Computers, Materials & Continua
DOI:10.32604/cmc.2022.020655
Article

Tech Science Press

HARTIV: Human Activity Recognition Using Temporal Information in Videos

Disha Deotale[1], Madhushi Verma[2], P. Suresh[3], Sunil Kumar Jangir[4], Manjit Kaur[2], Sahar Ahmed Idris[5] and Hammam Alshazly[6,*]

[1]CSE Department, G. H. Raisoni Institute of Engineering and Technology, SPPU University, Pune, India
[2]CSE Department, Bennett University, Greater Noida, India
[3]TML Business Services Limited, Pune, India
[4]CSE Department, Anand International College of Engineering, Jaipur, Rajasthan, India
[5]College of Industrial Engineering, King Khalid University, Abha, Saudi Arabia
[6]Faculty of Computers and Information, South Valley University, Qena, 83523, Egypt
[*]Corresponding Author: Hammam Alshazly. Email: hammam.alshazly@sci.svu.edu.eg
Received: 02 June 2021; Accepted: 12 July 2021

Abstract: Nowadays, the most challenging and important problem of computer vision is to detect human activities and recognize the same with temporal information from video data. The video datasets are generated using cameras available in various devices that can be in a static or dynamic position and are referred to as untrimmed videos. Smarter monitoring is a historical necessity in which commonly occurring, regular, and out-of-the-ordinary activities can be automatically identified using intelligence systems and computer vision technology. In a long video, human activity may be present anywhere in the video. There can be a single or multiple human activities present in such videos. This paper presents a deep learning-based methodology to identify the locally present human activities in the video sequences captured by a single wide-view camera in a sports environment. The recognition process is split into four parts: firstly, the video is divided into different set of frames, then the human body part in a sequence of frames is identified, next process is to identify the human activity using a convolutional neural network and finally the time information of the observed postures for each activity is determined with the help of a deep learning algorithm. The proposed approach has been tested on two different sports datasets including ActivityNet and THUMOS. Three sports activities like swimming, cricket bowling and high jump have been considered in this paper and classified with the temporal information i.e., the start and end time for every activity present in the video. The convolutional neural network and long short-term memory are used for feature extraction of temporal action recognition from video data of sports activity. The outcomes show that the proposed method for activity recognition in the sports domain outperforms the existing methods.

Keywords: Action recognition; human activity recognition; untrimmed video; deep learning; convolutional neural networks

1 Introduction

Computer vision is a domain handling complex challenges with precision and performs image analysis just like humans do. The algorithms are designed to understand the image content and process it in a way like that of the human brain using the concepts of machine learning. The neural networks that can handle and process images are called convolutional neural networks (CNNs). A video is a sequence of image frames that can be provided to a CNN network for image analysis. In still images, one can use CNN to identify the features [1], however, in videos, it is critical to capture the context between the image frames extracted from the video while labelling to avoid loss of data. Consider an image of a half-filled cardboard box that can be labelled as packing a box or unpacking a box depending on the frames before and after it. Such situations cannot be efficiently handled by CNNs and can only consider feature space, such as the visual information in an image, and not the temporal or time-related features. To solve this problem, we need to transfer the CNN's output into another model that can manage the temporal information present in the images. This sort of model is named a recurrent neural network (RNN).

A CNN can handle groups of pixels separately whereas an RNN can keep track of what it has already processed and use it in higher cognitive phases. RNN can deal with a wide range of input and output data. Classification of videos can be performed by training RNNs using a sequence of frames passed with labels. RNN compares its expected performance with the correct label using a loss or error function as it processes each sequence. It then changes the weights and repeats the procedure until it reaches a higher level of accuracy. The issue with these image and video models is that the amount of data needed to fully reproduce human vision is massive. If a model has been trained to recognize the picture of a duck and an image with the same lighting, colour, angle, and shape, is provided it can identify that it is a duck. If there is a change in any of these features, or even if the image is rotated then the algorithm may not be able to interpret what is it in the image. The stated situation explains the problem associated with these deep learning-based models. To enable an algorithm to understand and recognize image content in the same way as a human brain does, it must be fed with the massive quantity of data containing millions of objects from thousands of angles, all annotated and properly identified.

There are three basic levels in the process of interpreting a new human activity: (1) action classification which deals with the question of what i.e. what is the activity a person is performing, (2) the temporal action detection or localization deals with the problem based on the action classification result and helps in determining the action start and end time and (3) the last level is spatial-temporal action detection that answers the question about where the activity is in the video frame. These three basic steps can help in determining which person is performing the activity, what is the activity and when did the activity start or end in the video and provide an in-depth understanding of the human activity. Rich information in terms of space and time are present in the videos. The biggest challenge is how to extract this information and use it for prediction effectively and efficiently. One important problem setting is the trimmed video *vs.* untrimmed video available in a lot of datasets. Trimmed video means human and tailors have already removed the part of the video which is irrelevant to the labels and the other variant of the problem is untrimmed video classification. Some datasets like ActivityNet [2] and THUMOS [3] are built from untrimmed videos and irrelevant parts are still included in the dataset. So, an algorithm for untrimmed video classification must also take into consideration that there is irrelevant information in the video that must be removed for the sake of prediction. In the case of trimmed videos, the classification algorithms have achieved a very high accuracy but it is important to understand that these algorithms cannot deal with irrelevant information, so their real-world appli-

cation is somehow limited. Hence, to make such algorithms relevant for real-world applications, it needs additional processing so that it can tackle the irrelevant information, which is a part of these videos.

Human activity recognition from video data is one of the most difficult research areas, particularly in applications like surveillance, video tutorials for beginners in the sports domain, etc. The time information of human activity is useful for maintaining the records in the sports domain and is especially useful when a coach/trainer is unavailable. After noticing the portion in the video where an action has been executed, individual performance can be judged by the trainer and at the same time such a contribution can help the beginners in the sports domain to acquire the required skills and train themselves in the absence of a coach. Another advantage of this methodology is that it can be used on surveillance cameras, which are popularly used these days to identify unusual events that have occurred in the past. This video data is available in a database for identifying criminal parts with time information that occurred in public. Activity with time information play an important role in the application of abnormal event detection, suspicious activity detection from the video, etc. It's very useful in live camera to stop the unusual activity before completion, to learn the activity step by step for performing well after practice and understand the parts of the activity like how it should be performed, from where the beginners should start and observe the minute details of the activities. Some other applications that can be stated includes training of children at home by observing the activities along with time information for learning some specific tricks of the sports and patient activity monitoring system with time can help in analysing the progress in the patient's state.

The key contribution of this work includes the identification of localized temporal human activity in an untrimmed video by:

- Classifying different human activities from the sports domain. The model is trained with spatial-time-related action localization detector for untrimmed video dataset, which searches for local activity in the input video by passing it to the trained model and producing not only the frame-to-frame division but also the activity time of all three sports activities (swimming, cricket bowling, and high jump).
- Based on a temporal segment network and the CNN approach, the proposed method achieves time-related localization and space division of video into frames with multiple sports activities occurring locally in the input video.
- To precisely evaluate the proposed method and to establish a baseline for future research, a new dynamic class activity (DCA) Sport dataset is proposed, consisting of three sports class activity videos that will be useful for new researchers to annotate the human movement present in the untrimmed video.

The organization of this paper is as follows. In Section 2, the related work has been discussed. In Section 3, a detailed description of the methodology has been presented. The experimental analysis is stated in Section 4, and finally, the conclusion is presented in Section 5.

