

Action Recognition in Videos Using Block-Based CNN Method

Sabahat Tanveer

sabahattanveer6@gmail.com

Department of Computer Science & Information Technology, Mirpur University of Science & Technology (MUST), Mirpur AJK, Pakistan

Syed Yasser Arafat

syed.yasser.arafat@gmail.com

Department of Computer Science & Information Technology, Mirpur University of Science & Technology (MUST), Mirpur AJK, Pakistan

Abdul Ghafoor

ghafoor.csit@must.edu.pk

Department of Computer Science & Information Technology, Mirpur University of Science & Technology (MUST), Mirpur AJK, Pakistan

Corresponding Author: Sabahat Tanveer sabahattanveer6@gmail.com

Received: 02-02-2026

Revised: 26-02-2026

Accepted: 04-03-2026

Published: 20-03-2026

ABSTRACT

Human Action Recognition is an important research area in human behaviour analysis. A reliable action recognition system has many important applications in various fields of computer vision including robotics and human-computer interaction. While researchers have performed analysis of different human actions like dancing, clapping, handwaving, robust recognition remains a significant challenge for machine learning algorithms. Recognizing actions from images and videos is extremely difficult task due to a different kind of actions occurring under different backgrounds. Specific challenges include background clutter, occlusion, camera motion, variations in execution rate, and lighting conditions. To overcome such challenges, we have proposed block-based CNN methodology designed to recognize actions from video data effectively. Our approach extracts temporal features using a CNN model configured with specific temporal block sizes. The proposed model was trained and evaluated using two different block configurations (size 11 and size 5) across five different datasets: KTH, UCF101, UCF Sports Action, YouTube, and MUSTian Action dataset. Using an 80/20 train-test holdout ratio, our Model III achieved near-perfect accuracy, with training and validation performance reaching up to 100% on controlled benchmarks while mainlining competitive results on more complex, real-world datasets.

Keywords : Action Recognition, CNN, LSTM, Feature Extraction.

INTRODUCTION

Human action recognition is one of the most important research areas in computer vision [1]. The primary goal of this field is the automated classification of ongoing actions from video data [2]. More specifically, it enables a machine to recognize and interpret the activities performed by people, as well as their interactions with objects and the surrounding environment [3]. As a critical challenge in computer vision, robust action recognition facilitates significant applications across diverse domains, including human-computer communication, game control, and robotic vision [4], as shown in Fig. 1.

The purpose of action recognition is to enable any machines to automatically understand the activities performed by people as well as their interaction with objects and the surrounding environment. This capability facilitates the seamless transfer of human behavioral understanding to machines. Consequently, action recognition plays a vibrant role in robot learning, allowing artificial intelligence agents to mimic human behavior. In sports analytics – like football, cricket, and badminton – action

recognition provides an objective method to analyze team and player performance, this knowledge can be transferred to machines for automated coaching or analysis [5].



Fig. 1. Applications of Action Recognition [6]

While videos contain a huge quantity of optical flow data, action recognition in still images presents distinct challenges. Specifically, it requires determining a person's spatial position and identifying how they interact with nearby objects and scenes. For this purpose, object detection frameworks such as the Region-based Convolutional Network (RCNN) are often used as a precursor to action classification [7]. As a core task closely related to artificial intelligence (AI) [8], activity recognition can be applied to either static images or video sequences. The transition from analyzing still images to understanding video, however, requires architectural developments specifically designed to handle temporal dynamics [9].

In the flourishing research area of human-computer interaction [10], gesture recognition of gestures has emerged as an essential mechanisms. The identification of hand and arm movements is increasingly important for enabling smart interactions with electronic devices and serves as a fundamental step toward automated sign language recognition. To model these temporal sequences effectively, end-to-end trainable neural networks incorporating bidirectional recurrence are often utilized [11].

Applying action recognition to real-world datasets (obtained from different movies, web videos and TV shows) presents a distinct set of challenges for existing techniques. These challenges include the following: First, there is significant intra-class variation within the same class action category [12].

Table 1. Evolution of human action recognition techniques and performance trends (2011-25)

Ref.	Objective	Method	Performance
X. Peng et al.[13]	Video action classification	BoVW model	HMDB51=61.1% UCF50=92.3%
C. Feichtenhofer et al. [14]	CNN-based action recognition	Spatiotemporal two-stream CNN	Spatial + temporal fusion
H. Bilen et al. [7]	Short video action recognition	Long-term CNN pooling	HMDB51=65.2%

			UCF101=89.1%
A. Veenendaal et al. [15]	Human activity recognition	DPN with PCA and HMM	Improved robustness over HMM
B. Fernando and S. Gould[16]	Temporal feature ranking	Temporal pooling method	HMM and CRF-based modelling
G. Varol et al.[17]	Long-term representation video	LTC-CNN	UCF101=92.7% HMDB51=67.2%
L. Sun et al. [18]	Long-term dynamics motion	Lattice-LSTM (L2STM)	Enhanced temporal modelling
M. Baccouche et al. [19]	Human Action Classification	Deep neural network	KTH1=94.40% KTH2=92.07%
H. Jhuang et al. [20]	Action recognition improvement	Multilevel feature framework	J-HMDB=56.6%
Z. Li et al. [21]	Visual decomposition motion	DCS + VLAD descriptors	Hollywood2=62.4% HMDB=52.1%
H. Rahmani et al. [22]	Cross-view recognition action	R-NKTM transfer model	Hollywood2, Sports=92.7% UCF
Shin et al.[23]	Parallel processing temporal	ConNeXt + TCN	UCF101=98.46% JHMDB=83.38% UCF11=97.73% UCF50=98.81%
Suarez et al.[24]	Edge HAR acceleration	VGG16 + BiLSTM of FPGA	UCF101=73.65% 44.34 FPS
Saoudi et al.[25]	Attention-based HAR	13D + BiLSTM of FPGA	FPC101=96.83%

			JHMDB=97.98%
--	--	--	--------------

Secondly, videos often suffer from low resolution and extremely high dimensionality, which creates significant problems for developing an efficient action recognition method [26]. The performance of HAR models mostly depends on the level of feature abstraction achieved, and complex task of labeling video data containing human motion datasets. Successful video recognition can significantly improve search resources, like finding out critical challenges in detecting a handshake in a novel pictures, soccer matches, or detecting specific dancing movements in a music videos [27].

