International Journal of Multimedia Information Retrieval (2025) 14:18
https://doi.org/10.1007/s13735-025-00366-8

REGULAR PAPER

MMDL: a multi-modal deep learning for video highlight detection in sports

Qiaoyun Zhang[1] · Chih-Yung Chang[2] · Shih-Jung Wu[2] · Hsiang-Chuan Chang[3] · Diptendu Sinha Roy[4]

Received: 13 August 2024 / Revised: 17 January 2025 / Accepted: 3 March 2025 / Published online: 25 April 2025

Abstract
With the growing interest in sports events, the ability to capture highlights has become increasingly important. Traditionally, the process of editing these highlights required significant time and manpower. To address this challenge, this paper introduces an innovative multi-modal deep learning method for highlight detection (MMDL). The proposed MMDL integrates information from multiple modalities, including subtitles, static skeletal features, and video content, to gain a deep understanding of specific behaviors and identify sub-videos containing those highlights. Additionally, the proposed MMDL employed Siamese networks to accurately capture different aspects of behavior by comparing the similarity between input and training videos across different modalities. Experiments conducted on two datasets, MLB-YouTube and ELTA, demonstrate that the proposed MMDL significantly outperforms existing models, achieving at least a 5% improvement in F1-Score compared to the baseline models, such as I3D and NPL.

Keywords Behavior recognition · Contrastive learning · Highlight detection · Multi-modal

1 Introduction

In recent years, sports events have garnered substantial interest and attention from audiences worldwide. Apart from simply watching the games, supporters particularly enjoy revisiting the incredible highlights when star athletes accumulate millions of views but also create vast businesses to showcase their skills. These highlights not only entertain but also present more opportunities in related industries. In the past, editing such remarkable moments relied solely on the experience of video editors, requiring extensive time and manpower.

With the advancement of technology, this paper aims to utilize artificial intelligence (AI) [1–5] techniques to process sports competition videos [6–10]. The highlight can be accurately captured by recognizing the behaviors of the athletes in the video [11, 12]. The proposed mechanism can achieve automated clip editing based on predefined features, revolutionizing the video editing process to make it more efficient, precise, and cost-effective in terms of time and manpower. For example, if the user wants to extract the 'Catching' highlight from a lengthy video, this paper focuses on accurately identifying the desired highlight that corresponds. to the 'catching' action. The system then returns the localized video segment to the user. This approach enables users to precisely pinpoint and retrieve specific segments of interest within lengthy videos, significantly reducing the time and effort needed for manual searching.

Previous research in behavior recognition predominantly focused on RGB-based [13–17] and skeleton-based features

✉ Chih-Yung Chang
cychang@mail.tku.edu.tw

Qiaoyun Zhang
zqyun@chzu.edu.cn

Shih-Jung Wu
wushihjung@mail.tku.edu.tw

Hsiang-Chuan Chang
149190@o365.tku.edu.tw

Diptendu Sinha Roy
diptendu.sr@nitm.ac.in

1 School of Artificial Intelligence, Chuzhou University, Chuzhou 239000, China

2 Department of Computer Science and Information Engineering, Tamkang University, New Taipei 25137, Taiwan

3 Department of Transportation Management, Tamkang University, New Taipei 25137, Taiwan

4 Department of Computer Science and Engineering, National Institute of Technology Meghalaya, Shillong 793003, India

[18–23]. RGB-based behavior recognition methods [15–17] primarily emphasized modeling spatial and temporal representations from RGB frames and/or temporal optical flow. Despite achieving promising results, these methods have certain limitations, such as challenges in handling background clutter, dealing with illumination changes, and coping with appearance variations.

Skeleton-based features represent the structural body using a collection of coordinate positions of key joints. They exhibit viewpoint and appearance invariance, reducing intra-class variations compared to RGB-based features. Moreover, skeleton features exclusively capture high-level information describing human movements. The utilization of skeleton-based features in behavior recognition simplifies the process by removing irrelevant signals, leading to improved performance. Yao et al. [18] showed that skeleton-based features outperformed appearance-based features when applying the same classifier to the same dataset.

Some researchers [19, 20] have successfully developed well-designed multilayer Recurrent Neural Networks (RNNs) for recognizing behavior based on skeleton features. Several studies [21, 23] adopted skeleton-based behavior recognition with multi-layer Long Short-Term Memory (LSTM) [28–30]. However, while promising recognition performances, they were difficult to differentiate behavior with similar skeletal structures.

This paper proposes a novel mechanism, named MMDL, for recognizing specific behaviors. The proposed MMDL utilizes a combination of features, including subtitles, RGB, and skeleton features, to accurately identify dynamic patterns involved in highlighting. Initially, it processes draft videos that may contain highlights by analyzing the subtitles and detecting videos with static skeletal features. To accurately identify the highlight, the draft videos are segmented into many clips using a sliding window approach. Till now, it is expected that some of the candidate videos are the required highlight. To accurately identify the extract target, the proposed MMDL employs Siamese [34–36] to extract distinct features from candidate videos by focusing on prominent RGB features and processing complete skeleton features. Being the cores of the classifier, both models generate scores indicating the similarity between the input and training videos. The Siamese then compares the highlight and the candidate video to identify whether the features of the two inputs are identical. Through the design of multi-modal fusion, two-phase segmentation, and layer-wise feature matching, the proposed MMDL enhances detection accuracy while reducing computational overhead. The main contributions of the paper are summarized as follows:

