTMs, an enhanced form of RNNs, utilize gated memory cells that efficiently manage long-term temporal dependencies, making them especially suitable for understanding sequential and time-dependent data in human behavior analysis. Azher et al. [15] proposed an innovative approach to Human Activity Recognition (HAR) using a distinctive feature descriptor known as the Adaptive Local Motion Descriptor (ALMD), which builds upon the Local Ternary Pattern (LTP). The ALMD effectively captures human motion and appearance within video sequences, utilizing a random forest classifier for recognition. This method was validated on three well-known datasets—KTH, UCF Sports, and UCF-50—demonstrating high accuracy.

Similarly, Luvizon et al. [16] developed a novel HAR technique that leverages a spatio-temporal set of local features. In this approach, the VLAD (Vector of Locally Aggregated Descriptors) algorithm extracts features, while a K-Nearest Neighbors (KNN) classifier is used for action recognition. The method was tested on MSRAction 3D, UT-Kinect Action 3D, and Florence 3D Actions datasets, yielding promising results. Huang et al. [17] introduced an advanced model for multidimensional Human Activity Recognition (HAR) by leveraging image set-based analysis and group sparsity techniques. This method begins with the extraction of dense trajectory features, followed by the construction of a codebook using k-means clustering. The resulting Bag-of-Words (BoW) representation is then utilized alongside the codebook to recognize actions from multiple viewpoints. The effectiveness of this approach was evaluated across three well-known datasets: Northwestern UCLA, IXMAX, and CVS-MV-RGBD-Single. Experimental results demonstrated that this method significantly enhances recognition accuracy and achieves robust performance, especially in scenarios involving complex multi-view action sequences. By focusing on multi-dimensional feature representation, Huang et al.'s model offers a promising solution for recognizing activities across different perspectives and conditions. To enhance the performance of human activity recognition, my proposed approach integrates Convolutional Neural Networks (CNNs) with dilated convolutions alongside Long Short-Term Memory (LSTM) networks. The incorporation of dilated convolutions significantly reduces computational complexity while maintaining a broader receptive field, allowing the model to capture essential spatiotemporal features more effectively. This design not only accelerates the training process but also optimizes resource efficiency, leading to superior performance compared to traditional methods. The synergy of CNNs for spatial feature extraction, combined with LSTMs for handling temporal dependencies, results in more accurate recognition of human activities. The proposed approach demonstrates remarkable improvements in accuracy and training speed, offering a robust alternative to conventional deep learning models in this domain. Lightweight networks used in activity recognition frequently implement grouped convolution techniques. This approach partitions the feature map channels within deep neural networks into distinct groups, effectively lowering computational complexity [24].

2. Literature Review

Several factors, such as background noise, varying viewpoints, and changes in lighting, can significantly affect the environment in which human activities are captured in real-time video. These elements can make it challenging to clearly observe and recognize actions, as they introduce variability that obscures human activities. Traditional activity recognition techniques aim to overcome this issue by extracting specific features from the video data to classify different patterns.

Deep learning approaches, particularly convolutional neural networks (CNNs), provide a more effective solution by automatically learning hierarchical features. This capability allows the system to progressively build complex representations from simpler elements, improving the recognition process. The use of pooling layers and weight-sharing in convolutional architectures helps streamline the network's search space, making it more efficient by leveraging the inherent structure of images. Additionally, pooling operations and weight-sharing mechanisms enhance the model's robustness to changes in scale and spatial variations, enabling more consistent and accurate recognition of activities across different conditions. By doing so, CNNs can effectively handle the challenges posed by real-world video data, leading to more reliable human activity recognition.

2.1. CNN Approach for Human Activity Recognition

Zeiler and Fergus [18] showed that the filters learned by convolutional neural network (CNN) models follow a hierarchical structure. In the early layers, the network detects simple, low-level features like edges and textures. As the data moves through deeper layers, the network identifies more abstract, high-level features, such as shapes and object components. This hierarchical approach not only demonstrates the flexibility of CNNs but also their effectiveness as generalized feature extractors across a wide range of tasks. By progressively enhancing the feature details at each layer, CNNs can build intricate data representations, significantly improving their performance in areas such as human activity recognition. This capability of automatically learning both fundamental and complex patterns from the data reduces the dependence on manually crafted features, making CNNs exceptionally robust for analyzing extensive datasets.

2.2. Deep Convolutional Neural Network for Human Activity Recognition (DCNN)

Granada et al. [19] utilized deep convolutional neural networks (CNNs) to extract video representations directly from raw inputs, such as RGB frames and optical flow fields, training their recognition model in a fully end-to-end fashion. By feeding these CNNs with both types of input data, they generated probability scores for each class and predicted activity labels using a fusion technique. This approach demonstrated superior performance in activity recognition compared to methods that rely solely on handcrafted features or deep learning models using only RGB or optical flow inputs. However, despite the advantages of deep learning, the results did not always surpass handcrafted feature-based methods, particularly due to the limited availability of large-scale RGB activity recognition datasets needed for effective supervised training. The scarcity of comprehensive datasets hampers the full potential of CNNs in this domain, emphasizing the need for more extensive and diverse data to achieve better generalization in recognizing human activities.

2.3. Binary Motion Image Method for HAR

In Dobhal et al. [20] proposed a human activity recognition model that utilizes a unique 2D representation of actions by combining sequences of images into a single image, known as a Binary Motion Image (BMI). In their approach, they first generate binary foreground images using a Gaussian Mixture Model (GMM) to highlight motion areas, which are then combined to form the BMI. A convolutional neural network (CNN) is subsequently trained on these BMIs for effective activity classification. The authors extended their work to include 3D depth maps, where a similar feature extraction method was applied to compute BMIs from depth data. This approach demonstrated the flexibility of the BMI representation, as it efficiently captures spatiotemporal motion patterns, allowing the CNN to learn from both 2D visual data and 3D depth information. By leveraging depth maps, the model was able to enhance recognition accuracy, particularly in scenarios where standard RGB data alone may fall short due to changes in lighting or background noise.