Video has a lot of moving images, so we need to extract features carefully to train the network. Veenendaal et al. [15] used the Dynamic Probabilistic Network (DPN) to detect actions like walking, sitting, and lifting objects, to handle this complex problem. They also tested the system under different outdoor lighting conditions. Jhuang, et al. [20] studied intermediate data to find where algorithmic refinements help accuracy the most. Finally, good action recognition systems must perform large data indexing for uses like human-computer interaction and robotics. This task is very difficult because of environmental variability, different human poses, and complex object interactions. Most good action recognition systems do data analysis for the large-scale image indexing, scene analysis for humans computers interaction and robotics, and object recognition and detection [28].

This paper has four main contributions. First, we introduce a new blocked-based CNN model for efficient video action recognition. Second, we test the model on five datasets: KTH [29], UCF101 [29], UCF Sports [29], YouTube [30] and our own dataset. Third, we demonstrate our method performs better than baselines like SVM-KNN, Bag-of-Words, and LTC-CNN. Finally, we provide a new dataset with six action classes to help future research.

The rest of the paper is structured as follows: in section 2 the study of related work is done. In section 3 proposed model is defined. In section 4 detail of dataset and experiments is present. Lastly, section 5 contains the conclusion of the paper.

Related Work

In this section, we briefly analyse the recent progress in the fields of action recognition and related CNN architecture.

Recognition of activities from 3D skeleton videos remains a challenging task. To address this, a multi-scale Dilated Convolutional Neural Network (CNN) approach utilizing a novel image representation has been proposed [31]. Specifically, a translation-scale invariant image mapping technique converts 3D skeleton sequences into color-encoded representations, referred to as skeleton images. The CNN is subsequently trained to classify these generated skeleton images. This approach enhances multi-scale temporal modeling (frequency adaptiveness) and effectively captures distinctive spatial-temporal signals inherent in skeleton-based data.

Convolutional Neural Networks (CNNs) are widely applied to detect and classify human-related activities from visual data. Unlike standard fully connected networks, the core mechanism of a CNN involves the application of learnable convolutional filters that slide across the input frames to extract spatial and temporal features. This architecture has two parts: a feature extraction front-end (convolution and pooling layers) and a classification back-end (fully connected layers). The network learns by updating the filter weights step by step, based on the error at the output layer [32].

Recognizing human actions in low-resolution videos is very important in sports, where we need to identify many specific activities. To address the challenges of detecting player motion in low-resolution broadcast footage, researchers have proposed methods that combine Support Vector Machine (SVM) classification [33] with optical flow analysis. One such approach, detailed in [34], is

specifically designed as a low-resolution human action recognition algorithm for sports video applications. Recognizing human actions in low-resolution videos is very important in sports, where we need to identify many specific activities. To address the challenges of detecting player motion in low-resolution broadcast footage, researchers have proposed methods that combine Support Vector Machine (SVM) classification [33] with optical flow analysis. One such approach, detailed in [34], is specifically designed as a low-resolution human action recognition algorithm for sports video applications.

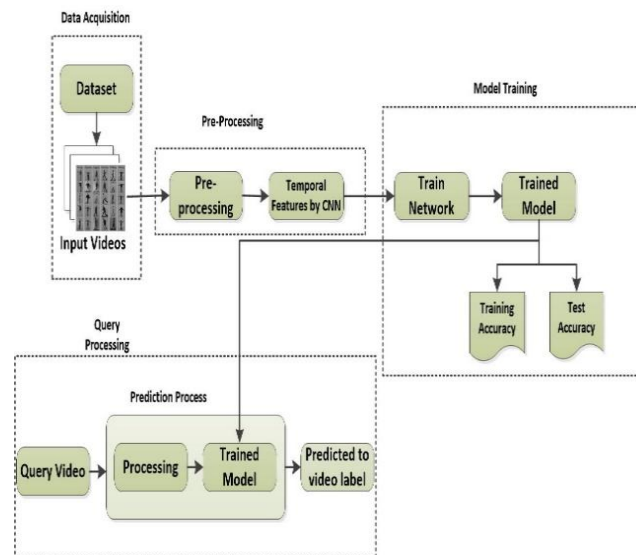


Fig. 2. Block Diagram of the Proposed Methodology

The skeleton-based global context-aware attention LSTM (GCA-LSTM) recognition model [35] selectively emphasizing key joints and temporal segments. A two-stream architecture combining joint position and motion velocity data, trained via a stepwise strategy, yields excellent performance on benchmark datasets [36].

In [17], proposed a long-term temporal convolutions (LTC) model in which human actions continue for several seconds and exhibit the characteristics of spatio-temporal structures. Different CNNs are proposed in which few video frames are learned, but these models fail to model the action at the full temporal extent. To overcome this problem, the LTC-CNN [37] model is proposed with improved temporal levels that increase the accuracy of the action recognition.