1) Subtitle Generation Mechanism for Data Augmentation: The standard Bidirectional Encoder Representations from Transformers (BERT) model [25–27] does not possess the capability to classify baseball highlights. This limitation is primarily due to the lack of sufficient labeled data. To overcome this, the proposed MMDL introduces a novel subtitle generation mechanism that expands the training dataset by automatically generating varied sentence labels. This is achieved through techniques like synonym replacement, and word-order modification. This approach enables the generation of a large quantity of labeled data with minimal manual annotation, facilitating more effective training of the BERT model for behavior classification in the context of baseball highlights.
2) Multi-Modal Fusion for Comprehensive Behavior Detection: The integration of multiple modalities (subtitles, RGB, and skeleton features) is not merely a combination of existing techniques, but a carefully designed fusion that leverages the strengths of each modality. Subtitles provide rich semantic context, RGB features capture dynamic visual cues, and skeleton features offer insights into the physical actions and gestures involved. By combining these heterogeneous features, the proposed MMDL achieves a more comprehensive and nuanced understanding of video content, leading to improved behavior recognition performance.
3) Behavior Recognition through Hierarchical Feature Comparison in Siamese Networks: The proposed MMDL introduces an innovative strategy where the hidden layers of the Siamese network utilize Long Short-Term Memory (LSTM) and 3D Convolutional Neural Network (3DCNN) technologies to compare feature similarity, assessing the consistency of features across layers. This detailed matching process not only accurately identifies unique features within the data but also focuses on the nuances of behavior and action. Unlike traditional methods that compare only the final output layer, the proposed MMDL conducts feature comparisons at multiple levels, significantly enhancing the accuracy and robustness of behavior recognition.
4) Coarse-Grain and Fine-Grain Video Segmentation: To reduce computational costs and improve processing efficiency, the proposed MMDL divides video analysis into two distinct phases: coarse-grain and fine-grain. In the coarse-grain phase, it identifies video segments that are likely to contain highlights, effectively narrowing down the candidate clips. This process acts like a retriever, efficiently selecting potential highlight segments from longer videos. In the fine-grain phase, it applies a sliding-window approach and compares features across multiple models and layers to precisely segment the clips that contain the desired highlights. This two-phase approach significantly improves both speed and accuracy by tackling the challenge of overlapping video segments and refining highlight detection.

The remainder of the paper is organized as follows. Section II reviews and compares relevant previous work. Section III explains the notations, assumptions, problem description, and objectives. Section IV introduces the proposed MMDL. Section V presents simulation and performance evaluation. Section VI discusses the summary and future work.

2 Related work

In this section, the existing work closely related to the proposed MMDL mechanism is briefly reviewed. The review encompasses two categories: RGB-based and skeleton-based behavior recognition methods.

2.1 RGB-based behavior recognition

In real-world application scenarios, RGB features are commonly employed for behavior recognition due to their ease of collection and rich information content. Previous studies [15]-[17] have extensively investigated various features in RGB-based behavior recognition. Shi et al. [15] proposed a three-stream architecture consisting of a spatial stream, temporal stream, and sequential deep trajectory descriptor stream to capture spatial, short-term, and long-term features, respectively. However, their behavior recognitions were only based on RGB features, which had certain limitations. In sports competitions, the attention is more on the movements of athletes, and analyzing skeletal features is equally important.

To address data scarcity, Liu et al. [16] introduced a dataset comprising RGB and depth sensor data from multiple viewpoints for action recognition. However, the lack of contextual information limited its application in multi-person interactions or complex scenes. To address this limitation, Zhou et al. [17] introduced the Graph-based High-order Relation Modeling (GHRM) module, specifically designed for long-term action recognition in RGB videos. The GHRM module effectively captured high-order relations within long-term actions by modeling both local temporal relations and global semantic relations. This approach enhanced the recognition of long-term actions, leading to improved performance in RGB-based behavior recognition tasks. However, graph construction often involves operations and computations on large-scale graph structures, which might lead to high computational complexity.

Recent advancements in RGB-based behavior recognition have significantly enhanced multispectral pedestrian detection. Chu et al. [48] developed an illumination-guided transformer-based network that adapted to various lighting conditions, improving detection robustness. Chen et al. [49] introduced an attentive alignment network, which enhanced accuracy by aligning features across different spectral ranges, focusing on modality integration.

Additionally, Gao et al. [50] emphasized the importance of adaptability in detection systems with their work on generalizable multispectral pedestrian detection, aiming for consistent performance across diverse conditions. These studies mitigate traditional RGB-based methods' limitations by incorporating richer data to resolve issues like viewpoint variations and background clutter, thereby boosting the reliability of behavior recognition systems. Nevertheless, most RGB-based methods remain sensitive to viewpoint changes and background disturbances. To tackle these challenges, the proposed MMDL mechanism combines subtitle-based and skeleton-based processing techniques for video analysis, effectively leveraging both visual and contextual cues in the RGB data.