2 Related Work

A variety of work has been done to perform human action detection and classification on image and video data. Several techniques have been developed to train the computers so that it can interpret and understand the contents inside the images. A video is just a set of consecutive images or frames. For applying the deep learning models to accurately detect and classify an image, only the action part can be identified and not an activity. It is possible to identify

human activity in an untrimmed video using the images or different frames along with the frame sequence. Researchers have worked on different human activities and have proposed techniques to classify them. To identify the time of a human activity in an untrimmed video and to comprehend the main activity's behaviour, an activity recognition framework can be used. When the framework is applied on the dataset, essentially three steps are executed i.e., detecting the objects, identifying the actions, and recognising the temporal movements with time information. Most of the research in this domain is based on trimmed video to acknowledge the activities present in the video. In the literature available, the activities have been classified into four groups namely single activity, group activity, human-human interaction activity, and human-object interaction activity.

In [4], a method for performing a generic action classification in a continuous video using long-term video action recognition (LVAR) has been proposed. ReHAR was introduced in [5], which is a new robust and efficient human activity recognition technology that can be used to predict the individual and group activities. A method for recognizing and locating human actions using temporal action recognition was introduced in [6]. To extract features, a multilayer CNN was used. Experiments were performed using surveillance video from an offshore oil production station. Standing, walking, and falling were all recognized and located in the uncut long video. In [7], an automatic human activity identification system that recognizes human's actions without human intervention has been described. The authors tested four deep learning approaches and thirteen machine learning algorithms, including neural networks, random forest, support vector machine (SVM), decision tree classifier, and others, to find the most efficient process of human movement recognition. Laying, sitting, standing, walking, walking downstairs, and walking upstairs all are activities that should be recognized. A new framework for providing immediate assistance to crime victims and swift action against perpetrators was created in [8]. By merging adaptive video compression and a convolutional 3D (C3D) network with contextual multiple scales based on temporal information, many scales based on context can be created. A deep neural network with convolutional layers and long short-term memory (LSTM) has been demonstrated in [9]. Here, they collect and categorize activity features using a few model parameters. To speed up convergence, a batch normalization layer (BN) was added after the global average pooling (GAP) layer. In [10], the model predicts the highest k labels for every untrimmed video by analyzing global video-level features. First, frame-level binary classification is combined with dynamic programming to come up with temporally trimmed activity proposals. Second, each frame is assigned a label supported by the worldwide label and scored with the score of the temporal activity proposal along with the global score. Finally, the author has performed untrimmed video classification with the assistance of an SVM. In [11], the author has developed a long-term video modelling framework called long-term video action recognition (LVAR) that can store temporal information over time. Firstly, the extra partial temporal recurrence (PTR) block in a C3D network showed a performance increment as compared to its vanilla C3D network. Secondly, information of previous time step is employed in current time step to permit information to propagate through longer duration. Finally, the data propagation along with time makes each time step feature to hold the contextual information of the previous time step, enabling them to acknowledge the clip instances with higher performances. In [12], the authors have combined the capabilities of both 3D-CNNs and RNNs into one framework. They designed an easy network that takes a sequence of video features from the C3D model as input to an RNN and can classify all of them into an activity category. In [13] a strong framework for the classification of varied categories of sports using neural networks and InceptionV3 has been used. This framework has achieved a mean accuracy of 96.64%. In [14], the model generates an optical flow image for every video

frame. Then, both video frames and their corresponding optical flow images are fed into one frame representation model to get the representations.

Video activity detection is the task of temporally dividing a video into semantic activity. An activity is a series of consecutive actions and specially for the sports domain, it depicts different sports activities (where each clip is a series of frames taken from the same camera at the same time) like swimming, cricket bowling, high jump, etc. Feature extraction from images of video data is used as an approach in the image recognition problems to have a better image representation than raw pixel values. Then the output is fed into the machine and deep learning algorithms to recognize a specific category of objects from the images. The video surveillance task involves two kinds of algorithms: firstly, tracking of objects and secondly, classification of actions. Tracking of objects is a process wherein the current image and the previous ones in the video stream are observed and compared to determine the pose change. From the first image to the last, one must access the full video in a video sequence. An image with motions depicted in it is the only difference between a video and a still image [15]. It is a powerful thing to monitor, and it can lead to action comprehension, pose estimation, or motion tracking. In video analysis, this primary problem is called optical flow estimation. Optical flow is the concept of calculating a pixel change between any two continuous video frames. As illustrated in Fig. 1, this is treated as a correspondence problem. Different segmentation approaches are commonly used to detect moving objects from a sequence of images collected from a static camera. The goal of this approach was to detect moving items by comparing the existing and linked frames. During frame sequences, F_i is the i^{th} frame's value. F_{i+1} is the value of the $i+1^{th}$ image in the frame sequence. $F_{d(i,i+1)} = |F_{i+1} - F_i|$ is the definition of frame difference. The frame difference method is the most used method for detecting motion. To locate the moving object, this method uses pixel-based difference.

2.1 Temporal -Segment-Networks (TSN)

The most recent and popular work on untrimmed videos includes the TSN framework [16], which is a general framework and has two main components for modelling long-range temporal structures. The first component is a sparse sampling system in which each video is split into a predetermined number of parts, and each segment is sampled at random with one snippet. The second one is a segment aggregation scheme in which each snippet receives a classification mark and then each snippet information is aggregated to achieve the final classification at the video level through a segment's consensus feature.

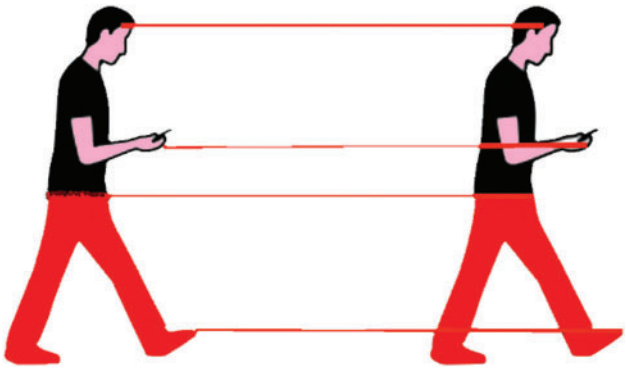


Figure 1: Change from frame 0 to frame 1 to determine the optical flow

There are several benefits of studying an untrimmed video. First, the scheme becomes sufficiently general to deal with any video (untrimmed and trimmed). Second, since the input videos are first divided into a fixed number of parts, without substantially raising the processing cost, the system can handle videos of any length. Finally, the effect of repetitive knowledge on the learning process is mitigated, as the samples are taken from these segments. Fig. 2 shows how the activity can be recognized with time, based on the sequence of frames of the same action. Optical flow identifies the motion of an activity based on frames. A single frame simply identifies the action. On the other hand, sequences of frames with optical flow identifies the human's movement or actions. To begin with, firstly the optical flow between two frames is to be determined. It is important to ensure that the average displacement vector field is big when extracting flow at that moment. The optical flow between frames results in an extremely quick movement activity when measured in frames per second (fps). It is valid in some activities, such as swimming, where horizontal movement is equivalent and height movement on either side is equivalent. The model is created by stacking optical flow displacement fields over multiple successive frames. This type of input precisely defines the motion between video frames, making detection easier because the network does not have to calculate.