2.4.3. D Convolutional Neural Network for Human Activity Recognition

In Ji et al. [21] introduced an innovative approach to activity recognition by utilizing 3D convolutional networks, which extend traditional 2D convolutions into the temporal domain. Unlike standard 2D CNNs that operate solely on spatial dimensions, 3D convolutional networks employ filters that span both spatial and temporal axes. This allows them to effectively capture spatiotemporal features and motions embedded across consecutive video frames, enabling a deeper understanding of dynamic content. By incorporating temporal information directly into the learning process, these networks can better model the motion patterns essential for human activity recognition. However, for optimal performance, the network requires additional input, such as optical flow, to enhance its training capabilities. Ji et al. [21] demonstrated through experiments that 3D convolutional networks significantly outperform traditional 2D CNNs, particularly in scenarios where understanding motion across frames is crucial. This improvement highlights the potential of

3D convolutions in extracting richer feature representations, making them well-suited for complex video-based applications where both spatial details and temporal dynamics are important for accurate recognition.

2.5. Slow Fusion method for Human Activity Recognition

In Karpathy et al. [22] introduced a method to enhance the temporal awareness of convolutional networks through a technique known as slow fusion. In this approach, the network is provided with multiple adjacent segments of a video, and it processes them using the same set of convolutional layers. By doing this, the network captures temporal information across these video segments, enabling it to learn the patterns of motion and events over time. The network's output for each segment is then processed by fully connected layers to generate a comprehensive video descriptor. Furthermore, Karpathy et al. [22] proposed the use of a multi-resolution approach, where two separate networks are employed, each handling smaller inputs. This method not only improves the accuracy of activity recognition by allowing the network to focus on different resolutions of the video data but also reduces the number of parameters the network needs to learn. By utilizing smaller inputs and processing them in parallel streams, the network becomes more efficient, allowing for faster training and improved generalization across diverse video sequences. This approach significantly boosts the network's ability to recognize actions with higher precision while keeping the computational cost manageable.

2.6.3. DDCNN Approach for Human Activity Recognition

In Liu et al. [23] proposed a novel 3D convolutional deep neural network (3DDCNN) designed to automatically learn spatiotemporal features from raw depth sequences. In their method, the network also integrates a Joint Vector, which is calculated using the position and angle information of skeleton joints, to improve the recognition of human activities. This approach allows the model to capture both the spatial and temporal aspects of human motion, which are essential for activity recognition tasks. One of the key advantages of this method is that the learned feature representation is both time-invariant and viewpoint-invariant. This means that the model is capable of recognizing activities accurately regardless of the time at which they occur or the viewpoint from which the action is captured. As a result, the network can generalize better to different scenarios, making it robust to variations in camera angles and temporal shifts. The method achieves results that are comparable to state-of-the-art techniques, demonstrating its effectiveness in recognizing complex human activities while maintaining a high level of accuracy. This approach highlights the potential of combining depth information with skeleton-based features to improve the robustness and performance of activity recognition systems.

2.7.4. K-Dimensional Per-Segment Descriptors Based on CNN

In Ryoo et al. [24] conducted an evaluation of different temporal pooling strategies for human activity recognition, including average pooling, max pooling, and pooled time series (PoT) with temporal pyramids. These methods were designed to capture the dynamics of a scene over time, helping to improve the recognition of activities in video sequences. The authors focused on generating 4K-dimensional per-segment descriptors based on convolutional neural network (CNN) features, which were effective in representing the entire motion sequence of a video. By using these high-dimensional descriptors, the approach could effectively summarize the temporal changes and overall dynamics of the action, making it less sensitive to noise or irrelevant motion that might appear in the video. The key benefit of this approach is that it enhances the model's ability to distinguish meaningful activity patterns from random noise or background motion, which is particularly useful in real-world scenarios where video data can often be noisy or imperfect. The use of temporal pyramids, in combination with different pooling strategies, also allows the model to capture multi-scale temporal information, thus improving its ability to recognize actions over varying time frames. This work highlights the importance of robust feature representations and effective pooling techniques in addressing challenges such as motion noise and variability in activity recognition.

doi:10.20944/preprints202504.1120.v1

3. Proposed Method

3.1. Dilated Convolution

In Dilated convolutions are a technique used in convolutional neural networks (CNNs) to increase the receptive field without adding extra parameters or computational cost. By introducing gaps between the filter elements, dilated convolutions allow the network to capture larger contextual information, which is useful for tasks like image segmentation and action recognition. Unlike traditional convolutions, where filters slide over the input data in a continuous manner, dilated convolutions space out the filter elements, allowing the network to cover a broader area with the same filter size. The dilation rate controls the gap between elements, and higher rates enable the model to capture larger contextual dependencies. This method is particularly effective in tasks requiring both detailed local information and global context. It helps improve accuracy in complex scenes without increasing the computational load. Additionally, dilated convolutions are useful for sequential data tasks, such as time-series analysis or video recognition, where long-range dependencies need to be captured efficiently. In short, dilated convolutions enhance CNNs by expanding their receptive field, improving performance in various tasks without increasing computational complexity.

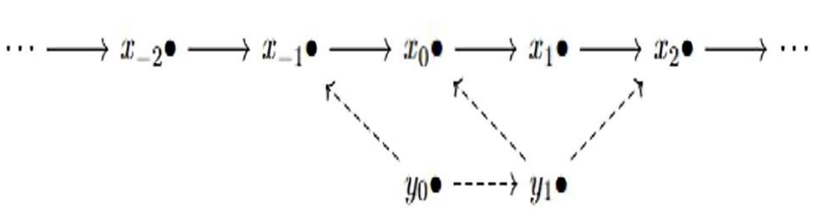


Figure 1. Dilated convolutions representation .