2.2 Skeleton-based behavior recognition

Some approaches [19–21] have utilized RNNs to recognize behavior based on skeleton features. Du et al. [19] introduced a multi-layer RNN framework for behavior recognition, employing a hierarchy of skeleton-based inputs. Wang et al. [20] leveraged the geometric relationships among joints to enhance behavior recognition and proposed a comprehensive end-to-end RNN-based network capable of incorporating joints, edges, and surfaces as inputs. Du et al. [21] presented an end-to-end hierarchical RNN for skeleton-based behavior recognition. Their mechanism partitioned the human skeleton into five main parts and employed five independent subnets to extract local features from each part. However, they may face challenges in capturing long-term dependencies, resulting in problems like vanishing or exploding gradients.

The LSTM model was considered a popular choice in behavior recognition [22–24]. Zhang et al. [22] introduced a multi-stream LSTM model for skeleton-based behavior recognition. The approach incorporated multiple geometric feature streams, and they developed a new smoothed score fusion technique to effectively learn classification from these streams. Liu et al. [23] extended the RNN design to investigate action-related information within skeleton sequences, considering the temporal and spatial aspects. Their approach incorporated a tree structure-based skeleton traversal method and utilized a gating mechanism with LSTM to address issues related to noise and occlusions. Liu et al. [24] introduced a global context-aware attention LSTM to effectively model the temporal context information within the skeleton sequences for behavior recognition. However, the behaviors recognized by these approaches were relatively regular. In baseball competitions, the image patterns of each highlight might be different depending on the event. For example, some highlight behaviors might involve highlighting the ball while

throwing, while others might include jumping up and then throwing down to catch the ball. It is challenging to accurately identify them using these methods directly.

Apart from the above-mentioned architectures, Convolution Neural Network (CNN) [37–39] has also been applied to behavior recognition. For instance, Cao et al. [37] employed selective convolutional layer activations to create a discriminative descriptor for videos, guided by body joint positions obtained from any off-the-shelf skeleton estimation algorithm. However, skeleton data lacks rich appearance information like color and texture, making it challenging to differentiate between actions with similar skeleton structures, particularly in certain situations. Skeleton data only captures joint positions, ignoring finer pose details. Thus, using skeletons for action recognition may be affected by significant pose variations.

To overcome these limitations, this paper proposes a novel approach that combines subtitle-based and skeleton-based processing techniques for video identification. Subsequently, the approach leverages multi-layer LSTM networks to process videos with complete skeleton features, which are easier to identify. By incorporating both subtitle-based and skeleton-based information, the proposed mechanism enhances the accuracy of the action recognition method.

Table 1 presents a summary of the comparisons between the proposed MMDL mechanism and the related work. The columns 'RGB,' 'Skeleton,' and 'Subtitles' indicate whether the mechanism is applied. The 'Pattern' category includes static or dynamic. The 'Temporal' and 'Spatial' categories indicate whether the mechanism considers the temporal or spatial features. The 'Models' category denotes the models adopted by each mechanism. In comparison with the related work, the proposed MMDL employs a multi-model, which exploits a variety of features from different aspects and achieves better performance in terms of precision, recall as well as F1-score for the highlights.

3 Notation, problem description, and objectives

This section explains the notations, assumptions, problem descriptions, and objectives.

3.1 Notations and assumptions

Assume that there is a baseball video $V = (\widehat{f_1}, \widehat{f_2}, \ldots, \widehat{f_m})$ which consists of m time-sequential frames, where $\widehat{f_i}$ denotes the *i-th* frame of V. It is noticed that the video can be presented if the m frames are played continuously at a certain speed. Generally, A frame typically contains the played time point, subtitles, objects, and poses in the image. Assume that $\widehat{f_i}$ can be presented by (t_i, s_i, o_i, p_i). That is, $\widehat{f_i} = (t_i, s_i, o_i, p_i)$. Notation t_i denotes the time at which the frame $\widehat{f_i}$ appears. The notation s_i denotes the subtitles that frame $\widehat{f_i}$ contains to convey important information to the viewers. Notation o_i denotes the objects that frame $\widehat{f_i}$ includes to make up the visual elements of the scene. And p_i represents the pose captured in the frame $\widehat{f_i}$.

Assume that there is a set of exercise behaviors B in baseball. Let $b \in$ denote a specific exercise behavior, such as strikeout, catching, or double play. Let $V_b = \left\{V_1^b, V_2^b, \ldots, V_{|V_b|}^b\right\}$ denote $|V_b|$ continuous frames which organize the behavior b in the video V. As shown in Fig. 1, let $V_i^b = (f_{i,1}^b, f_{i,2}^b, \ldots, f_{i,|V_i^b|}^b)$ $(1 \le i \le |V_b|)$ denote the i-th set of continuous frames which are considered as labels. Let $P_b^M = (P_1^{b,M}, P_2^{b,M}, \ldots, P_{|P_b^M|}^{b,M})$ denote predicted $|P_b^M|$ continuous frames which organize the behavior b in the video V, where $P_i^{b,M} = \{f_{i,1}^{b,M}, f_{i,2}^{b,M}, \ldots, f_{i,|P_i^{b,M}|}^{b,M}\}$ denote the predicted i-th continuous fragment using mechanism M.