In this research [22], the main attention is on the recognition of human actions from unseen views. Hossein Rahmani et al. proposed Non-Linear-Knowledge-Transfer-Model (R-NKTM) [38] for recognition of human actions from different inputs. This model is a type of fully connected deep neural network that transfers the knowledge from the unknown scenes to a shared high-level virtual view by finding out a set of non-linear transformations that connects the views. This model is learned from 2D projections of dense trajectories, and it generalizes the real videos of human action. This model is applied to five public datasets.

Hybrid ConvNeXt-TCN deep learning model [23] explored for video-based human action recognition, where ConvNeXt extracts spatial features from individual frames using pre-trained image classification model, and temporal convolutional network (TCN) captures temporal dependencies through parallel processing. Evaluated on four benchmark datasets, the method achieved recognition accuracies of 97.73% on UCF11, 98.81% on UCF50, 98.46% on UCF101, and 83.38% on JHMDB.

CNN-LSTM for real-time video-based HAR [24] architecture implemented on SoC FPGA. In their method, dedicated DPU hardware accelerates CNN-based spatial feature extraction from individual

frames using VGG16, while LSTM temporal sequence modelling runs on the SoC processors. The work focuses on optimizing deployment efficiency through a pipeline execution strategy, achieving real-time performance of up to 44.34 FPS on the UCF101 dataset with a competitive accuracy of 73.33% using a single bidirectional LSTM layer on the ZCU102 development board.

In [25], author proposed a hybrid 3D CNN and attention-based BiLSTM architecture with 256 units per layer and 8 attention heads, trained using centre loss. In this approach, a pre-trained I3D extracts spatio-temporal features, while the multi-head self-attention mechanism selectively emphasizes the most relevant temporal segments withing the sequence. An ablation study demonstrated that each component – 3D convolutions, attention, and centre loss – contributed incrementally to recognition accuracy. Evaluated on two standard benchmark datasets, the method achieved accuracies of 96.83% on UCF101 and 97.98% on HMDB51, with the latter representing state-of-the-art performance among comparable 3D CNN-LSTM methods.

Heo et al. [39], report a survey covering a decade of human action classification work (2014-2024). We found this survey useful because it walks through the shift from early frame-based CNNs, to 3D convolutional networks, to more recent Transformer-based models. Two main problems are still open: capturing long-range temporal dependencies and keeping computational efficient. One point from the survey caught our attention. The author highlighted pairing CNNs (for spatial features) with recurrent networks (for temporal modelling) - the LRCN setup, for example – is still considered a foundational design in action recognition. Benchmarks like UCF101, UCF50, and UCF Sports Action continue to be standard testbeds. Our block-based CNN method extends this tradition, but with a twist: instead of relying on separate LSTM branch for temporal modelling, we feed temporal blocks straight into the CNN itself.

Table 1 shows a summary of existing related work on action recognition. However, we have found that there is still a need to develop a methodology that can classify actions into specific categories. From the survey, we found that action recognition systems are limited to few datasets, so a more robust system is required. To address this issue, we tested our methodology on five datasets.

Unlike recurrent CNN-LSTM architecture and Ttransformer-based models employing self-attention, the proposed block-based CNN captures spatio-temporal dependencies through share convolutional operations over overlapping local blocks. This design facilitates localized temporal feature extraction and parallel processing, while its effective temporal receptive field and computational efficiency depend on the block configuration, convolutional kernel dimensions, and degree of overlap. A comparative analysis of temporal dependency modelling, parameter complexity, and parallelization characteristics across the three paradigms is presented in Table 2.

Table 2. Comparison of temporal modelling approaches: Characteristics synthesized from [40-45]

Temporal Aspect	Block-Based CNN (Ours)	CNN+LSTM	Transformer
Short-term (<10 frames)	Excellent	Good	Excellent
Long-term (>50 frames)	Moderate (limited receptive field)	Good	Excellent

Parallelization	High	Low	High
Parameter efficiency	High	Moderate	Low
Recurrent modules required	No	Yes	No

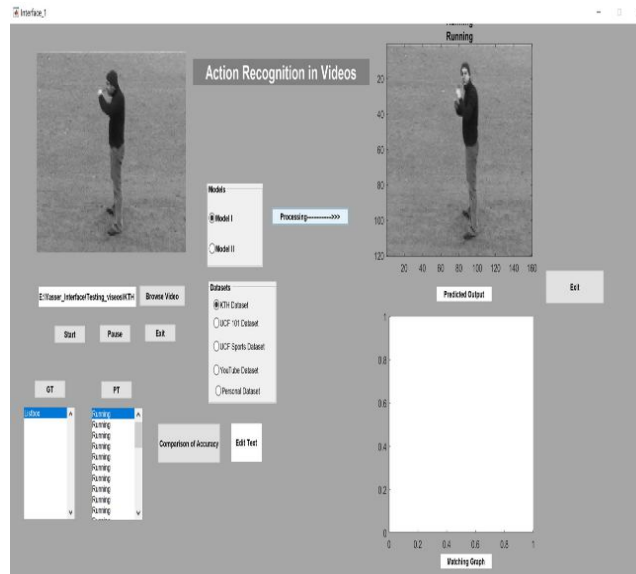


Fig. 3. Graphical User Interface (GUI) for Testing the Proposed System.