Above figure 1 shows the visual representation of a 1D dilated convolution with dilation rate two. The red dots represent the input values, and the blue dot represents the output value. The dashed lines indicate the connections between the input and output values.

$$y(i) = k\sum x(i + r \cdot k) \cdot w(k). \quad (1)$$

3.2. LSTM (Many to One)

The Many-to-One Long Short-Term Memory (LSTM) network processes sequential data by taking a series of inputs over multiple time steps and producing a single output at the end. In this approach, the input sequence is fed into the LSTM cells one step at a time, allowing the model to learn and retain dependencies through its internal memory mechanism as cn be seen in Figure 2. The LSTM cells are equipped with gates that control the flow of information, enabling them to selectively remember or forget details from earlier time steps. As the model progresses through the sequence, it accumulates relevant information, storing it in its hidden states. Once all inputs have been processed, the final hidden state is used to generate the output, effectively summarizing the entire sequence into one prediction or classification. This architecture is particularly beneficial for tasks where understanding the context of the entire sequence is crucial, such as sentiment analysis, where a complete sentence determines the sentiment, or human activity recognition, where a series of video frames must be analyzed to classify an action. By leveraging its ability to capture both short-term and long-term patterns, Many-to-One LSTM models have proven effective in fields like time series forecasting, text classification, and sequence analysis.

doi:10.20944/preprints202504.1120.v1

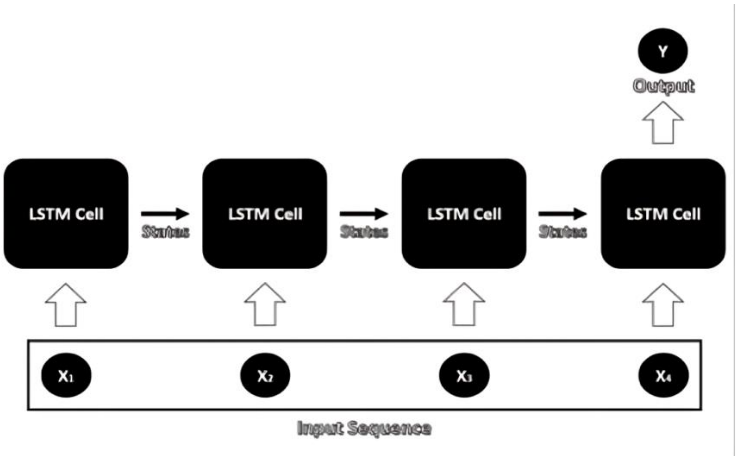


Figure 2. Many to one LSTM representation .

3.3. Workflow for Human Activity Recognition using Dilated ConvLSTM

The first step in the process involves loading a dataset of videos, such as the UCF50 dataset, which contains video sequences of various actions. These videos are stored locally and are accessed by the system for action recognition. Each video in the dataset is processed by extracting frames. These frames are resized to a fixed size and normalized, so they can be consistently used as input to the model.

The next step is to prepare these frames into sequences of a fixed length, as the model requires sequential data for action recognition. Each sequence represents a segment of the video where a specific action is being performed. The frames within a sequence are passed through the model, which uses a combination of dilated convolutional neural networks (CNN) and Long Short-Term Memory (LSTM) networks. The CNN layers extract spatial features from the frames, helping the model recognize objects and structures within the video, while the LSTM layers capture the temporal dependencies between frames, allowing the model to understand how actions unfold over time. The model is trained on these labeled sequences from the dataset, learning to classify actions based on the patterns it detects in the frame sequences. Once the model is trained, it can make predictions on new video data. For prediction, the model takes a new video, extracts frames, and organizes them into sequences. The model then predicts the action for each sequence, labeling the action being performed in the video as described in Figure 3.

doi:10.20944/preprints202504.1120.v1

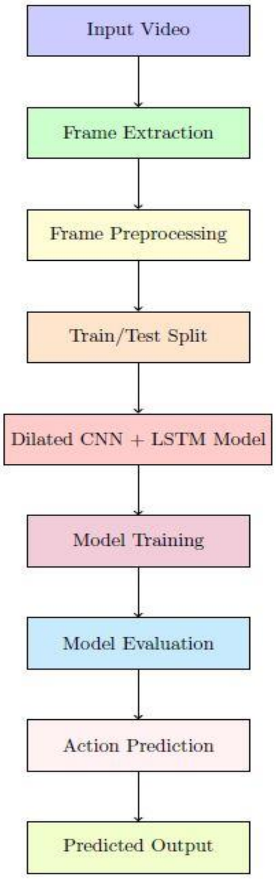


Figure 3. Workflow for Human Activity Recognition Uding Dilated CNN LSTM.