Assume that function $\Phi(\cdot)$ maps the frame with the notation $f_{i,1}^b$ to the frame $\widehat{f_i}$ of V. That is,

$$\Phi\left(f_{i,1}^b\right) = \hat{f}_i$$

For a labeled frame $f_{i,|V_i^b|}^b$, the mapping can be expressed as Exp. (1).

$$\Phi\left(f_{i,|V_i^b|}^b\right) = \widehat{f}_{i+|V_i^b|-1} \tag{1}$$

Similarly, if $\left(f_{i,1}^{b,M}\right) = \widehat{f_p}$, the mapping for $f_{i,|P_i^{b,M}|}^{b,M}$ can be expressed as in Exp. (2).

$$\Phi\left(f_{i,|P_i^{b,M}|}^{b,M}\right) = \widehat{f}_{p+|P_i^{b,M}|-1} \tag{2}$$

3.2 Problem descriptions

Let $f_{i-1,i}^{mid}$ and $f_{i,i+1}^{mid}$ denote the frames mapped in the middle of V_{i-1}^b and V_i^b, V_i^b and V_{i+1}^b, respectively. Let Boolean variable $\lambda_{i,j}$ represent whether the frame $\widehat{f_j}$ is in label V_i^b. The value of $\lambda_{i,j}$ can be derived using Exp. (3).

$$\lambda_{i,j} = \begin{cases} 1, \ \hat{f}_j \in V_i^b, \ for \Phi\left(f_{i-1,i}^{mid}\right) \le \hat{f}_j \le \Phi\left(f_{i,i+1}^{mid}\right) \\ 0, otherwise \end{cases} \tag{3}$$

Let Boolean variable $\rho_{i,j}^M$ represent whether the frame f_j is in the predicted $P_i^{b,M}$. The value of $\rho_{i,j}^M$ can be derived using Exp. (4).

Table 1 The comparisons of the proposed MMDL and the related work

Method	RGB	Skeleton	Subtitles	Pattern	Temporal	Spatial	Models
Shi et al. [15]	✓	×	×	Static	✓	✓	Three-stream
Liu et al. [16]	✓	×	×	Static	✓	✓	Multiple viewpoints CNN
Zhou et al. [17]	✓	×	×	Static	✓	✓	GHRM
Du et al. [19]	×	✓	×	Static	✓	✓	RNN
Wang et al. [20]	×	✓	×	Static	✓	✓	RNN
Du et al. [21]	×	✓	×	Static	✓	✓	Hierarchical RNN
Zhang et al. [22]	×	✓	×	Static	✓	✓	Multi-stream LSTM
Liu et al. [23]	×	✓	×	Static	✓	✓	Tree-structure LSTM
Liu et al. [24]	×	✓	×	Static	✓	✓	Attention LSTM
Cao et al. [37]	×	✓	×	Dynamic	✓	✓	CNN
Wang et al. [38]	×	✓	×	Static	✓	✓	CNN
Zhang et al. [39]	×	✓	×	Static	✓	✓	CNN
Chu et al. [48]	✓	×	×	Dynamic	✓	✓	Illumination-guided transformer-based network
Chen et al. [49]	✓	×	×	Dynamic	✓	✓	Attentive alignment network
Gao et al. [50]	✓	×	×	Dynamic	✓	✓	Thermal-first and fusion-second network
Proposed MMDL	✓	✓	✓	Dynamic	✓	V	BERT + 3DCNN + LSTM + Siamese

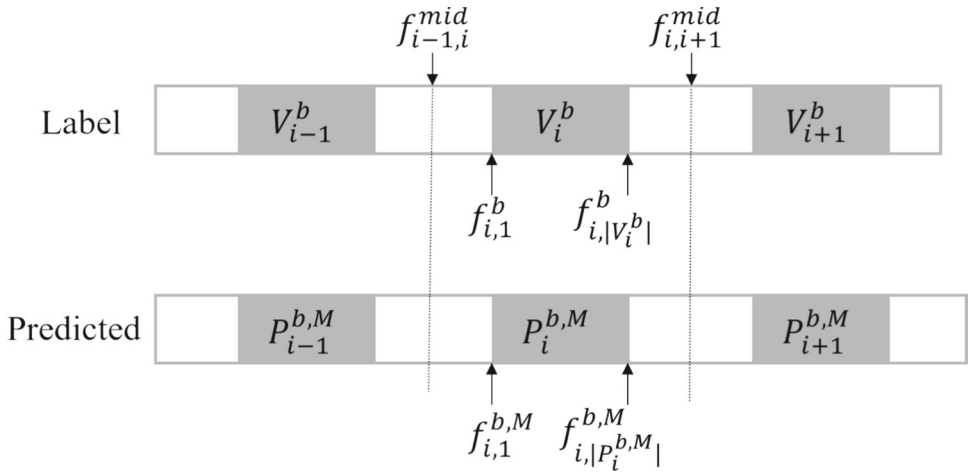


Fig. 1 The label and prediction of continuous frames which organize the behavior b

$$\rho^M_{i,j} = \begin{cases} 1, \ \hat{f}_j \in P^{b,M}_i, \ for \Phi\left(f^{mid}_{i-1,i}\right) \le \hat{f}_j \le \Phi\left(f^{mid}_{i,i+1}\right) \\ 0, \ otherwise \end{cases} \tag{4}$$