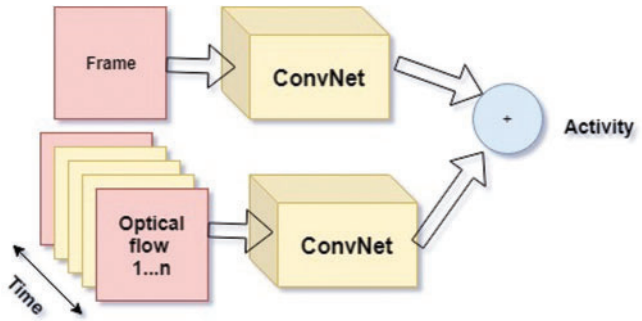


Figure 2: Typical architecture for two-stream video [11]

The two-stream method is another significant line of work in action recognition. This system was first proposed in [17] and uses the features from two sources of input i.e., (i) RGB images and (ii) Optical Flow images. The two-stream approach aims to capture the fact that time in every activity recognition process is a factor. This is achieved using optical flow data at a frame level, thus modelling low-level temporal characteristics.

2.2 Enhancements in State-of-the-art (SOTA)

The current state-of-the-art machine learning models are data-intensive and require a sufficiently large dataset for training. CNNs can detect both spatial and temporal dependencies among signals and model scale-invariant features as well. Feature extraction, dictionary learning to shape a representation for a video based on the extracted features, and final classification of the video using the definition are the three basic steps in image classification. The video classification of human activity is based on the spatial-temporal concept. It is beneficial to build the model that can automatically recognize the human activity with time information of every activity as it finds application in different domains like health monitoring, sports, etc., to compare the results based on the activity time and check if any improvement is required in the performance of the activity available at hand. In this work, numerous state-of-the-art techniques [18] designed specifically for the task of action recognition has been integrated by using the key benefits of each and by

relaxing their limitations. To explore two neural network architecture's performance in the sense of recognizing behaviour in untrimmed videos, the TSN system has been used.

Combining TSN's sparse sampling method with various CNN architectures have shown the highest accuracy for the trimmed videos. The following targets were met: (i) extend the applicability of the CNN architectures to uncut video architectures, (ii) measure the possible improvement in the accuracy rates of the TSN (relative to its baseline CNN architecture). The two alternatives to explore formally are TSN segment sampling combined with the CNN architecture R(2+1)D [19] and TSN segment sampling together with the Inflated 3D Convnet or (I3D). With time, numerous efficient image classification architectures have been generated based on the architecture of I3D, by painstaking trial and error attempts.

By pooling kernels and inflating $n \times n$ filters, it becomes $n \times n \times n$. The strategy of R(2+1)D is to approximate a 3D convolution employing a 2D convolution followed by a 1D convolution. This is the basic concept behind the stated architecture, employing a scale $t \times d \times d$ filter and an entire 3D convolution is performed, where t stands for temporal extent and d stands for spatial length and width. A R(2+1)D convolution block divides the computation into spatial 2D and temporal 1D convolutions. The R(2+1)D is formed of five (2+1) D convolution blocks. As compared to a 3D CNN with an analogous number of blocks, non-linearity is twice as large since each block contains one spatial convolution and one temporal convolution.

3 Methodology

In human activity recognition, the inputs can be of multiple forms. It can be obtained from multiple sources, which can be videos and even images having different activities of people and other signals like skeleton data that is fetched from pose estimation records, the sensory data can be fetched from a mobile or eye watch, the YouTube videos, surveillance videos and movies. Video contains rich information like space-time interaction and motion information and poses a huge challenge in terms of computation and storage for the model operating on them. The bias and overfitting to the background is quite common in the wild dataset and can be removed using the controlled collection. In the wild dataset, the data is collected from videos that were not originally designed for action recognition and mostly annotated by human annotators based on their action labels or some other meta information. The videos contain a lot of content and the variations existing in such content induces the challenge of determining the action or activity using an appropriate algorithm. Creating the datasets from YouTube videos is a methodology that is being popularly used in a lot of large-scale human activity recognition problems. Another aspect of human activity recognition is to identify the categories of activities.

A dataset containing a lot of categories as well as many samples for each category may be beneficial for designing an efficient algorithm. In case, hundreds of classes and thousands of samples per class are available then the model can learn from a large variation and perform accurate predictions. The architecture depicted in Figs. 3 and 4 represents how the proposed model works for an untrimmed video in combination with the different activities present in it. The model recognizes all the activities in a long video using the start to the end time of every activity in the video. The video under consideration is a long one and has one or more activities present in it along with some irrelevant information. The video is first passed to the proposed model, which converts it to frames and identifies the activity part based on the sequence of frames, as well as classifies the activity and time information using a CNN. The CNN uses a combination of multiple convolutional layers and feature extraction to predict every activity class as well as

the start and end time of each activity in an untrimmed video. The following is a step-by-step mathematical explanation of the working process.

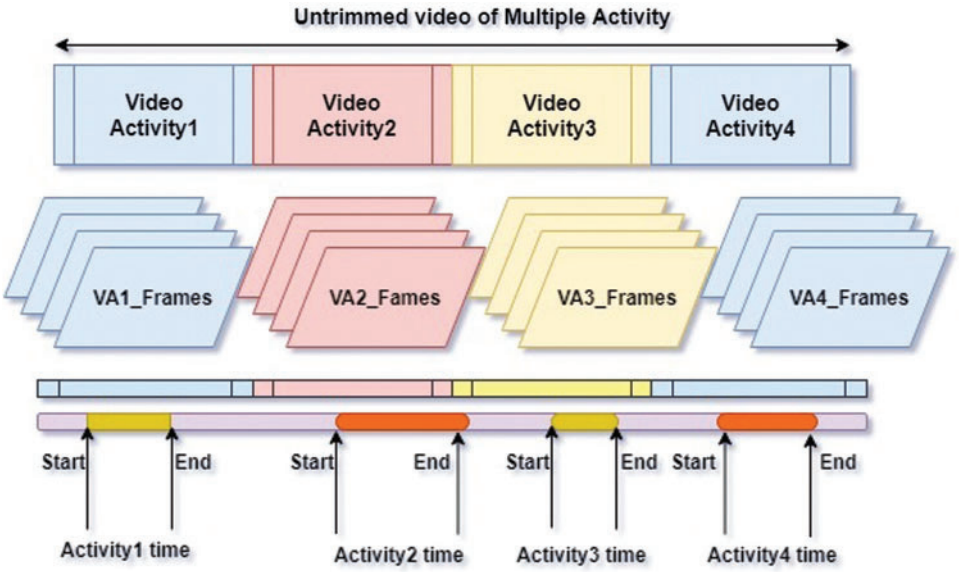


Figure 3: Process to determine the time information from the untrimmed video having different activities

For an untrimmed video V with m frames, we will denote it as $V = \{v_j\}_{j=1}^{m}$, where v_j is the j-th frame of the video. The temporal proposal generation task is to come up with a collection of proposals $Q = \{t_{si}, t_{ei}\}_{i=1}^{N_q}$ which will contain action instances for video V, where t_{si} and t_{ei} are the starting and ending time of the i^{th} proposal, and N_q is that the number of proposals. All temporal locations t_n, starting with t_{si} are either (1) greater than 0.5 max(p) or (2) at the probability peak, where max (t_s) is the maximum starting probability of the recorded video. These candidate starting points are labelled $G_s = \{t_{s,i}\}_{i=1}^{N_s}$. In the same way, we can generate ending locations to obtain the set G_E. Then, if the duration of each starting location t_s in G_S and ending location t_e in G_E is less than a pre-defined maximum duration D, it is matched as a proposal. The generated proposal is denoted as $\psi = (t_s, t_e, q^s_{t_s}, q^e_{t_s}, q_{cc}, q_{cr})$, where $q^s_{t_s}$, $q^e_{t_s}$, are the starting and ending probabilities in t_s and t_e, respectively, and q_{cc}, q_{cr} are the classification and regression confidence scores calculated from the $[t_e - t_s, t_s]$ points of the G_M confidence map M_{CC} and M_{CR} respectively.