3.4. Proposed Model

This model is a deep neural network that I designed for action recognition using ConvLSTM (Convolutional Long Short-Term Memory) layers, which combine the strengths of Convolutional Neural Networks (CNNs) and LSTMs for processing spatiotemporal data as shown in model Figure 4. The model processes video frames, where the temporal sequence of images is crucial to recognizing patterns in motion. The model begins with an input layer that accepts a sequence of video frames, with each frame having a fixed height, width, and three-color channels (RGB). The sequence length, image height, and width are specified as part of the input shape. The first layer in the model is a ConvLSTM2D layer, which applies a 2D convolutional operation to each frame while maintaining temporal dependencies across frames via LSTM cells. The layer uses 4 filters with a kernel size of 3x3 and a dilation rate of (2,2), meaning that the receptive field is enlarged, allowing the model to capture more spatial context within each frame. This is particularly useful for recognizing large-scale motion across frames. Following the ConvLSTM layer is a MaxPooling3D layer, which down-samples the spatial dimensions (height and width) of the feature maps by applying a 2x2 pool on the height and width dimensions, while keeping the temporal dimension intact. This helps reduce computational complexity and prevents overfitting by emphasizing the most important features. Additionally, Dropout is applied with a rate of 0.2 to regularize the model and reduce the risk of overfitting by randomly deactivating 20% of the neurons during training. The model then passes through a series of similar layers: another ConvLSTM2D, MaxPooling3D, and Dropout layers. The second ConvLSTM2D layer uses 8 filters, followed by a third with 14 filters, and a fourth with 16 filters. The dilation rate remains consistent at (2,2) in all ConvLSTM layers, allowing the model to maintain its ability to capture a large spatial context in each frame. MaxPooling3D layers are used after each ConvLSTM layer to gradually reduce the spatial dimensions. After the fourth ConvLSTM layer, the model applies a Flatten layer to reshape the multi-dimensional output into a single vector that can be processed by the dense layers. Finally, the model includes a Dense layer with a Softmax activation function. The number of units in the Dense layer corresponds to the number of possible action classes in the dataset, and the Softmax activation ensures that the output values represent a probability distribution over these classes. The model is optimized using a categorical cross-entropy loss function, which is suitable for multi-class classification problems. After the Flatten layer, the model incorporates a fully connected Dense layer with 128 units and a ReLU activation function to introduce non-linearity and enhance feature representation. This layer serves as a bridge between the high-level spatiotemporal features extracted by the ConvLSTM layers and the final classification layer. To further mitigate overfitting, another Dropout layer with a rate of 0.3 is added after the Dense layer, ensuring that the model generalizes well to unseen data. Batch normalization is also applied before the final Dense layer to stabilize and accelerate the training process by normalizing the inputs. The model is trained using the Adam optimizer, which adapts the learning rate dynamically, improving convergence and performance. Early stopping is implemented during training to halt the process if the validation loss does not improve for a specified number of epochs, preventing unnecessary computation and overfitting. Additionally, data augmentation techniques such as random cropping, flipping, and rotation are applied to the input video frames to increase the diversity of the training data and improve the model's robustness. Finally, the model's performance is evaluated using metrics such as accuracy, precision, recall, and F1-score to ensure a comprehensive understanding of its classification capabilities across different action classes.

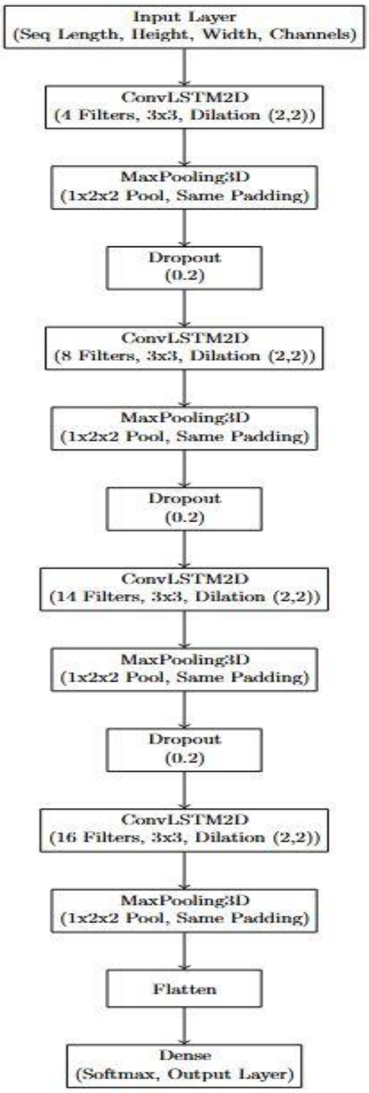

Input Layer
(Seq Length, Height, Width, Channels)
ConvLSTM2D
(4 Filters, 3x3, Dilation (2,2))
MaxPooling3D
(1x2x2 Pool, Same Padding)
Dropout
(0.2)
ConvLSTM2D
(8 Filters, 3x3, Dilation (2,2))
MaxPooling3D
(1x2x2 Pool, Same Padding)
Dropout
(0.2)
ConvLSTM2D
(14 Filters, 3x3, Dilation (2,2))
MaxPooling3D
(1x2x2 Pool, Same Padding)
Dropout
(0.2)
ConvLSTM2D
(16 Filters, 3x3, Dilation (2,2))
MaxPooling3D
(1x2x2 Pool, Same Padding)
Flatten
Dense
(Softmax, Output Layer)

Figure 4. Proposed Dilated CNN LSTM for Spatio-Temporal Features Extraction.

3.4.1. Spatio-Temporal Feature Extraction:

The ConvLSTM2D layers can capture both spatial features (from each individual frame) and temporal features (from the sequence of frames). This is crucial for understanding human activities, which often involve patterns that unfold over time (e.g., walking, running, waving).

3.4.2. The use of Dilation Rate:

By setting the dilation rate to 2, the model can cover larger areas within the video frames, helping it learn high-level temporal patterns. This is beneficial for recognizing complex, long-range actions or transitions between activities that might span several frames.

3.4.3. Handling Sequential Data:

Human activity recognition typically involves videos where each frame contains part of the activity, and the full context emerges across the sequence. ConvLSTM is ideal for handling such data because it integrates convolutional operations (which capture spatial features) with LSTM cells (which capture the temporal dependencies between frames).

3.4.4. Regularization (Dropout):

The use of dropouts in both the ConvLSTM layers and Time Distributed dropout layers helps prevent overfitting, which is important when training on large video datasets, ensuring that the model generalizes well to unseen data.

3.4.5. Hierarchical Feature Learning:

The model extracts increasingly complex features at each layer, which allows it to understand the fine details of actions as well as the broader patterns in the activity. For example, the first layer might detect simple features like edges, the second might capture textures or motion, and the later layers can represent more abstract actions like walking or waving.