Consider an exercise behavior identification mechanism M. Let $TP^M_{i,j}$ denote the true positive of the result predicted by M. That is, the frame $\widehat{f_j}$ is predicted in i-th fragment of behavior V^b_i and the prediction is correct. The value of $TP^M_{i,j}$ can be calculated by Exp. (5).

$$TP^M_{i,j} = \begin{cases} 1, \ \rho^M_{i,j} * \lambda_{i,j} = 1, \\ 0, \quad otherwise. \end{cases} \tag{5}$$

Similarly, let $FP^M_{i,j}$ and $FN^M_{i,j}$ denote false positive and false negative of the prediction result for determining whether the frame $\widehat{f_j}$ is in i-th fragment of behavior V^b_i, respectively. The value of $FP^M_{i,j}$ and $FN^M_{i,j}$ can be calculated by Exps. (6) and (7), respectively.

$$FP^M_{i,j} = \begin{cases} 1, \ \rho^M_{i,j} - \lambda_{i,j} = 1, \\ 0, \quad otherwise, \end{cases} \tag{6}$$

$$FN^M_{i,j} = \begin{cases} 1, \ \lambda_{i,j} - \rho^M_{i,j} = 1, \\ 0, \quad \text{otherwise}, \end{cases} \tag{7}$$

$$TN_{i,j}^{M} = \begin{cases} 1, & \lambda_{i,j} + \rho_{i,j}^{M} = 0, \\ 0, & \text{otherwise.} \end{cases} \tag{8}$$

Let TP_i^M, FP_i^M, and FN_i^M denote true positive, false positive, and false negative of a set of continuous predictions for those frames starting from $\Phi\left(f_{i-1,i}^{mid}\right)$ to $\Phi\left(f_{i,i+1}^{mid}\right)$ by applying mechanism M, respectively. The values of TP_i^M, FP_i^M, and FN_i^M can be calculated by Exps. (9) to (11), respectively.

$$TP_i^M = \sum_{j=1}^{|V_i^b|} TP_{i,j}^M \tag{9}$$

$$FP_i^M = \sum_{j=1}^{|V_i^b|} FP_{i,j}^M \tag{10}$$

$$FN_i^M = \sum_{j=1}^{|V_i^b|} FN_{i,j}^M \tag{11}$$

Let TP^M, FP^M, and FN^M denote true positive, false positive, and false negative whether all frames organized the behavior b are in the video V by applying mechanism M, respectively. The values of TP^M, FP^M, and FN^M can be calculated by Exps. (12) to (14), respectively.

$$TP^M = \sum_{i=1}^{|V_b|} TP_i^M = \sum_{i=1}^{|V_b|} \sum_{j=1}^{|V_i^b|} TP_{i,j}^M \tag{12}$$

$$FP^M = \sum_{i=1}^{|V_b|} FP_i^M = \sum_{i=1}^{|V_b|} \sum_{j=1}^{|V_i^b|} FP_{i,j}^M \tag{13}$$

$$FN^M = \sum_{i=1}^{|V_b|} FN_i^M = \sum_{i=1}^{|V_b|} \sum_{j=1}^{|V_i^b|} FN_{i,j}^M \tag{14}$$

The precision and recall of behavior $b \in B$ in the video V denoted by PC^M and RC^M, respectively, can be calculated as follows:

$$\begin{aligned} PC^M &= \frac{TP^M}{TP^M + FP^M} \\ &= \sum_{i=1}^{|V_b|} \sum_{j=1}^{|V_i^b|} TP_{i,j}^M / \left(\sum_{i=1}^{|V_b|} \sum_{j=1}^{|V_i^b|} TP_{i,j}^M + \sum_{i=1}^{|V_b|} \sum_{j=1}^{|V_i^b|} FP_{i,j}^M \right) \end{aligned} \tag{15}$$

$$\begin{aligned} RC^M &= \frac{TP^M}{TP^M + FN^M} \\ &= \sum_{i=1}^{|V_b|} \sum_{j=1}^{|V_i^b|} TP_{i,j}^M / (\sum_{i=1}^{|V_b|} \sum_{j=1}^{|V_i^b|} TP_{i,j}^M + \sum_{i=1}^{|V_b|} \sum_{j=1}^{|V_i^b|} FN_{i,j}^M \end{aligned} \tag{16}$$

Let $F1^M$ denote $F1$-score of action $b \in B$ in the video V. The value of $F1^M$ can be denoted by Exp. (17).

$$F1^M = \frac{2 * PC^M * RC^M}{PC^M + RC^M} \tag{17}$$

3.3 Objectives

Let Ψ denote a set of all possible mechanisms for exercise behavior identification. The primary objective of the proposed MMDL is to find the best mechanism M_{best}^1, which satisfies the Exp (18).

Objective function 1:

$$M_{best}^1 = arg \max_{M \in \Psi} F1^M \tag{18}$$

In case there is more than one mechanism that satisfies the objective function 1, the second objective of this paper is to find the best mechanism M_{best}^2 which have a maximal interaction of union between the labeled and predicted frames. Exp. (19) presents the second objective function of the paper.

Objective function 2:

$$M_{best}^2 = \arg \max_{M \in M_{best}^1} \sum_{i=1}^{|V_b|} \frac{P_i^{b,M} \cap V_i^b}{P_i^{b,M} \cup V_i^b} \tag{19}$$

Several constraints need to be satisfied while achieving the above objective functions. The following space constraint guarantees that exercise behavior $b \in B$ should be the highlight in the video V. It can be presented by Exp. (20).