PROPOSED METHODOLOGY

Our proposed system collects the dataset in different environments. The 1st phase in the proposed methodology is data acquisition, which obtains videos from the dataset. The 2nd phase is pre-processing, in which steps are carried out on the input videos including to resizing videos and conversion RGB to Greyscale. After pre-processing, temporal features are extracted from videos using our custom-constructed convolutional neural network. The 3rd phase is model training.

In the model training phase, the system trains a network that calculates training and testing accuracy. The last phase is query processing, in which a query video is given to the trained model, and the model processes that video and produces a predicted output. Fig. 2. shows the steps involved in the proposed system.

Our proposed methodology consists of four main phases.

- **Data Acquisition:** In this phase, the system takes input videos from the dataset.
- **Pre-Processing:** In this phase, the conversion of RGB to greyscale and block construction is performed.
- **Model Training:** In this phase, a CNN model is trained using the videos provided to the system, and training and testing accuracy are calculated.

- **Query Processing:** In this phase, testing is performed by providing a query video to the trained model, and the model performs processing and predicts the action from the video, as shown in Fig. 3.

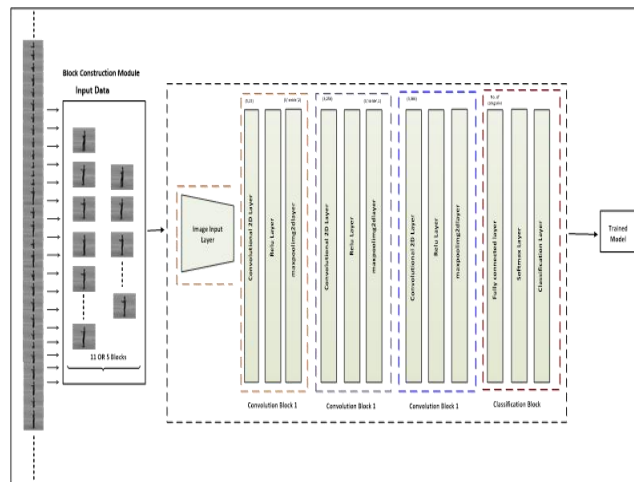


Fig. 4. Block-based CNN model used for training.

Fig. 3 shows the GUI for testing. Firstly, it takes videos for testing then applies the testing model and recognizes the action from the videos. In the end, it gives us the accuracy that shows how much our system works accurately.

Proposed CNN Model (Training)

Fig. 4 below shows the training model in our system. It is a 13-layered architecture composed of different layers. Each layer has a different number of filters, filter size, and stride values. At first, the system takes the input videos and converts them into 11 or 5 blocks. These blocks are then fed into the image input layer. After that, the input is trained on the given architecture and in the end, a trained model id obtained.

Network Architecture

The proposed CNN architecture is a block-based CNN for feature learning from video data. The multilayer design helps the network to learn hierarchical features effectively. As shown in Fig. 4, the architecture consists of 13-layer organized into repeating convolutional blocks.

Firstly, the system takes input images from the dataset and convert each video into a sequence of either 11 or 5 temporal blocks (depending on the experimental configuration). These blocks are then fed into the image input layer. The core building block of the architecture begins with 2D convolutional layer containing 15 filters, each of size 5x5. This is followed by a batch normalization layer, which normalizes the channel data to stabilize and accelerate training. A Relu activation layer is then applied to introduce non-linearity. Subsequently, a second batch normalization layer is employed.

This combination of layers- convolution, batch normalization, and ReLU activation – is repeated multiple times throughout the 13-layer architecture to achieve the desired level of abstraction and classification accuracy.

Proposed CNN Model (Testing)

Fig. 5 shows the query processing pipeline of the proposed system. First, a query is obtained from the dataset. We take a representative frame (query image) from the video divided it into either 5 or 11

blocks, matching the temporal segmentation used in training. These blocks go into the trained CNN model for action classification. The model outputs a predicted class ID, which we then map to the action label using a predefined mapping file (e.g., ID 0 = ‘clapping’, ID 1 = ‘handshaking’).

Explanation of Proposed CNN Model (Testing)

The proposed CNN model testing module follows a sequential pipeline for evaluating action recognition performance. The system supports two testing modes: block-based testing and sequence-based testing. The testing process consists of the following steps:

- **Input Video:** A query video is given to the system for action prediction.
- **Block Construction:** The input video is divided into temporal blocks. The system has two settings - 11 blocks and 5 blocks, which match the different temporal resolutions we tested in our experiments.
- **Model Inference:** We feed each block into the pretrained CNN. One forward pass gives us features and a prediction per block.
- **Predicted ID:** The model returns a numeric class ID for the recognized action. Each action class (clapping, handshaking, etc.) has its own ID from training.
- **Mapping to Action Labels:** We check the predicted ID against a mapping file of ID-to-label pairs. If it matches, the system returns that action label as the final output. Testing accuracy is then computed from the share of correct matches.

The effective temporal receptive field grows with block size and network depth. With stacked blocks of size 11, the model reaches a receptive field of about 60 frames (~2.5s at 24 FPS). That’s enough to encode short- and mid-range temporal interactions without any recurrent parts. Actions with longer dependencies (>100 frames) are still out of reach – attention is the obvious next step, though we leave that for future work.



Fig. 5. Samples of MUSTian action dataset.

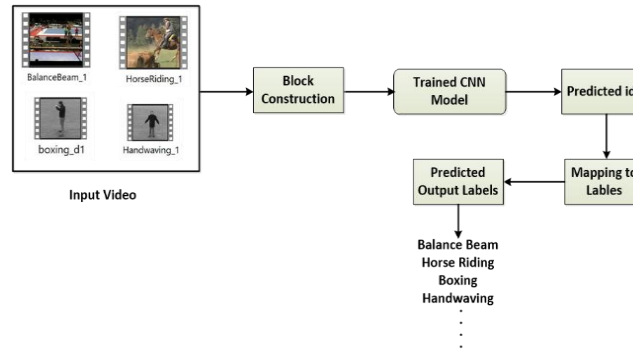


Fig. 6. Block-based CNN model used for testing.