As a result, the candidate proposals set can be obtained by $\psi = \{\psi_j\}_{j=1}^{N_q}$, where N_q is the number of candidate proposals. The algorithm for training images, intelligently picks highly activated feature maps and discards the ones that are not activated.

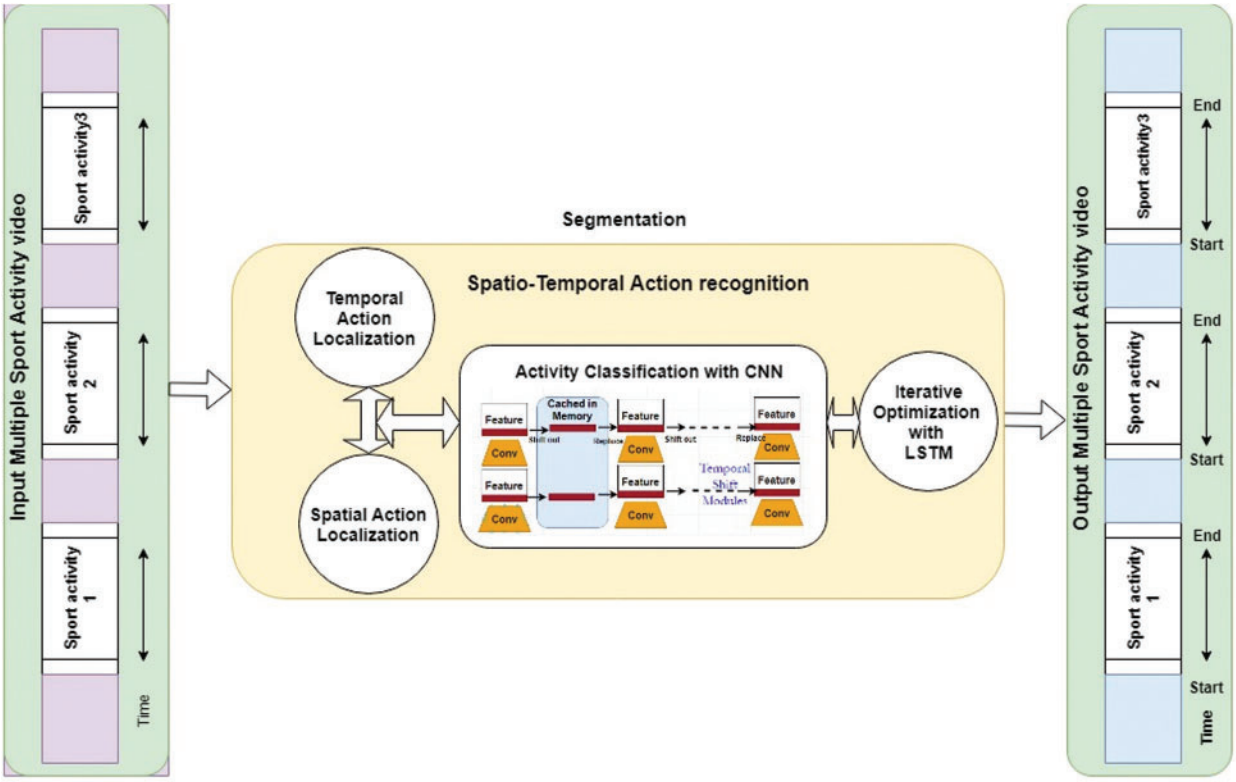


Figure 4: An overview of the proposed architecture

The algorithm's primary function is to choose only those feature maps that are highly activated and important for human regions. A global mean that indicates human activation in each feature map has been determined for each feature map. The Euclidean distance has been estimated between both the feature vectors after extracting feature vectors from two successive frames of the video stream. If the distance value exceeds a certain threshold t, the current shot is distinguished from the prior one. The threshold is determined by comparing distance measurements from several photos including both important and irrelevant visual data. Further, a step-by-step explanation of the process has been presented.

Algorithm 1: Activity recognition with temporal information

Input Video Image Dataset (VID)

Output Recognised dynamic class activity with a confidence score

Preparation
- ImageNet CNN model
- F_m feature map in 'Conv'
- Exploitation index (E_i) of size (VID) $\times F_m$
- Optical flow CNN model
- Trained multilayer LSTM

Step 1 **for** each video dataset image VID_j in VID
Feed ImageNet-CNN
Extract feature map F_m for 'conv' layer

(Continued)

Calculate global mean GM_j of each feature map F_j *to get* F_m value
Locate feature map whose GM_j=0
Mark their index with 1 in the E_i for VID_j
End for

Step 2 **while** (video set of frames)
Forward frame to ImageNet CNN model
Extract 'conv' selected feature map (step 1)
Calculate the distance between two frames to select the shot of interest (SoI)
if (SoI)
- Feed frame j and frame j+1 to ImageNet CNN
- Extract feature map from Conv. Layer
- Compute global max-pooling (GP), GP_j of each feature map F_j to obtain feature vector F_v
- F_v is input to the LSTM at timestamp 't' to save the start activity time

Predict the activity from the trained LSTM model and at the end of activity movement note the end time of activity
Display the confidence score of detected activity from video
end if
end while

3.1 Initial Basic Module

The purpose of this unit is to manage the sequence of input features, extend the receptive field, and serve as the network backbone to provide the temporal evolution phase and proposal evaluation phase with a common feature sequence. Since untrimmed videos have an unknown duration, the function cut the untrimmed series with length l_f after a longer monitoring section of length l_w. Denote an observation process as $w = t_{ws}$, t_{we}, w, f_w where t_{ws} and t_{we} the start and end time w, w and f_w are separated within the frame annotations and the sequence of functions. Depending on the dataset, the section length $l_w = \mathrm{t}_{we} - \mathrm{t}_{ws}$ is set.

3.2 Feature Extraction

The term feature extraction refers to the process of extracting the features from a video sequence. TSN [20] has been used to extract the sequence of training videos and classification of action features that use their RGB [21] and flow stream to train the model on the basis temporal shift module. To capture the temporal interactions, the compact representation of the colour feature of an image has been used.

$$\mu_c = \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N} p_{ij}^{c} \tag{1}$$

$$\partial_c = \sqrt{\left[\frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}\left(P_{ij}^{c}-\mu_c\right)^2\right]} \tag{2}$$

$$\theta_c = \sqrt{\left[\frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}\left(P_{ij}^{c}-\mu_c\right)^3\right]} \tag{3}$$

where the p_{ij}^{c} is the value of the c^{th} column component of the colour pixel in the i^{th} row and j^{th} column of an image in [22]. Eqs. (1)–(3) represents the skewness of the colour image. The 2D image considers the orientation f_{re} Laplacian operator as the second derivative. The Laplacian of theorem [23] i.e., f can be computed using Eqs. (4)–(6):

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}, \tag{4}$$

$$\frac{\partial f}{\partial x} = f(x+1, y) - f(x, y) \tag{5}$$

$$\frac{\partial^2 f}{\partial x^2} = \frac{\partial}{\partial x}\{f(x+1, y) - f(x, y)\} \tag{6}$$

$f((x+2, y) - f(x+1, y) - f(x+1, y) + f(x, y)$ this is differentiation with respect to a point $x+1$. However, if the requirement is to find the differentiation with respect to x, then 1 should be subtracted in such cases.

$$\frac{\partial^2 f}{\partial x^2} = f(x+1, y) - f(x, y) - f(x, y) + f(x-1, y) = f(x+1, y) + f(x-1, y) - 2f(x, y) \tag{7}$$

$$\frac{\partial^2 f}{\partial y^2} = f(x, y+1) + f(x, y-1) - 2f(x, y) \tag{8}$$

$$\nabla^2 f = f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1) - 4f(x, y) \tag{9}$$

To determine the mask corresponding to the Laplacian mask [24], Eqs. (7)–(9) can be used as shown in Tab. 1. It can then be applied on the image to detect the zero-crossing corresponding to the situation where pixels in the neighbourhood differ from each other in an image, i.e., $|\nabla^2 f(p)| <= |\nabla^2 f(q)|$, here p and q are the two pixels.