1) Space constraint

$$\exists f_j \in V_i^b, \ such\ that\ \Phi\left(f_{i,1}^b\right) \le f_j \le \Phi\left(f_{i,|V_i^b|}^b\right) \tag{20}$$

Let b and $\overline{b}$ denote two different behaviors, $b, \overline{b} \in B$. Assume that $V_i^b = \{\widehat{f}_i, \widehat{f}_{i+1}, \ldots, \widehat{f}_{i+|V_i^b|-1}\}$ and $V_j^{\overline{b}} = \{\widehat{f}_j, \widehat{f}_{j+1}, \ldots, \widehat{f}_{j+|V_j^{\overline{b}}|-1}\}$ denote the continuous frames that organize behaviors b and $\overline{b}$, respectively. Let F^b and $F^{\overline{b}}$ denote the sets of features of V_i^b and $V_j^{\overline{b}}$, respectively. The time constraint guarantees that the sets of frames V_i^b and $V_j^{\overline{b}}$ that organize different behaviors b and $\overline{b}$ should have different

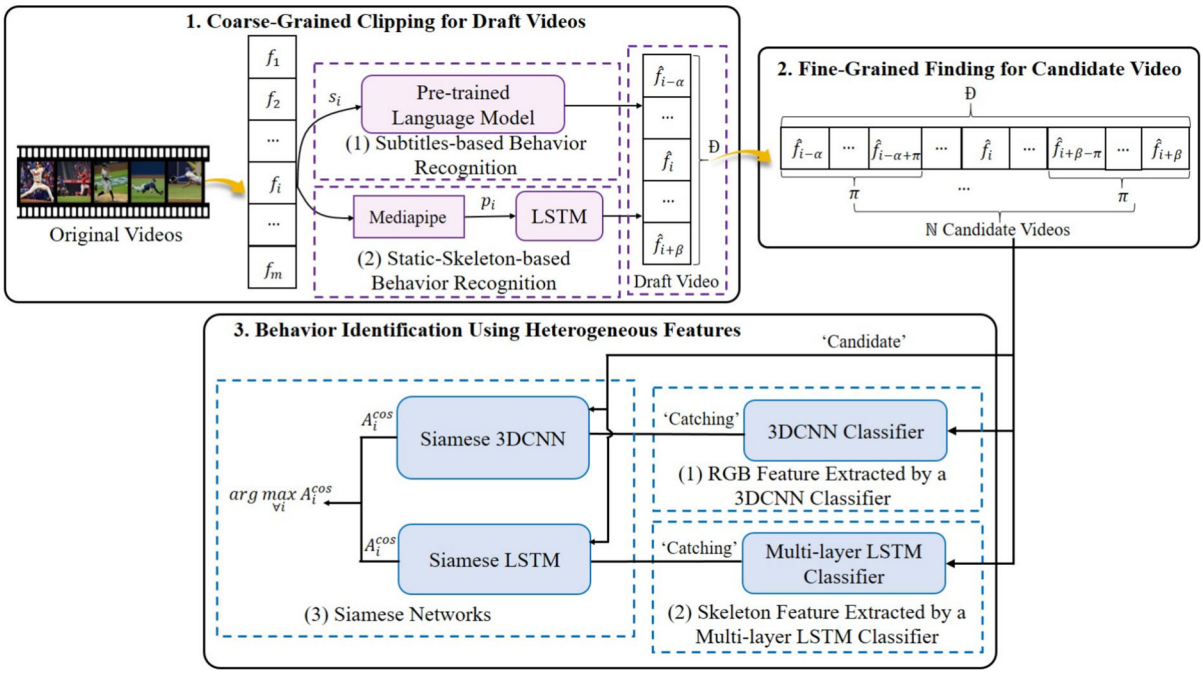


Fig. 2 The architecture of the proposed MMDL

features that can distinguish the behaviors b and$\bar{b}$.. It can be presented by Exp. (21).

2) Time constraint

$$\begin{aligned}&\exists F^b \text{ and } F^{\bar{b}} \text{such that } F^b \subseteq V_i^b \text{ and } F^{\bar{b}} \subseteq V_j^{\bar{b}}\\&F^b \cap F^{\bar{b}} = \emptyset(\forall \hat{f}_i \in V_i^b,\ for 0 \le i \le \left(i + \left|V_i^b\right| - 1\right),\\&\forall \hat{f}_j \in V_j^{\bar{b}},\ for 0 \le j \le \left(j + \left|V_j^{\bar{b}}\right| - 1\right))\end{aligned} \tag{21}$$

4 The proposed MMDL mechanism

This paper introduces the MMDL mechanism, aiming to identify specific behavior, such as 'catching' highlights, from a given video V. As shown in Fig. 2, the proposed MMDL consists of three phases, namely Coarse-Grained Clipping for Draft Videos, Fine-Grained Finding for Candidate Videos, and Behavior Identification using Heterogeneous Features. In the first phase, a pre-trained language model is employed to predict behaviors based on subtitles, while Mediapipe [47] is utilized to extract skeleton features, which are then processed through an LSTM for behavior recognition. The results from these two components are combined to generate initial draft video segments. In the second phase, these draft videos are further divided into multiple candidate video clips through fine-grained segmentation. In the third phase, the candidate clips are processed in parallel through two pathways: a Siamese 3DCNN to extract RGB features and a Siamese LSTM to extract skeleton sequence features. Each pathway employs its respective classifier for behavior recognition, and the results are fused through similarity computation to identify the best-matched behavior label. By integrating both subtitles and skeleton-based features, the model benefits from complementary information: subtitles provide contextual understanding, while skeleton features capture physical actions and gestures. This dual input enhances the model's ability to recognize behaviors with higher accuracy and robustness, as the combination of textual and visual cues allows for a more comprehensive analysis of the actions being performed.