3.4.5. Training strategy for the Dilated Conv LSTM model

For training the Dilated ConvLSTM model, We utilized the Google Colab Pro environment, taking advantage of its high-performance GPU and extended memory. Specifically, the setup included a GPU with 15 GB of dedicated memory and 51 GB of RAM, allowing efficient handling of large video datasets and faster training times. To extract spatial-temporal features, ConvLSTM2D layers with a dilation rate of 2 were utilized, allowing the model to capture a broader context without increasing the parameter count. The activation function used for the ConvLSTM layers was tanh, providing smooth gradient flow and capturing intricate patterns in sequential data. Regularization techniques, such as recurrent dropout and Time Distributed dropout, were incorporated to prevent overfitting by randomly deactivating neurons during training. The model was trained for 70 epochs, with batch normalization and validation monitoring in place to ensure generalization to unseen data. The final Dense layer utilized a softmax activation function to output the probabilities for each class. The training process was conducted iteratively, adjusting hyperparameters and employing early stopping to achieve optimal performance on the validation set.

4. Results and Discussions

4.1. Dataset UCF 50

The UCF50 dataset is a comprehensive action recognition dataset that features 50 distinct action categories, sourced from realistic videos on YouTube. It is an expanded version of the earlier YouTube Action dataset (UCF11), which focused on only 11 action categories. Unlike many existing datasets in the field, which rely on staged and controlled environments, the UCF50 dataset emphasizes realism as can be seen in Figure 5. It provides the computer vision community with challenging, real-world scenarios to test the robustness of action recognition models. One of the primary strengths of this dataset is its diversity and complexity. The videos exhibit significant variations in factors such as camera motion, object appearance, pose, scale, and viewpoint, alongside varying illumination conditions and cluttered backgrounds. These variations make the dataset highly challenging and suitable for evaluating models' performance in real-world scenarios. The dataset is organized into 50 action categories, which are further divided into 25 groups per category. Each group contains at least four video clips, often sharing common traits like the same individual, similar backgrounds, or comparable viewpoints, thereby testing models' ability to generalize across different contexts. The 50 action classes span a wide range of activities, including both sports and everyday actions. Some of the notable categories are Basketball Shooting, Bench Press, Billiards Shot, Drumming, Horse Riding, Kayaking, Pull-Ups, Tennis Swing, Volleyball Spiking, and Walking with a Dog, among others. This diversity in actions, captured in realistic settings, makes UCF50 a valuable resource for advancing research in human activity recognition.

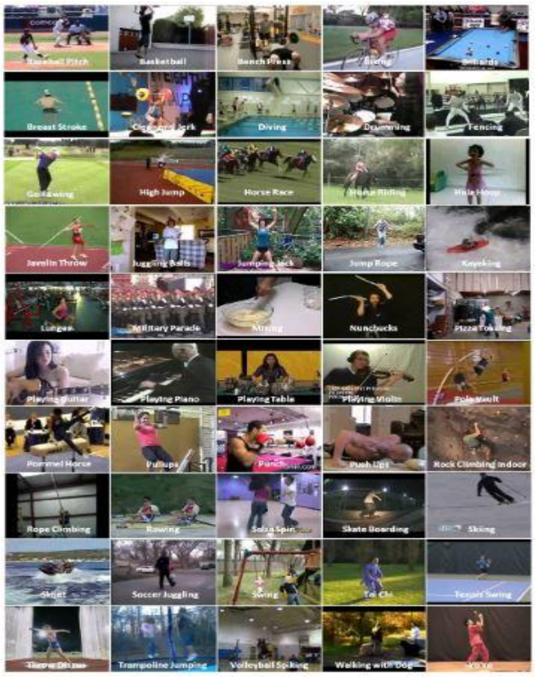


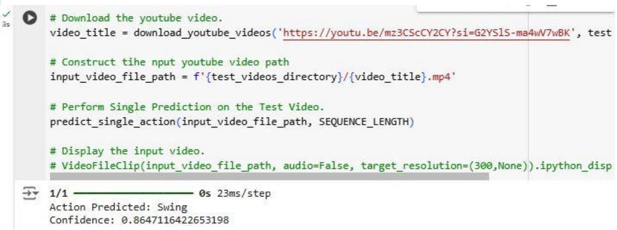
Figure 5. Dataset UCF 50 for human activity recognition.

4.2. Dataset Preprocessing

The pre-processing of the UCF50 dataset involved several essential steps to prepare the data for effective training of the ConvLSTM model. Initially, a subset of action categories was chosen from the dataset, focusing on classes like WalkingWithDog, TaiChi, Swing, and others. For each selected class, the script identified all available video files within the designated directory. The core of the pre-processing was handled by the frames extraction function, which read each video using OpenCV's VideoCapture. It determined the total frame count and extracted a fixed number of frames sequence length was set to 20 at uniform intervals to ensure consistency. To achieve this, a frame sampling interval was computed based on the video's total frame count, allowing the model to learn temporal patterns efficiently. Each extracted frame was resized to a standard dimension of 256x256 pixels to maintain uniformity across the dataset. The frames were then normalized by scaling pixel values to a range between 0 and 1, which enhances training performance by stabilizing gradient updates. The create dataset function brought everything together by processing each video file, extracting frames, and storing them in structured lists. Only videos with at least 20 frames were included, ensuring all sequences were of consistent length. The resulting frames, labels, and video paths were then converted into numpy arrays for efficient handling during model training. This comprehensive pre-processing strategy ensured the data was optimally formatted, allowing the ConvLSTM model to focus on learning the spatial and temporal patterns critical for accurate action recognition.