Table 1: A Laplacian mask Eq. (9)

0	1	0
1	**−4**	**1**
0	**1**	**0**

3.3 Module of Temporal Evaluation

The module's purpose is to establish the initial and finish points probabilities in the untrimmed video for all temporal positions. During post-processing, for the development of proposals, probability sequences of these boundaries are used. In module 1, the performance of convolution 4 layer with two sigmoid active filters begins with probability sequence P_{ws} and ends with probability sequence P_{we} for a wind observation.

3.4 Module for Proposal Evaluation

This module's objective is to produce a confidence map for boundary-matching [25,26]. The confidence scores for densely distributed proposals include the boundary-matching layer and a sequence of the convolutional layers 3D and 2D are included in the module for proposal

evaluation. In this technique, the temporal fragments are grouped into distinct sections of all training videos for an action. Then to obtain the candidate sub-actions, identical sections are combined. Finally, to obtain the final sub-actions, boundaries between the candidate sub-actions are adapted [27]. Sub-actions contained in this manner are consistent and semantically relevant. Assume that a video v is composed of fixed duration temporal segments. A Feature vector $x_i \in R_d$ computed over the characteristics, which is used as an input to the model and represents each temporal segment $l \in v$. Let n denote the number of actions and ml the sum of actions. Sub-actions ($l \in \{1, \ldots, n\}$) for action l. And let a single-hot n-dimensional vector $g_i \in R_n$ denote the action labels for segment l. If the l segment is not a part of the action, then $g_i\,(\mathrm{l}) = 0$ (zero). Otherwise 1 (one), requires $g_i(\mathrm{l})$ sub-action of the action l.

In this paper, the methodology of action is built in the classification and recognition scheme. Designing of segment and section classification methods have been performed for each of the ideas and classification measures to detect the difficulty from fine to coarse [28]. Res3D and regression network has been trained as differential segment level and frame level classifiers, respectively [29]. A group of sub-action classifiers $Tz(\cdot)(1 <= z <= ml)$ is separately trained for each action l, after obtaining the final sub-action partitions for all training images [30]. The model first detects sub-action candidates and then combines these sub-action detections to localize actions and identify temporally localized actions in the untrimmed testing images. Video action l into k clusters, k is set to 1/2 #segment in shortest learning video for action l formulated as minimization shown in Eq. (10),

$$L = \sum_{p} \sum_{i=0}^{|p|-1} \gamma\, b_{p+i}, p| \left| X_{bp+i} - C_p \right| | \tag{10}$$

where $\mathrm{b_{p+i}}\epsilon\mathrm{p}$ and $\mathrm{b_{p+}}\,|\mathrm{p}| = \mathrm{b_{p+1}}$, $\mathrm{X_{bp+i}}$ is a feature vector of i^{th} element belong to the cluster p and $\mathrm{C_p}$ is the centre of the cluster p. Most instances of the same action usually have a similar duration. The "jump" motion, for example, usually ends within 1 s, although it often takes several seconds to "clean and jerk". Assume that the duration d of a sub-action follows a Gaussian distribution $\Phi(d) \sim N(\mu, \sigma)$. The confidence value of sub-action z with duration d can be combined by comparing the prediction and duration scores. The human pose can be identified with the object key point similarity based on the map of every frame within the video using Eq. (11) [31]. Here, $\mathrm{d_i}$ is the Euclidian distance between the detected key point and the corresponding ground truth, with $\mathrm{v_i}$ visibility flag of the ground truth, s is the object scale, and k is per key point constant that control falloff. Theobject key point similarity (ObjKeySim) is the intersection over Union (IoU) equivalent ofkeypoint evaluation.

$$ObjKeySim = \frac{\sum_i exp\left(\frac{-d^2}{2s^2k_i^2}\right)\delta\,(v_i > 0)}{\sum_i \delta\,(v_i > 0)} \tag{11}$$

$$P_z\,(d) = \frac{1}{|d|} \sum_{i \in d} T_z\,(X_i)\,(\varnothing_z d) \tag{12}$$

In Eq. (12), P_z is the action detection score, $T_z\,(X_i)$ is the appearance term, $(\varnothing_z d)$ denotes the calculated duration term. The value score is determined as the classifier prediction product and score for length. The output of the SVM probability is the average from all the segments within that length. Assume that they have parts in a video for which sub-action scores have

already been determined. In this video, the objective is to decide the start and end of a sub-action section. A matrix of upper triangular detection scores is calculated. The matrix column and row indices represent the starting and ending candidate segments, respectively. The sub-action score is determined using Eq. (12) for each beginning and ending segment pair representing a candidate time. Then, optimal evaluation of the starting and ending segments for the sub-action using dynamic programming is performed.

4 Performance Evaluation and Analysis

For long videos, the proposed approach is very successful as compared to the sliding section-based methods, where the model can only calculate the low-level features once for each section. In the testing, validation, and context sets, Res3D architectures are trained. By sampling 16 frames uniformly, non-overlapped segments are produced. During the training process, each portion of the trimmed videos is marked as successful in the proposal stage. For the untrimmed videos in the validation package, positive labels are assigned to the segments of ground truth and negative labels are assigned to the segments that do not overlap with the truth. A negative mark is assigned to each context video section as well. As the dataset of the THUMOS'14 validation set is untrimmed, during the validation collection, the regression network is learnt. Using sliding sections with multiple scales as training examples, 50 percent overlapping sections are created with varying length of 2, 3,....50 segments. For a section, a positive label is allocated if: (1) The segment has a ground truth that overlaps with the most significant temporal intersection over Union (tIoU); or (2) For any ground reality, the section has a tIoU greater than 0.5. A section that has no overlap with any ground reality, a negative mark is assigned. LSTM trains the preparation, validation, and context sets. Every trimmed video in the training set as well as every action example in the training validation series are used. To mark context samples as relevant action category samples, sections of 2 to 50 segments can be created from validation and background sets. There is no overlap of sections from the validation collection with any ground fact. After the stated implementation of THUMOS'14, experimentation was performed on the ActivityNet1.3 dataset. The only difference was that there was no background selection and the videos in the training set were untrimmed. As a result, its generated sequence and category training samples from the untrimmed videos in the training and validation sets. The activity discussed in the above section was determined for untrimmed videos as shown in Figs. 4 and 5. Along with the activity determination, the start/end time of the activity based on the selected function and movement in the video has been determined and noted as the sample output of how to find the activity time information for single as well as multiple activities present in the video.