4.1 Coarse-grained clipping for draft videos

This phase aims to create draft videos that might contain the specific behavior $b \in B$ from a given video V. It is divided into two tasks: Subtitle-based and Static-skeleton-based behavior recognitions. The first task utilizes a pre-trained language model to analyze the video's subtitles, aiming to detect keywords related to the given behavior. The second task analyzes the static skeleton to identify the specific behavior. To simplify the presentation, the following uses 'catching' as an example of exercise behavior $b \in B$. The following describes these models in detail.

1) Subtitles-based behavior recognition

This task utilizes a pre-trained language model, BERT, to capture the relationship between videos and subtitles. During

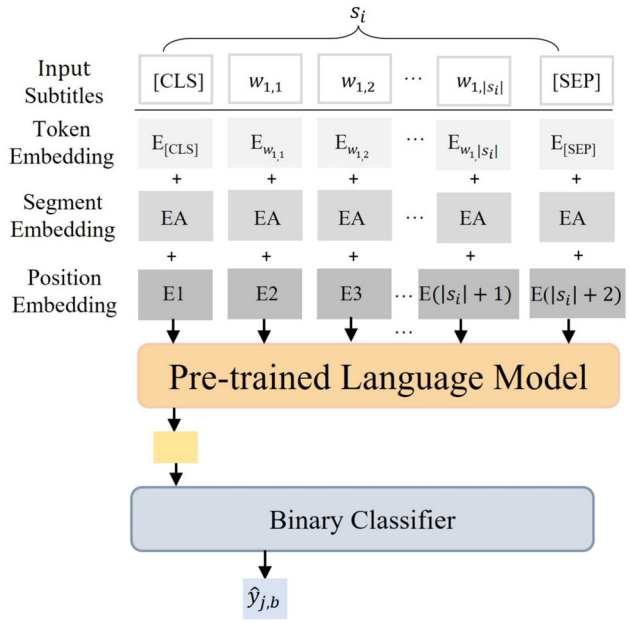


Fig. 3 The procedure of subtitles-based behavior recognition

data collecting, several videos have been labeled as 'catching'. The subtitles of these videos can be used as inputs and the output is 'catching'. Similarly, it is easy to collect some other videos which are labeled with 'not catching'. The subtitles of these videos are also used as the input of the pre-trained model, and their outputs are 'not catching'. BERT can capture contextual relationships between words in a sentence, which is crucial for accurately predicting specific behaviors such as 'catching' from subtitle text. It is fine-tuned on a labeled subtitle dataset, and its output is a binary classification indicating whether a given subtitle is associated with a behavior highlight or not. By learning the features of these subtitles, the BERT model determines whether an input subtitle corresponds to the behavior 'catching'.

The following presents how the model can learn the subtitles which are related to 'catching' behavior. Recall that s_i denotes the subtitles of the frame $\widehat{f_i}$. Let $S = s_1, s_2, ..., s_m$ denote a continuous set of subtitles of video V, where each subtitle s_i can be segmented into a set of words $(w_{i,1}, w_{i,2}, ..., w_{i,|s_i|})$ using the tool CKIP (Chinese Knowledge and Information Processing Toolkit). As shown in Fig. 3, each $s_i \in S$ is an input text. The tokens [CLS] and [SEP] are added at the beginning and end of each subtitle, respectively. Token embedding, segment embedding, and position embedding are applied to the input subtitles, resulting in the input embedding. Assume that $\backslash$ denotes the maximal input length . The value of $\backslash$ can be derived by Exp. (22).

$$n = \max_{1 \le i \le m} |s_i| \tag{22}$$

Let $X_{i,j}$ denote the word vector of $w_{i,j}$ $(1 \le i \le m, 1 \le j \le \backslash)$. Let $X_i = \{X_{i,j}\}$ denote the input matrix for s_i. Let W_Q^i, W_K^i and W_V^i denote weights matrices of query, key, and value, respectively. Let Q_i, K_i and V_i denote the queries, keys, and values matrices, respectively. They can be derived by Exps. (23) to (25), following the attention mechanism introduced by Vaswani et al. [46].

$$Q_i = X_i \times W_Q^i \tag{23}$$

$$K_i = X_i \times W_K^i \tag{24}$$

$$V_i = X_i \times W_V^i \tag{25}$$

Let O_i denote the output matrix, which combines the normalized relevance of the word $w_{i,j}$ with other words and retains its original components. It can be derived by Exp. (26).

$$O_i = \sum_{j=1}^{n} \frac{exp(Q_i^T K_j)}{\sum_{k=1}^{n} Q_i^T K_k} \times V_j \tag{26}$$