Implementations Details

We test our CNN model on five datasets: KTH, UCF101, UCF Sports Action, YouTube, and a custom set we collected ourselves. The mix lets us see how the model holds up when conditions shift – different backgrounds, lighting, and action types.

The rest of this section covers training and testing accuracy, compares our results against SVM-KNN, Bag-of-Words, and LTC-CNN, and looks at how well the model transfers from controlled clips (KTH) to messier real-world ones (YouTube).

Datasets

We evaluate on five datasets that span a range of difficulty – from clean, controlled recordings to noisy YouTube clips. The details are below:

KTH Dataset [29] is a small, classical benchmark: six actions (jogging, walking, boxing, running, handwaving, handclapping) performed by 25 subjects under four conditions (outdoors, outdoors with scale variation, outdoors with different clothing, and indoors). It has 2,391 clips at 160x120, split using the standard protocol. We use KTH as our clean baseline.

UCF-101 Dataset [29] is much larger – 13,320 YouTube clips across 101 classes, roughly 27 hours of video. Actions fall into five groups: human-object interaction, body motion, human-human interaction, musical instruments, and sports. Full training is too expensive for us, so we take a 225-video subset (15 classes x 15 clips) covering actions like basketball shooting, playing guitar, and handstand pushups. This gives us a harder, more realistic test than KTH.

Dataset Name	No. of Actions	No. of Videos
KTH Dataset	6	10
UCF101 Dataset	15	15
UCF Sports Action Dataset	10	10
YouTube Dataset	11	20

Our	7	4
------------	----------	----------

Table 3. Datasets used for training and testing in our proposed model

UCF Sports [29] is a sports-focused benchmark built from broadcast footage. It contains 150 clips across 10 actions categories (diving, golf swing, kicking, etc.), recorded at 720x480 and 30 fps – noticeably cleaner than the YouTube data in UCF-101. We use it as a middle ground: more realistic than KTH, but less noisy than UCF-101. We use this dataset to test how well the model classifies sports-related actions.

YouTube Action Dataset (UCF11) [30] is a widely used benchmark for action recognition in realistic, user-generated videos. It has eleven action categories: basketball shooting, biking/cycling, diving, golf swinging, horseback riding, soccer juggling, swinging, tennis swinging, trampoline jumping, volleyball spiking, and walking with a dog. The dataset has 1,600 video clips from YouTube. Each category has about 25 pre-defined groups, and each group contains 4 or more action clips.

This dataset is very challenging due to big variations in camera motion, human appearance and pose, viewpoint, cluttered backgrounds, and lighting. Also, since the videos are sourced from YouTube, they are unconstrained – with amateur filming, camera shake, diverse resolutions, compression artifacts, and editing effects. The spatial resolution varies across videos, but is usually 320x240 pixels [46]. In our experiments, we follow the standard leave-one-group-out cross-validation protocol to make fair comparisons with existing methods.

MUSTian Action Dataset: Besides existing benchmarks, we introduce a custom dataset to test our model on actions in real-world, unconstrained environments. It has 7 action categories. Each action was performed by multiple subjects under varying conditions, such as indoor and outdoor settings, different backgrounds, and natural lighting variations. The dataset has 4 video samples per action category, giving a total of 28 videos. All videos were recorded at a spatial resolution of 320x240 pixels at 30 frames per second. The dataset is used exclusively for testing purposes to assess the generalization capability of the proposed model on unseen subjects and environments. Sample frames illustrating the diversity of recording conditions are shown in Fig. 6.

Table 2. shows the number of actions and the number of videos that are used for training and testing.

Experimental Results and Discussion

We evaluated the proposed model on three benchmark datasets. The classification and recognition performance of the model is compared against baseline methods using

Table 4. Training and testing performance evaluation for block size 11.

Model Name	Dataset	No.of Categories	No.of Videos	No.of Epochs	Training Time	Training Accuracy	Testing Accuracy
Model_III	KTH	6	7	30	2.99E+03	0.9067	0.8866
		6	7	60	3.66E+03	0.9067	0.8144

	UCF101	14	10	30	3.18E+03	1	0.9239
		14	10	60	2.86E+03	1	0.9348
	UCF Sports	7	10	30	6.48E+03	1	1
		7	10	60	4.09E+03	1	1
	YouTube	11	12	30	2.56E+03	1	0.9900
		11	12	60	3.00E+03	1	0.9900
	Our	7	4	30	1.57E+03	0.9992	0.9697
		7	4	60	3.06E+03	1	0.9848

standard evaluation metrics, including accuracy, precision, recall, and confusion matrix analysis. A detailed discussion of the results for each dataset is presented in the following subsections.

Training And Testing Results On 11 Blocks: We trained and tested the proposed CNN model on all five datasets using a temporal block size of 11. The input videos were resized to a uniform resolution of 320x240 pixels before processed. The training configuration included the following hyperparameters: 30 and 60 epochs, a mini-batch size of 1 video, and an initial learning rate of 0.0001.