A sample illustration has been shown in Fig. 6, where the activities of swimming, cricket bowling, and high jump are defined and the graph is created with activity occurrences in an untrimmed input video as acceptable seconds. $\{s_t^r, t = 1, \ldots, T\}$ provided the frame-level scores that can be allocated as a binary label $l_t \in \{1, 0\}$ (where zero denotes the class 'background' or 'no activity') to each frame of a video of length T, which is to be maximized as shown in Eq. (13) [32]:

$$E(L) = \sum_{t=1}^{T} s_t^r - \lambda \sum_{t=2}^{T} \varphi_1(l_t, l_{t-1}) \tag{13}$$

where λ is a scalar parameter and φ_1 is the pairwise potential: $\varphi_1(l_t; l_{t-1}) = 0$ if $l_t = l_{t-1}$, else $\varphi_1(l_t; l_{t-1}) = \alpha$, where α is a parameter set by cross validation. This penalizes non-smooth labelling

$L = \{l1, .., l_T\}$ by implementing a piecewise constant solution. The desired operation proposal was formed of all contiguous sub-sequences (which are often as many as there are instances of activities). Each activity proposal received a world score S_t considering the common of its constituent frames' scores.

It is possible to use both THUMOS'14 and ActivityNet1.3 datasets for the comparative analysis. For the tasks correlated with behaviour, like identification of action, the benchmark THUMOS'14 is most used and ActivityNet1.3 is the largest current activity dataset to the best of our knowledge. Action detection task dataset THUMOS'14's plays an important role in detecting the temporal activity as it consists of 20 classes of sports behaviour. For training sub-action models, both training and validation sets have been used and the length of the segments have been fixed to 0.3 s. By a 10-fold cross-validation process, the thresholds and parameters for Gaussian distributions were learned. Following the measurement protocol for detection defined in the localization challenge for THUMOS'14 temporal actions, the untrimmed videos were included in the test set. The videos include one or more instances of actions from one or more categories of actions. In the training package, ActivityNet1.3 has 19,994 untrimmed images, a verification set, and a study set, with 200 operating categories. Res3D classifiers were compared in the experiments with segment-level recognition accuracy as a metric.

The generated videos are untrimmed, so it is difficult to determine surely that which frames correspond to the action. It is even more challenging to locate the activity time using untrimmed videos i.e., locate where the activity is present and labelled time information in the entire video instead of the individual frames. Here, the video data of multiple activities present in the video is given to the module which generates the time information for all the activities as shown in the graph of Figs. 5 and 6. Given a candidate proposal p made up of segments, $\left[(S_i)_s^e\right]$ with characteristics $\left[f_{RGB} = (fr_i)_s^e\right]$ and $\left[f_{OF} = (fo_i)_s^e\right]$, where, s and e are the indices of the beginning and end of the frame representing the activity of the proposal candidate p. fr_i and f_{OF} are the segment s_i's pool5 characteristics, respectively, as extracted by the qualified RGB and OF Res3D. The representation of the proposal candidate p is provided by the representation of the proposal candidate p.

$$\text{Concatenation}\left(A_{ve}L2\left(F_{RGB}\right), A_{ve}L2\left(F_{OF}\right)\right) \tag{14}$$

where $\text{A}_{\text{ve}}\text{L2}(F_{RGB}\&F_{OF})$ represents the process of averaging and L2-normalization [33]. The regression network takes the applicant as an input and outputs the proposal's confidence score, whether the applicant is an input proposal or an action proposal. The offsets for the temporal boundary regression can be defined as follows:

$$o_s = s_p - s_g \tag{15}$$

$$o_e = e_p - e_g \tag{16}$$

where , s_p and e_p are the proposal's first and last segment indices. s_g and e_g are the matched ground truth's starting and ending indices as presented in Eqs. (15)–(16). To train the classifier, multi-task loss L and temporal boundary regression has been used.

$$L = L_{cls} + \lambda L_{reg} \tag{17}$$

Figure 5: Untrimmed videos of multiple activities with start to end activity frames of all the activities present in the video

In Eqs. (17) and (18), L_{cls} is for the action/background classification [34], which is a normal binary softmax cross-entropy loss. L_{reg} is the loss of temporal regression, defined as:

$$\left[L_{reg} = \frac{1}{N_p} \sum_{i=1}^{N_p} l_{i(|O_s,i|+|O_i,e)} \right] \tag{18}$$

where l_i is the label, 1 indicates a relevant (action) sample and 0 indicates irrelevant (background) sample. The number of relevant samples is denoted by Np. In other way, we simply regress the relevant sample border to make the loss more resilient to the outliers by using L1's norm.

This technique has been used for the creation of training samples. In temporal data, an encoder with h_j as the hidden state at time-step j and a decoder with s_t as the unseen state at time-step t has been used. The context vector (c_t) and entire source input (X), decoder hidden state time t(s_t) can be stated as $s_t = f(s_{t-1}, y_{t-1}, c_t)$. The context vector c_t has been represented as $c_t = \sum_{j=1}^{T} \alpha_{t,j} h_j$, where $\alpha_{t,j}$ gives the degree of alignment between s_{t-1} and h_j, $\alpha\_(t,j) = \frac{\exp(score(s_{t-1}, h_j))}{\sum_{j'=1}^{T} \exp(score(s\_(t-1), hj'))}$ computes the context vector with the help of $\alpha_{t,j}$. The context vector thus generated was given as input to each time step of the decoder of recurrent neural network. The score was computed based on location-based attention i.e., $\alpha_{t,j} =$ softmax ($W_a s_t$) for THUMOS14.