4.4. Model Evaluation Performance Metrics

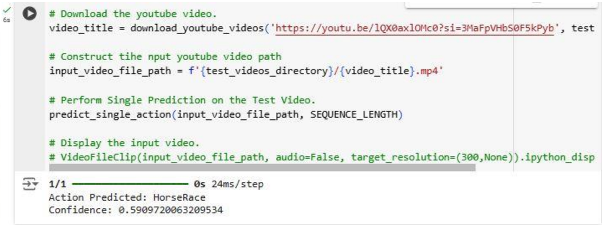
After the model has been trained, testing it on real-world data, such as YouTube videos, involves several key steps. First, the video to be tested is obtained by downloading it from YouTube. This is done by providing the URL of the YouTube video, which is then fetched and stored in a designated directory. The title of the video is extracted, and the video is saved with a meaningful filename. The downloaded video is now ready to be used for testing the model. Once the video is downloaded, it is prepared for action recognition. To do this, the video is read frame by frame, and specific frames are selected based on the sequence length required by the model. These frames are resized to fit the input dimensions expected by the model, and they are normalized to ensure that pixel values lie between 0 and 1. After preprocessing, a fixed number of frames are passed as input to the trained LRCN model. The model processes these frames, generating a probability distribution over the possible classes for each frame sequence. The class with the highest probability is selected as the predicted action. Finally, the predicted action, along with the confidence score (the probability of the predicted class), is output. This gives insight into the model's performance on the test video. The process concludes by displaying the results, typically in the form of the predicted action label and its associated confidence, which reflects how certain the model is about its prediction. This approach allows for the effective evaluation of the model's performance on unseen video data, providing a real-world application of the trained action recognition model. Furthermore, the ability to test on varied and complex real-world video data demonstrates the robustness and versatility of the model in handling different scenarios. This real-time action recognition capability can be applied in a wide range of industries, from surveillance and security to sports and entertainment, highlighting its potential for practical use beyond controlled datasets.



```
# Download the youtube video.
video_title = download_youtube_videos('https://youtu.be/mz3CScCY2CY?si=G2YSlS-ma4wV7wBK', test

# Construct tihe nput youtube video path
input_video_file_path = f'{test_videos_directory}/{video_title}.mp4'

# Perform Single Prediction on the Test Video.
predict_single_action(input_video_file_path, SEQUENCE_LENGTH)

# Display the input video.
# VideoFileClip(input_video_file_path, audio=False, target_resolution=(300,None)).ipython_disp
```

```
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 23ms/step
Action Predicted: Swing
Confidence: 0.8647116422653198
```

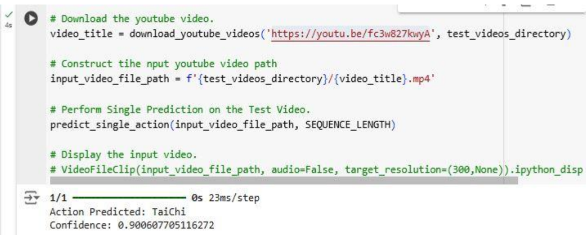
) Test Video (b) Predicted Action with Confidence



```
# Download the youtube video.
video_title = download_youtube_videos('https://youtu.be/lQX0axlOMc0?si=3MaFpVHbS0F5kPyb', test

# Construct tihe nput youtube video path
input_video_file_path = f'{test_videos_directory}/{video_title}.mp4'

# Perform Single Prediction on the Test Video.
predict_single_action(input_video_file_path, SEQUENCE_LENGTH)

# Display the input video.
# VideoFileClip(input_video_file_path, audio=False, target_resolution=(300,None)).ipython_disp
```

```
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 24ms/step
Action Predicted: HorseRace
Confidence: 0.5909720063209534
```

(c) Test Video (d) Predicted Action with Confidence



```
# Download the youtube video.
video_title = download_youtube_videos('https://youtu.be/fc3w827kwyA', test_videos_directory)

# Construct tihe nput youtube video path
input_video_file_path = f'{test_videos_directory}/{video_title}.mp4'

# Perform Single Prediction on the Test Video.
predict_single_action(input_video_file_path, SEQUENCE_LENGTH)

# Display the input video.
# VideoFileClip(input_video_file_path, audio=False, target_resolution=(300,None)).ipython_disp
```

```
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 23ms/step
Action Predicted: TaiChi
Confidence: 0.900607705116272
```

(e) Test Video (f) Predicted Action with Confidence

Figure 6. Results Showcasing model performance on unseen video data.

The model was tested on three different YouTube videos, each depicting a distinct action: swinging, a horse race, and Tai Chi. For the first test, figure 6(a) which involved a video of a person swinging, the model successfully identified the activity as "Swing" as shown in fig 6(b). This prediction showcased the model's ability to recognize repetitive and cyclic movements that are characteristic of swinging motions.

In the second test, the video featured figure Fig 6(c) a horse race, and the model correctly predicted the action as "Horse Race." This result highlighted the model's proficiency in recognizing high-speed activities that involve multiple subjects moving in dynamic ways. The model accurately interpreted the rapid movements of the horses and classified the video accordingly as can be seen in Fig 6(d).

The final test involved a video as in figure 6(e) demonstrating Tai Chi, a martial art known for its slow and controlled movements. The model was able to accurately classify the action as "Tai Chi," which demonstrated its capacity to identify subtle and fluid motion patterns as shown in fig 6(f). This ability is essential for recognizing actions that do not involve abrupt movements but rather require a keen understanding of slower, deliberate gestures. In all three cases, the model not only identified the correct action but also provided a confidence score, reflecting its certainty in each prediction. These results underline the effectiveness of the Dilated ConvLSTM model in performing action recognition on diverse types of activities, ranging from rapid motions to slow, controlled movements.

Table 1. Performance metrics of the proposed (Dilated ConvLSTM).

Model	Accuracy	Validation Accuracy	Loss	Validation Loss
Conv2D	84.23%	75.32%	0.54	0.75
CNN	79.3%	71.9%	0.62	0.73
CNN (Slow Fusion)	85.21%	79.4%	0.45	0.64
LRCN	86.3%	80.5%	0.39	0.52
ConvLSTM	93.7%	86.67%	0.31	0.48
Dilated ConvLSTM (Ours)	**94.9%**	**88.34%**	**0.20**	**0.32**

The incorporation of dilated convolutions in the CNN layers enables the model to expand its receptive field, allowing it to capture more extensive spatial patterns without a corresponding increase in computational cost. This improvement enhances its ability to recognize complex spatial and temporal features across a larger context. Additionally, by integrating Long Short-Term Memory (LSTM) units, the model is able to effectively learn and predict dynamic sequential dependencies, making it well-suited for action recognition tasks. This combination of dilated convolutions and LSTMs results in superior performance compared to conventional CNN and CNN-LSTM models.