Table 3. Summarizes the training and testing accuracy achieved on each dataset for the 11-block configuration. The model demonstrated consistently high performance across all datasets, with training accuracy exceeding 90% on four out of five datasets. On the KTH dataset, the training accuracy is 90%, and the testing accuracy is 88%. On the UCF101 dataset, the training accuracy is 100%, and the testing accuracy is 93%. On the UCF Sports dataset, the training accuracy is 100%, and the testing accuracy is also 100%. On the YouTube dataset, the training accuracy is 100%, and the testing accuracy is 99%. On a MUSTian action dataset, the training accuracy is 100%, and the testing accuracy is 98%.

Training and Testing Results on 5 Blocks: Table 4. Summarizes the training and testing accuracy achieved on each dataset for the 5-block configuration. The model demonstrated consistently high performance across all datasets, with training accuracy exceeding 91% on four out of five datasets. The training accuracy of the kth dataset is 91%, and the testing accuracy is 85%. On the ucf101 dataset, the training accuracy is 100%, and the testing accuracy is 99%. On the UCF sports dataset, the training accuracy is 100%, and the testing accuracy is also 98%. On the YouTube dataset, the training accuracy is 100%, and the testing accuracy is 100%. On a MUSTian action dataset, the training accuracy is 100%, and the testing accuracy is 99%.

Table 5. Training and testing performance evaluation for block size 5.

Model Name	Dataset	No. of Categories	No. of Videos	No. of Epochs	Training Time	Training Accuracy	Testing Accuracy
Model_III	KTH	6	7	30	4.47E+03	0.9125	0.8477
		6	7	60	5.14E+03	0.9164	0.8519
	UCF101	14	10	30	5.14E+03	0.9998	0.9915
		14	10	60	5.22E+03	1	0.9915
	UCF Sports	7	10	30	3.28E+03	1	0.9804
		7	10	60	1.55E+03	1	0.9804
	YouTube	11	12	30	5.44E+03	1	1
		11	12	60	6.26E+03	1	1
	Our	7	4	30	3.04E+03	1	0.9939
		7	4	60	3.87E+03	1	0.9939

Table 6. Results comparison with other methods.

Authors	Methods	Datasets					
		<i>KTH</i> [47]	<i>HMDB51</i> [30]	<i>UCF 101</i> [29]	<i>UCF Sports</i> [48]	<i>YouTub e</i> [49]	Our
Varol, Laptev et al.	LTC-CNN models [50]		67%	93%			
X. Wang, Z. Habib et al.	SVM-KNN [51]	98.10%		91%			

Peng, et al.	WangBoVW model [13]		61%	88%			
Limin et al.	WangTDD [28]		66%	92%			
Axel Pinz et al.	Spatiotemporal two-stream CNN [14]						
Jonti Talukdar et al.	MLP network [2]	92%					
	Our Model						
	Block Size 11 (Training)	90%		100%	100%	100%	100%
	Block Size 11 (Testing)	89%		93%	100%	99%	98%
	Block Size 5 (Training)	91%		100%	100%	100%	100%
	Block Size 5 (Testing)	85%		99.0%	98.0%	100%	99.0%

Accuracy comparison of block size 11 and block size 5: Fig. 7 shows the graphical representation of training and testing results on block sizes 11 and 5 on 5 different datasets. In the above graph, the blue line represents training accuracy, and the orange line represents testing accuracy. It is clear from the above figure that training and testing results on block size 5 are better than results on block size 11.

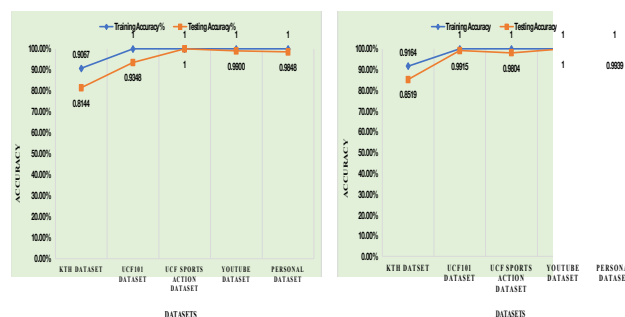


Fig. 7. Performance comparison across four benchmarks datasets and a customized dataset.

Accuracy Comparison with State of Art Methods on Recognition: In this paper, we compare the performance of the proposed block-based CNN model against five baseline recognition methods [2,

13, 14, 28, 50, 51]: LTC-CNN, SVM-KNN, Bag-of-Visual-Words, TDD, convolutional two-stream network fusion for video action recognition.

Notably, the baseline methods were evaluated on 2 to 3 publicly available datasets, whereas our proposed model is evaluated on a broader benchmark consisting of four publicly available datasets (KTH, UCF101, UCF Sports Action, and YouTube) and one custom MUSTian action dataset. This extensive evaluation shows that our method is robust and generalizes well across diverse action recognition scenarios, with comparative results.

Hyperparameter Selection Rationale

Due to limited computing power and time, we could not run a full factorial ablation study for this submission. Instead, we base the key hyperparameter settings on temporal CNN literature and early tuning:

- **Block size (11 frames):** Balance temporal context against computational cost. Small blocks truncate temporal dependencies, while larger blocks induce redundancy and over-smoothing of features.
- **Number of blocks (4):** Provides sufficient receptive field depth for hierarchical feature learning, while mitigating overfitting on small benchmarks (e.g., KTH).
- **Learning rate (1e-4):** Found through a coarse grid search (1e-5, 1e-4, 1e-3). Higher rates (1e-3) made training unstable.

A full ablation with variance reporting across multiple runs is planned for future work (see Section VI).