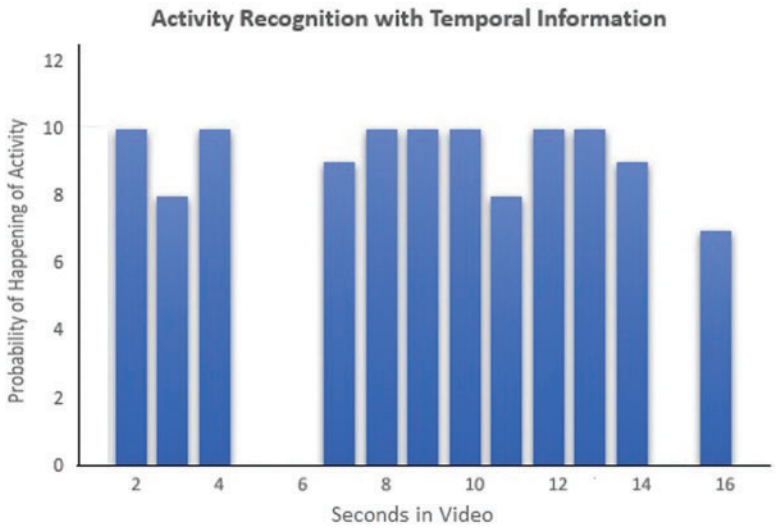


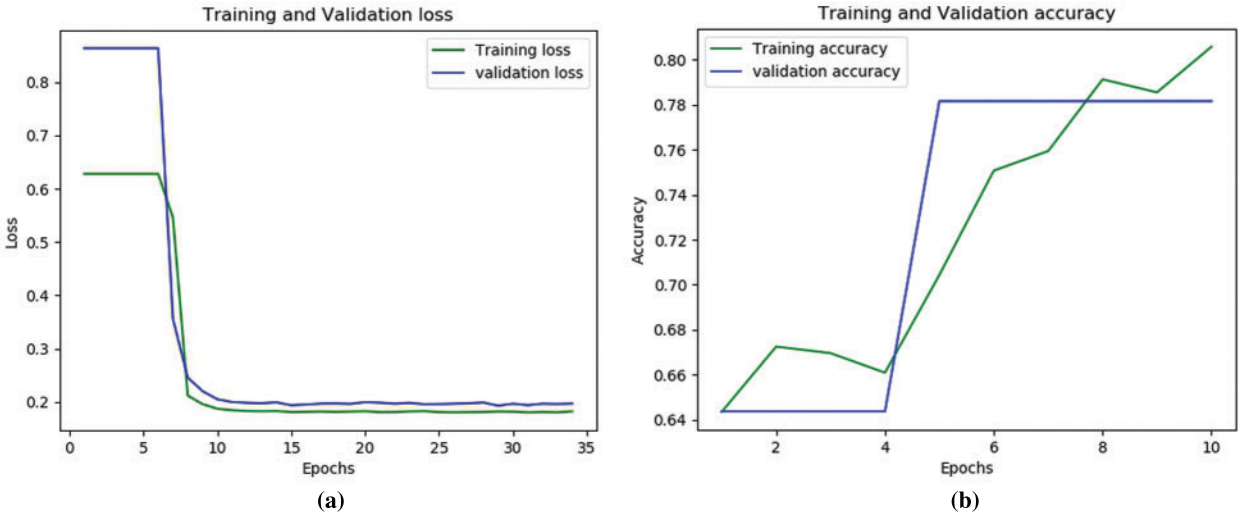
Figure 6: Activity time information from the untrimmed video in the form of start and end time

In Tab. 2, the proposed model has been compared to the state-of-the-art approaches. The proposed technique has a mathematical model that calculates the probability of overall criteria and has the highest mAP, indicating that the technique can recognize and localize the actions. The proposed framework, which uses a single model rather than an ensemble, has been able to achieve detection performance close to best performance at the IOU threshold of 0.5. With a higher IOU threshold of 0.75 and 0.95, the detectors are tested 83% to pinpoint the limits of action. Under such demanding settings, the proposed detector performs dramatically better than any competitor,

e.g., with an IOU threshold of 0.75, a mAP of 23.50% (*vs.* 23.48%) and with an IOU threshold of 0.95, a mAP of 5.96% (*vs.* 6.66%) has been obtained which is 90% close to the best in terms of the average mAP. The proposed technique is self-contained and can locate actions without the use of an external label. The applied method can still be tweaked to accommodate for external labels. To do so, all the proposals in that video are assigned to the top two video-level classes predicted by untrimmed.

Table 2: Results of action detection on ActivityNet v1.3, as measured by mean average precision (mAP) for various IoU thresholds and the average mAP of IOU thresholds ranging from 0.5 to 0.95

ActivityNet v1.3 (validation), mAP@α					ActivityNet v1.3 (testing), mAP@α				
Method	0.5	0.75	0.95	Avg	Method	0.5	0.75	0.95	Avg
DAPs [22]	22.51	–	–	–	SST [24]	42.47	2.88	0.06	14.13
SCNN-prop [23]	34.47	–	–	–	PGCN [27]	28.66	17.78	2.88	17.44
SST [24]	40.65	–	–	–	SCNN-prop [23]	36.39	11.05	0.14	17.83
BSN + Greedy-NMS	39.12 37.17	23.48 22.13	5.49 4.95	23.98 22.67	BSN + Greedy-NMS	40.68	26.01	6.66	26.05
BSN + Soft-NMS [25]									
Proposed	39.37	23.50	5.55	23.90	Proposed	40.88	18.05	5.96	24.89



(a) **(b)**

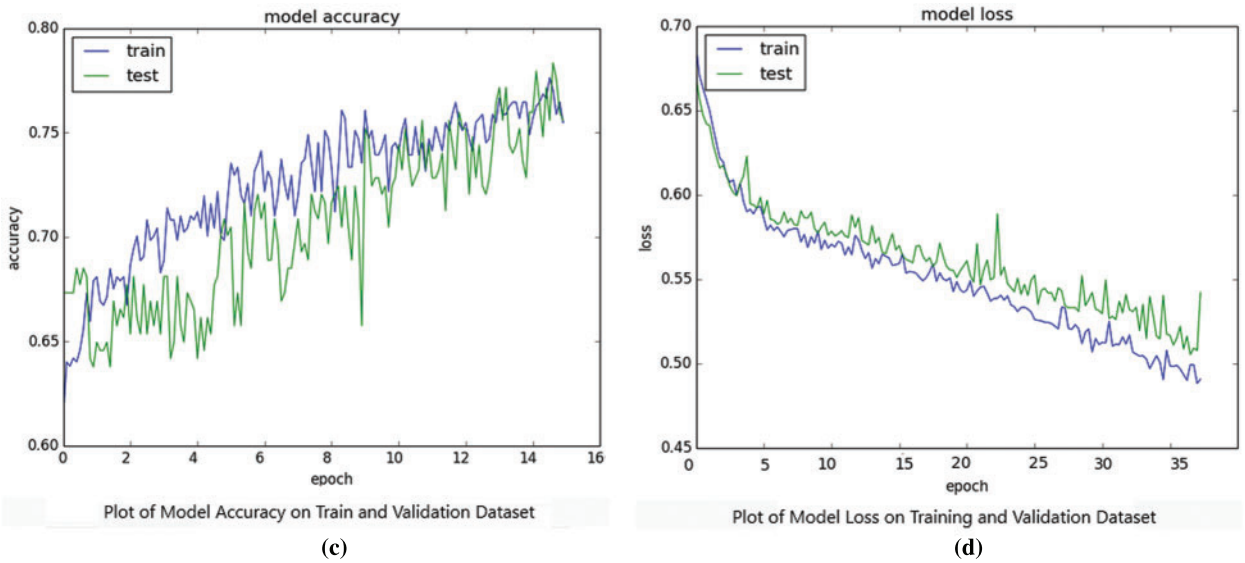


Figure 7: (a) to (d) Plots showing the training *vs.* validation accuracy/loss for different number of epochs

5 Conclusion

It has been observed that CNN with LSTM models can accurately identify the activity present as compared to RNNs, which lags the convolution method for the trimmed video data. However, the untrimmed videos help in detecting the temporal behaviour in real-time by learning the discriminatory and semantically relevant sub-actions. The number of activities in each video and the sub-actions is also immediately discovered. The state-of-the-art localization efficiency has been demonstrated on standard action datasets including temporal annotations for the action segment represented within the pipeline, start time and end time have been highlighted for the identical activity of a distinct person or different activity of various persons present within the video. The proposed approach has the potential to locate the precise temporal boundary of the instance of operation and takes into consideration the interdependency of the segments of action instance. Only the beginning and end of an activity in the video has been shown. As discussed, the proposed solution outperforms the efficiency of the state-of-the-art recognition methods for the sports domain dataset that includes broadly three activities namely swimming, cricket bowling and high jump.

Acknowledgement: The authors extend their appreciation to the Deanship of Scientific Research at King Khalid University for funding this work.

Funding Statement: This work was supported by the Deanship of Scientific Research at King Khalid University through a General Research Project under Grant Number GRP/41/42.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] S. Agahian, F. Negin and C. Kose, "Improving bag-of-poses with semi-temporal pose descriptors for skeleton-based action recognition," *The Visual Computer*, vol. 1, no. 1, pp. 591–607, 2019.

[2] F. C. Heilbron, V. Escorcia, B. Ghanem and J. C. Niebles, "ActivityNet: A large-scale video benchmark for human activity understanding," in *Proc. CVPR*, Boston, MA, USA, pp. 961–970, 2015.