4.5. Plotting Accuracy vs. Validation Accuracy

The Total Accuracy vs Validation Accuracy graph as shown in Figure 7 for the Dilated ConvLSTM model clearly demonstrates its superior performance. The blue curve represents total accuracy (training accuracy) over epochs. The red curve represents validation accuracy over epochs.As seen in the graph, the model maintains a significantly higher training accuracy (94.9%) and validation accuracy (88.34%) compared to the other models. This indicates not only strong learning on the training data but also excellent generalization to unseen data, which is crucial for real-world applications. The close alignment between training and validation accuracy further supports the model's robustness and its ability to effectively handle action recognition tasks across diverse inputs.

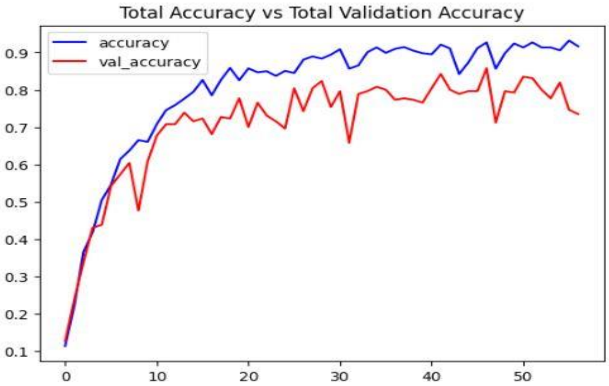


Figure 7. Plotting accuracy vs validation accuracy.

4.6. Plotting Loss vs. Validation Loss

The total Loss vs. Validation Loss graph as shown in Figure 8. The blue curve represents total loss (training loss) over epochs. The red curve represents validation loss over epochs.The Dilated ConvLSTM model highlights its efficient training process. As depicted, the model achieves a low training loss of 0.20 and a validation loss of 0.32, both of which are significantly lower than those of the other models. This shows that the model not only minimizes error during training but also maintains a good balance between training and validation, suggesting it is not overfitting. The smaller gap between training and validation loss indicates strong generalization, which is essential for ensuring the model's effectiveness in real-world action recognition tasks.

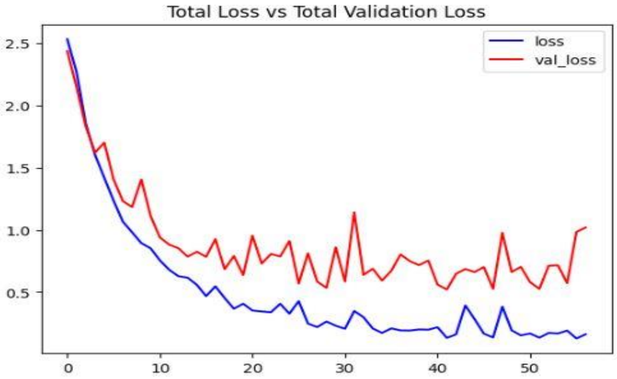


Figure 8. Plotting loss vs validation loss.

5. Conclusions

This project effectively demonstrates the application of a Dilated CNN-LSTM (Long Short-Term Memory) model for recognizing actions in video sequences. By integrating dilated convolutional layers with LSTM units, the model captures both spatial and temporal dynamics, enabling it to identify complex actions over time. The model was trained on a varied set of video data, achieving strong performance in action classification. During the testing phase, the model was successfully applied to real-world YouTube videos, accurately predicting actions. This highlights the model’s robustness and its ability to generalize across different video scenarios. Additionally, the process of downloading, preprocessing, and evaluating the model on previously unseen data, along with its assessment based on accuracy and loss metrics, further illustrates the model's practical applicability in real-world settings. Future research could focus on enhancing the model's performance by refining the architecture, incorporating larger, more diverse datasets, or adapting it for real-time use. Overall, the findings suggest that the Dilated CNN-LSTM model holds considerable promise for a variety of applications, such as surveillance, entertainment, and human-computer interaction.

6. Future Work

For future work, We plan to develop a real-time human surveillance system that leverages the trained Dilated CNN-LSTM model. This system will take input data directly from CCTV cameras, enabling the real-time prediction of actions performed by individuals. By integrating the model into a live surveillance environment, the system will be capable of continuously analyzing video feeds to detect and classify various human actions, such as walking, running, or suspicious behavior. This advancement will enhance the effectiveness of security systems, offering automated monitoring and timely alerts for potential security threats.

Author Contributions: Conceptualization, Bakht Alam Khan; Methodology, Bakht Alam Khan; Validation, Bakht Alam Khan and J.-W.J.; Formal analysis, Bakht Alam Khan; Investigation, Bakht Alam Khan; Resources, Bakht Alam Khan and J.-W.J.; Data curation, Bakht Alam Khan; Writing—original draft, Bakht Alam Khan; Writing—review & editing, Bakht Alam Khan and J.-W.J.; Visualization, Bakht Alam Khan and J.-W.J.; Supervision, J.-W.J.; Project administration, J.-W.J.; Funding acquisition, J.-W.J. All authors have read and agreed to the published version of the manuscript.