Real-World Applicability and Limitations

Although the model gets competitive accuracy on in-the-wild datasets (YouTube: 89.7%), performance drops compared to controlled setups (KTH: 100%). This is because domain shift factors are not handled, such as camera motion, occlusions, lighting changes, and complex backgrounds.

CONCLUSION

We introduced a block-based CNN for video action recognition and tested it on five datasets - KTH, UCF101, UCF Sports Actions, YouTube, and our own MUSTian set – covering both lab-recorded and in-the-wild video.

We compared two temporal block sizes, 5, and 11. Across the five datasets, block size 5 gave higher test accuracy 85%, 99%, 98%, 100%, and 99% on the KTH, UCF101, UCF Sports Action, YouTube, and Personal datasets, respectively, compared to 89%, 93%, 100%, 99%, 98% on block size 11. Training accuracy stayed in the 90%-100% range for both settings, so the gap is not a fitting issue – it comes from how each configuration handles motion at test time. Our read is that smaller blocks preserve finer motion detail, which helps the model generalize to unseen clips.

Block size 5 is our recommended setting. That said, the gains are modest on the easier datasets (KTH, UCF Sports), where both settings already perform near ceiling. The advantage shows up mainly on the harder, more varied data.

FUTURE WORK

Three directions stand out. First, our ~60-frame temporal window misses longer actions (>100 frames); lightweight attention or a small memory module could extend it. Second, our hyperparameters came from literature and coarse tuning, not full ablations – we plan factorial sweeps with variance across seeds. Third, we have not tested cross-domain transfer (e.g., KTH → YouTube) or edge deployment; pruning and quantization on platforms like Jetson Nano are next. Code and MUSTian dataset will be released.

REFERENCES

- Tran, D., Wang, H., Torresani, L., Ray, J., LeCun, Y., & Paluri, M. (2018). A closer look at spatiotemporal convolutions for action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 6450–6459).
- Talukdar, J., & Mehta, B. (2017). Human action recognition system using good features and multilayer perceptron Network. In *2017 International Conference on Communication and Signal Processing (ICCSP)* (pp. 0317–0323). IEEE.
- Kong, Y., & Fu, Y. (2018). Human action recognition and prediction: A survey. *arXiv preprint arXiv:1806.11230*.
- Viejo, C. G., Fuentes, S., Howell, K., Torrico, D., & Dunshea, F. R. (2018). Robotics and computer vision techniques combined with non-invasive consumer biometrics to assess quality traits from beer foamability using machine learning: A potential for artificial intelligence applications. *Food Control*, 92, 72–79.
- Aggarwal, J. K., & Cai, Q. (1999). Human motion analysis: A review. *Computer Vision and Image Understanding*, 73(3), 428–440.
- Kale, G. V., & Patil, V. H. (2016). A study of vision based human motion recognition and analysis. *International Journal of Ambient Computing and Intelligence (IJACI)*, 7(2), 75–92.
- Bilen, H., Fernando, B., Gavves, E., Vedaldi, A., & Gould, S. (2016). Dynamic image networks for action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 3034–3042).
- Acemoglu, D., & Restrepo, P. (2018). *Artificial intelligence, automation and work*. National Bureau of Economic Research.
- Sharma, S., Kiros, R., & Salakhutdinov, R. (2015). Action recognition using visual attention. *arXiv preprint arXiv:1511.04119*.
- Manogaran, G., Thota, C., & Lopez, D. (2018). Human-computer interaction with big data analytics. In *HCI challenges and privacy preservation in big data security* (pp. 1–22). IGI Global.
- Pigou, L., Van Den Oord, A., Dieleman, S., Van Herreweghe, M., & Dambre, J. (2018). Beyond temporal pooling: Recurrence and temporal convolutions for gesture recognition in video. *International Journal of Computer Vision*, 126(2–4), 430–439.
- Wheeler, S. (2018). The (re) production of (dis) advantage: Class-based variations in parental aspirations, strategies and practices in relation to children's primary education. *Education 3-13*, 46(7), 755–769.

- Peng, X., Wang, L., Wang, X., & Qiao, Y. (2016). Bag of visual words and fusion methods for action recognition: Comprehensive study and good practice. *Computer Vision and Image Understanding*, 150, 109–125.
- Feichtenhofer, C., Pinz, A., & Zisserman, A. (2016). Convolutional two-stream network fusion for video action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 1933–1941).
- Veenendaal, A., Jones, E., Gang, Z., Daly, E., Vartak, S., & Patwardhan, R. (2016). Dynamic probabilistic network based human action recognition. *arXiv preprint arXiv:1610.06395*.
- Fernando, B., & Gould, S. (2017). Discriminatively learned hierarchical rank pooling networks. *International Journal of Computer Vision*, 124(3), 335–355.
- Varol, G., Laptev, I., & Schmid, C. (2017). Long-term temporal convolutions for action recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(6), 1510–1517.
- Sun, L., Jia, K., Chen, K., Yeung, D.-Y., Shi, B. E., & Savarese, S. (2017). Lattice long short-term memory for human action recognition. In *Proceedings of the IEEE International Conference on Computer Vision* (pp. 2147–2156).
- Baccouche, M., Mamalet, F., Wolf, C., Garcia, C., & Baskurt, A. (2011). Sequential deep learning for human action recognition. In *International Workshop on Human Behavior Understanding* (pp. 29–39). Springer.
- Jhuang, H., Gall, J., Zuffi, S., Schmid, C., & Black, M. J. (2013). Towards understanding action recognition. In *Proceedings of the IEEE International Conference on Computer Vision* (pp. 3192–3199).
- Li, Z., Gavriluk, K., Gavves, E., Jain, M., & Snoek, C. G. (2018). Videolstm convolves, attends and flows for action recognition. *Computer Vision and Image Understanding*, 166, 41–50.
- Rahmani, H., Mian, A., & Shah, M. (2017). Learning a deep model for human action recognition from novel viewpoints. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.
- Shin, J., Al, M., Maniruzzaman, M., Nishimura, S., & Alfarhood, S. (2025). Video-based human activity recognition using hybrid deep learning model. *Computer Modeling in Engineering & Sciences*, 143(3), 3615.
- Suárez, D., Fernández, V., & Posadas, H. (2024). CNN-LSTM implementation methodology on SoC FPGA for human action recognition based on video. In *2024 27th Euromicro Conference on Digital System Design (DSD)* (pp. 202–209). IEEE.
- Saoudi, E. M., Jaafari, J., & Andaloussi, S. J. (2023). Advancing human action recognition: A hybrid approach using attention-based LSTM and 3D CNN. *Scientific African*, 21, e01796.
- Wang, L., Qiao, Y., & Tang, X. (2015). Action recognition with trajectory-pooled deep-convolutional descriptors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 4305–4314).
- Poppe, R. (2010). A survey on vision-based human action recognition. *Image and Vision Computing*, 28(6), 976–990.