[3] H. Idrees, A. Zamir, Y. Jiang, A. Gorban, I. Laptev *et al.,* "The THUMOS challenge on action recognition for videos in the wild," *Computer Vision and Image Understanding*, vol. 155, no. 1, pp. 1–23, 2017.

[4] S. N. Boualia and N. E. B. Amara, "Pose-based human activity recognition: A review," in *Proc. IWCMC*, Tangier, Morocco, pp. 1468–1475, 2019.

[5] X. Li. and M. C. Chuah, "ReHAR: Robust and efficient human activity recognition," in *Proc. WACV*, Lake Tahoe, NV, USA, pp. 362–371, 2018.

[6] J. Rafferty, C. D. Nugent, J. Liu and L. Chen, "From activity recognition to intention recognition for assisted living within smart homes," *IEEE Transactions on Human-Machine Systems*, vol. 47, no. 3, pp. 368–379, 2017.

[7] H. Ding, F. Gong, W. Gong, X. Yuan and Y. Ma, "Human activity recognition and location based on temporal analysis," *Journal of Engineering*, vol. 2018, no. 1, pp. 1–11, 2018.

[8] A. Gupta, K. Gupta, K. Gupta and K. Gupta, "A survey on human activity recognition and classification," in *Proc. ICCSP*, Chennai, India, pp. 915–919, 2020.

[9] D. R. Beddiar, B. Nini, M. Sabokrou and A. Hadid, "Vision-based human activity recognition: A survey," *Multimedia Tools and Applications*, vol. 1, no. 1, pp. 30509–30555, 2020.

[10] S. N. Muralikrishna, B. Muniyal, U. D. Acharya and R. Holla, "Enhanced human action recognition using fusion of skeletal joint dynamics and structural features," *Journal of Robotics*, vol. 1, no. 1, pp. 1–16, 2020.

[11] H. Singh, D. Kumar and A. Nasra, "IoT based real-time road traffic monitoring and tracking system for hilly regions," *International Journal of Engineering and Advanced Technology*, vol. 8, no. 5, pp. 2199–2205, 2019.

[12] A. Bevilacqua, K. MacDonald, A. Rangarej, V. Widjaya, B. Caulfield *et al.,* "Human activity recognition with convolutional neural networks," *Springer Lecture Notes in computer Science*, vol. 1, no. 1, pp. 541–552, 2019.

[13] C. Szegedy, V. Vanhoucke, S. Loffe, J. Shlens and Z. Wojna, "Rethinking the inception architecture for vision," in *Proc. CVPR*, Las Vegas, NV, USA, pp. 2818–2826, 2016.

[14] T. Lin, X. Liu, X. Li, E. Ding and S. Wen, "BMN: Boundary-matching network for temporal action proposal generation," in *Proc. ICCV*, Seoul, Korea, pp. 3889–3898, 2019.

[15] Y. L. Chang, C. S. Chan and P. Remagnino, "Action recognition on continuous video," *Neural Computing and Applications*, vol. 33, pp. 1233–1243, 2021.

[16] G. Singh and F. Cuzzolin, "Untrimmed video classification for activity detection: Submission to ActivityNet challenge," arXiv preprint arXiv:1607.01979, 2016.

[17] D. Deotale, M. Verma and S. Perumbure, "Human activity recognition in untrimmed video using deep learning for sports domain," in *Proc. ICICNIS*, Kerala, India, pp. 596–607, 2020.

[18] O. Elharrouss, N. Almaadeed, S. Al-Maadeed, A. Bouridane and A. Beghdadi, "A combined multiple action recognition and summarization for surveillance video sequences," *Applied Intelligence*, vol. 1, no. 1, pp. 690–712, 2021.

[19] D. Tran, H. Wang, L. Torresani, J. Ray, Y. Lecun *et al.,* "A closer look at spatiotemporal convolutions for action recognition," in *Proc. CVPR*, Salt Lake City, Utah, pp. 6450–6459, 2018.

[20] M. K. Bhuyan, "Applications of computer vision," in *Computer Vision and Image Processing Fundamentals and Applications*, 1st ed., vol. 1, Boca Raton, London, New York: CRC Press, pp. 312–350, 2019.

[21] D. G. Shreyas, S. Raksha and B. G. Prasad, "Implementation of an anomalous human activity recognition system," *SN Computer Science*, vol. 1, no. 3, pp. 1–10, 2020.

[22] S. Yeung, O. Russakovsky, G. Mori and L. Fei-Fei, "End-to-end learning of action detection from frame glimpses in videos," in *Proc. CVPR*, Las Vegas, NV, USA, pp. 2678–2687, 2016.

[23] F. C. Heilbron, V. Escorcia, B. Ghanem and J. C. Niebles, "ActivityNet: A large-scale video benchmark for human activity understanding," in *Proc. CVPR*, Boston, MA, USA, pp. 961–970, 2015.

[24] L. Anselma, L. Piovesan and P. Terenziani, "Temporal detection and analysis of guideline interactions," *Artificial Intelligence in Medicine*, vol. 76, no. 1, pp. 40–62, 2017.

[25] M. Tammvee and G. Anbarjafari, "Human activity recognition-based path planning for autonomous vehicles ignal, image and video processing," *Signal Image and Video Processing*, vol. 1, no. 1, pp. 1–8, 2020.

[26] S. N. Muralikrishna, B. Muniyal, U. D. Acharya and R. Holla, "Enhanced human action recognition using fusion of skeletal joint dynamics and structural features," *Journal of Robotics*, vol. 1, no. 1, pp. 1–16, 2020.

[27] U. Amin, M. Khan, D. S. Javier, W. B. Sung and H. Victor, "Activity recognition using temporal optical flow convolutional features and multilayer LSTM," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 12, pp. 9692–9702, 2019.

[28] L. Tianwei, Z. Xu, S. Haisheng, W. Chongjing and Y. Ming, "BSN: Boundary sensitive network for temporal action proposal generation," in *Proc. ECCV*, Munich, Germany, pp. 3–19, 2018.

[29] D. Tran, J. Ray, Z. Shou, S. Chang and M. Paluri, "ConvNet Architechture search for spatiotemporal feature learning," arXiv preprint arXiv: 1708.05038, 2017.

[30] U. Amin, M. Khan, T. Hussain, M. Lee and S. W. Baik, "Deep LSTM-based sequence learning approaches for action and activity recognition," in *Deep Learning in Computer Vision*, 1st Ed., Boca Raton, London, New York: CRC Press, pp. 127–150, 2020.

[31] Z. Runhao, H. Wenbing, T. Mingkui, R. Yu, Z. Peilin *et al.,* "Graph convolutional networks for temporal action localization," in *Proc. ICCV*, Seoul Korea, pp. 7094–7103, 2019.

[32] U. Amin, M. Khan, D. Wiping, P. Vasile, U. Ijaz *et al.,* "Efficient activity recognition using lightweight CNN and DS-GRU network for surveillance applications," *Applied Soft Computing Journal*, vol. 103, no. 1, pp. 1–13, 2021.

[33] X. Yuanjun, Z. Yue, W. Limin, L. Dahua and T. Xiaoou, "A pursuit of temporal accuracy in general activity detection," arXiv preprint arXiv: 1703.02716, 2017.

[34] J. Gao, Z. Yang, C. Sun, K. Chen and R. Nevatia, "TURN TAP: Temporal unit regression network for temporal action proposals," in *Proc. ICCV*, Venice, Italy, pp. 3648–3656, 2017.