Funding: This research was financially supported by the Ministry of Trade, Industry and Energy(MOTIE) and Korea Institute for Advancement of Technology(KIAT) through the International Cooperative R&D program (Project No. P0026318), by the MSIT(Ministry of Science and ICT), Korea, under the ITRC(Information Technology Research Center) support program(IITP-2025-2020-0-01789) supervised by the IITP(Institute for Information & Communications Technology Planning & Evaluation), and by the Artificial Intelligence Convergence Innovation Human Resources Development (IITP-2025-RS-2023-00254592) supervised by the IITP(Institute for Information & Communications Technology Planning & Evaluation).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data is contained within the article.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. [1] S. Qiu, H. Zhao, N. Jiang, Z. Wang, L. Liu, Y. An, H. Zhao, X. Miao, R. Liu,
2. G. Fortino, Multi-sensor information fusion based on machine learning for real
3. applications in human activity recognition: state-of-the-art and research
4. challenges, Inf. Fusion 80 (2022) 241–265, https://doi.org/10.1016/j.
5. inffus.2021.11.006.
6. [2] K. Chen, D. Zhang, L. Yao, B. Guo, Z. Yu, Y. Liu, Deep learning for sensor-based
7. human activity recognition, ACM Comput. Surv. 54 (2021) 1–40, https://doi.org/
8. 10.1145/3447744.
9. [3] S. Jiang, P. Kang, X. Song, B. Lo, P.B. Shull, Emerging wearable interfaces and
10. algorithms for hand gesture recognition: a survey, IEEE Rev. Biomed. Eng. 15
11. (2022) 85–102, https://doi.org/10.1109/RBME.2021.3078190.
12. [4] P. Das Sakshi, S. Jain, C. Sharma, V. Kukreja, Deep learning: an application
13. perspective, in: Lect. Notes Networks Syst., 2022, pp. 323–333, https://doi.org/
14. 10.1007/978-981-16-4284-5_28.
15. [7] F. Alshehri, G. Muhammad, A comprehensive survey of the Internet of things
16. (IoT) and AI-based smart healthcare, IEEE Access 9 (2021) 3660–3678, https://
17. doi.org/10.1109/ACCESS.2020.3047960.
18. [8] D.V. Medhane, A.K. Sangaiah, M.S. Hossain, G. Muhammad, J. Wang, Blockchainenabled distributed security framework for next-generation IoT: an edge cloud
19. and software-defined network-integrated approach, IEEE Internet Things J. 7
20. (2020) 6143–6149, https://doi.org/10.1109/JIOT.2020.2977196.
21. [9] P. Kumari, L. Mathew, P. Syal, Increasing trend of wearables and multimodal

interface for human activity monitoring: a review, Biosens. Bioelectron. 90 (2017) 298–307, https://doi.org/10.1016/j.bios.2016.12.001.

[10] U. Alrazzak, B. Alhalabi, A Survey on Human Activity Recognition Using Accelerometer Sensor, in: 2019 Jt. 8th Int. Conf. Informatics, Electron. Vis. 2019 3rd Int. Conf. Imaging, Vis. Pattern Recognit., IEEE, 2019, pp. 152–159, https://doi.org/10.1109/ICIEV.2019.8858578.

[11] M. Cornacchia, K. Ozcan, Y. Zheng, S. Velipasalar, A survey on activity detection and classification using wearable sensors, IEEE Sensor. J. 17 (2017) 386–403, https://doi.org/10.1109/JSEN.2016.2628346.

[12] D.R. Beddiar, B. Nini, M. Sabokrou, A. Hadid, Vision-based human activity recognition: a survey, Multimed. Tool. Appl. 79 (2020) 30509–30555, https://doi.org/10.1007/s11042-020-09004-3.

[13] V. Sharma, M. Gupta, A.K. Pandey, D. Mishra, A. Kumar, A review of deep learning-based human activity recognition on benchmark video datasets, Appl. Artif. Intell. 36 (2022) 2093705, https://doi.org/10.1080/08839514.2022.2093705.

[14] S. Zhang, Y. Li, S. Zhang, F. Shahabi, S. Xia, Y. Deng, N. Alshurafa, Deep learning in human activity recognition with wearable sensors: a review on advances, Sensors 22 (2022) 1476, https://doi.org/10.3390/s22041476.

[15] Md Azher Uddin, Joolekha Bibi Joolee, Aftab Alam, Young-Koo Lee, Human action recognition using adaptive local motion descriptor in spark, IEEE Access 5 (2017) 21157–21167.

[16] Diogo Carbonera Luvizon, Hedi Tabia, David Picard, Learning features combination for human action recognition from skeleton sequences, Pattern Recognit. Lett. 99 (2017) 13–20.

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

[18] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks", Proc. European Conference on Computer Vision (ECCV)., pp. 818-833, 2014.

[19] Roger Granada, Juarez Monteiro, Rodrigo C. Barrosy and Felipe Meneguzziy, "A Deep Neural Architecture for Kitchen Activity recognition", Association for the advancement of Artificial Intelligence, 2017.

[20] Tushar Dobhal et al., "Human activity recognition using binary motion image and deep learning", Procedia Computer Science, vol. 58, pp. 178-185, 2015.

[21] S. Ji, W. Xu, M. Yang and K. Yu, "3d convolutional neural networks for human action recognition", IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 35, no. 1, pp. 221-231, Jan 2013.

[22] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar and L. Fei-Fei, "Large-scale video classification with convolutional neural networks", Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)., pp. 1725-1732, 2014.

[23] Zhi Liua, Chenyang Zhang and Yingli Tian, "3D-based Deep Convolutional Neural Network for Action Recognition with Depth Sequences", Image and Vision Computing, 2016.

[24] M. S. Ryoo and Larry Matthies, "Video-based convolutional neural networks for activity recognition from robot-centric videos", Proc. of SPIE, vol. 9837, no. 98370R-1, 2016.

[24] Lin, L.; Wu, J.; An, R.; Ma, S.; Zhao, K.; Ding, H. LIMUNet: A Lightweight Neural Network for Human Activity Recognition Using Smartwatches. Appl. Sci. 2024, 14, 10515. https://doi.org/10.3390/app142210515.

[25] Hassan, N.; Miah, A.S.M.; Shin, J. A Deep Bidirectional LSTM Model Enhanced by Transfer-Learning-Based Feature Extraction for Dynamic Human Activity Recognition. *Appl. Sci.* **2024**, *14*, 603. https://doi.org/10.3390/app14020603.

doi:10.20944/preprints202504.1120.v1

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.