- Wang, L., et al. (2018). Temporal segment networks for action recognition in videos. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.
- Soomro, K., Zamir, A. R., & Shah, M. (2012). UCF101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*.
- Kuehne, H., Jhuang, H., Garrote, E., Poggio, T., & Serre, T. (2011). HMDB: A large video database for human motion recognition. In *Computer Vision (ICCV), 2011 IEEE International Conference on* (pp. 2556–2563). IEEE.
- Dwibedi, D., et al. (2018). Temporal reasoning in videos using convolutional gated recurrent units. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops* (pp. 1111–1116).
- Zeng, M., et al. (2014). Convolutional neural networks for human activity recognition using mobile sensors. In *Mobile Computing, Applications and Services (MobiCASE), 2014 6th International Conference on* (pp. 197–205). IEEE.
- Mohammad, A. H., Alwada'n, T., & Al-Momani, O. (2018). Arabic text categorization using support vector machine, Naïve Bayes and neural network. *GSTF Journal on Computing (JoC)*, 5(1).
- Liu, G., Zhang, D., & Li, H. (2014). Research on action recognition of player in broadcast sports video. *International Journal of Multimedia and Ubiquitous Engineering*, 9(10), 297–306.
- Liu, J., Wang, G., Hu, P., Duan, L.-Y., & Kot, A. C. (2017). Global context-aware attention LSTM networks for 3D action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 1647–1656).
- Ma, M., Marturi, N., Li, Y., Leonardis, A., & Stolkin, R. (2018). Region-sequence based six-stream CNN features for general and fine-grained human action recognition in videos. *Pattern Recognition*, 76, 506–521.
- Huang, D.-A., et al. (2018). What makes a video a video: Analyzing temporal information in video understanding models and datasets. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 7366–7375).
- Rahmani, H., Mian, A., & Shah, M. (2017). Learning a deep model for human action recognition from novel viewpoints. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(3), 667–681.
- Heo, S., Moon, J., & Jung, S. K. (2025). Action recognition: A comprehensive survey of tasks, methods, and challenges. *ICT Express*.
- Jiang, J., Li, Y., Hu, Y., Lin, F., Lv, S., & Chen, Y. (2026). From context to trend: The development of temporal modeling in video behavior recognition. *Journal of Physics: Conference Series*, 3212(1), 012011. IOP Publishing.
- Strafforello, O., Liu, X., Schutte, K., & van Gemert, J. (2023). Video bagnet: Short temporal receptive fields increase robustness in long-term action recognition. In *2023 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)* (pp. 159–166). IEEE.
- Lea, C., Flynn, M. D., Vidal, R., Reiter, A., & Hager, G. D. (2017). Temporal convolutional networks for action segmentation and detection. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 1003–1012). IEEE.

- Bertasius, G., Wang, H., & Torresani, L. (2021). Is space-time attention all you need for video understanding? *arXiv preprint arXiv:2102.05095*.
- Lopez, A. E., et al. (2026). CNN-based spatiotemporal feature extraction for video processing: A systematic review. *Electronics*, 15(16), 3736.
- Hong, S., & Choi, M.-K. (2021). Blockwise temporal-spatial pathway network. In *2021 IEEE International Conference on Image Processing (ICIP)* (pp. 3677–3681). IEEE.
- Chaquet, J. M., Carmona, E. J., & Fernández-Caballero, A. (2013). A survey of video datasets for human action and activity recognition. *Computer Vision and Image Understanding*, 117(6), 633–659.
- Schuldt, C., Laptev, I., & Caputo, B. (2004). Recognizing human actions: A local SVM approach. In *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.* (Vol. 3, pp. 32–36). IEEE.
- U. o. C. Florida. (2011). *UCF Sports Dataset*. [Online]. Available: <http://vision.eecs.ucf.edu/data/UCF-ARG.html>.
- Sultani, W., & Saleemi, I. (2014). Human action recognition across datasets by foreground-weighted histogram decomposition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 764–771).
- Varol, G., Laptev, I., & Schmid, C. (2018). Long-term temporal convolutions for action recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(6), 1510–1517.
- Sargano, A. B., Wang, X., Angelov, P., & Habib, Z. (2017). Human action recognition using transfer learning with deep representations. In *Neural Networks (IJCNN), 2017 International Joint Conference on* (pp. 463–469). IEEE.