The downstream network is a binary classifier that adopts the input vectors to determine whether the output is the 'catching' behavior. During training, input sentences and their labels indicating whether they are related to 'catching' are prepared for supervised learning of the pre-trained language model and the downstream network. To correct the weights of the pre-trained language model and downstream network, the loss of the predictions should be measured. Let $\widehat{y}_{j,b}$ and $y_{j,b}$ denote the predicted and labeled behavior b, respectively. The value of $\widehat{y}_{j,b}$ can be derived from Exp. (27).

$$\hat{y}_{j,b} = Binary(O_i) \tag{27}$$

Let N_1 denote batch size. The loss function L_b of the BERT model can be defined as Exp. (28).

$$L_b(y_{j,b}, \hat{y}_{j,b}) = -\frac{1}{N_1} \sum_{j=1}^{N_1} (y_{j,b} log \hat{y}_{j,b} + (1 - y_{j,b})(1 - log \hat{y}_{j,b})) \tag{28}$$

For a given i-th pair of sentences s_i and label $y_{j,b}$, the loss of prediction $\widehat{y}_{j,b}$ should be calculated according to Exp. (27). Then the backward corrections for the weights of the pre-trained language model and downstream network are processed to learn the intelligence to determine whether the sentence s_i represents behavior b.

However, preparing videos that contain 'catching' is time-consuming. In cases where the size of the training dataset

is small, the pre-trained language model and downstream network are difficult to have high accuracies. To address this issue and generate a larger training dataset, this phase includes a generative mechanism to produce more subtitles related to 'catching'. The mechanism aims to automatically generate a variety of subtitles that contain the features of 'catching'. Let $V = \{v_1, v_2, ..., v_\varphi\}$ denote the thesaurus of 'catching'. For a subtitle s_i, the mechanism can generate φ new subtitles $\mathbb{S}_i$. Assume that $w_{i,j} = w$ is the word 'catching' in the j-th word of the sentence s_i. That is,

$$s_i = \left(w_{i,1}, \ldots, w_{i,j-1}, w, w_{i,j+1}, \ldots, w_{i,|s_i|}\right)$$

The word w of the subtitle s_i can be replaced by v_k. Let $s_{i,k}$ denote the subtitle replaced by v_k. Let $\mathbb{S}_i$ denote a set of the generated sentences labeled with 'catching'. It can be denoted by the Exp. (29).

$$\mathbb{S}_i = \{s_{i,k} | s_{i,k} = \left(w_{i,1}, \ldots, v_k, \ldots, w_{i,|s_i|}\right)\}(1 \le k \le f) \tag{29}$$

These new subtitles in the set $\mathbb{S}_i$ can be used as a training set, providing more samples to enhance the generalization capability of the pre-trained language model and the downstream networks.

2) Static-skeleton-based behavior recognition

The pre-trained language model can observe the subtitles and easily extract the features of 'catching' behavior using NLP. However, the subtitles of some 'catching' behaviors might not contain the 'catching' or its thesaurus. This leads to the loss of some catching behaviors by applying the previous scheme. To completely identify all possible sub-videos that contain 'catching' behavior, the LSTM is used to detect the sub-videos containing static skeletal features. It is well known that tools such as Openpose [40] or Google Mediapipe [41] are usually used to extract skeletal features. The extracted skeletal coordinates are used as the input of LSTM, which outputs the probability of 'catching' based on the historical skeletal features.

Recall that p_i represents the pose captured in the frame $\widehat{f_i}$. Let $P = p_1, p_2, ..., p_m$ denote a continuous set of poses of video V, where each p_i can be denoted by a set of skeletal coordinates extracted by Google Mediapipe. Let $\widehat{y_i}$ and y_i denote the predicted and labeled behaviors represented by static-skeleton features, respectively. Let N_2 denote batch size. The loss function L_s of the LSTM model can be defined as Exp. (30).

$$L_s\left(y_i, \hat{y}_i\right) = -\frac{1}{N_2}\sum_{i=1}^{N_2}(y_i log\hat{y}_i + (1-y_i)\left(1 - log\hat{y}_i\right)) \tag{30}$$

According to the above mechanisms, the draft video containing 'catching' behavior can be extracted. However, the identified frames may only have some features of 'catching' behavior, and the draft video may not be precise. To obtain a sub-video that truly represents the 'catching' behavior, forward and backward extensions are required. Assume that the identified frame is $\widehat{f_i}$, which contains 'catching' features. The extension of the draft 'catching' video is formed by including α frames before $\widehat{f_i}$ and β frames after $\widehat{f_i}$. Let V_k^b denote the draft video containing 'catching' behavior b. The range of V_k^b can be denoted by Exp. (31).

$$\hat{f}_{i-\alpha} \le \hat{f}_k \le \hat{f}_{i+\beta}\left(\hat{f}_k \in V_k^b\right) \tag{31}$$

By repeatedly applying Exp. (31), other draft videos containing the same behavior can be clipped. Similarly, this process can be applied to clip other behaviors. Therefore, the BERT and LSTM models can extract draft videos containing the 'catching' behavior based on subtitles and static skeletons. However, these extracted draft videos may not be precise representations of the full 'catching' behavior. Further processing and extensions are required to obtain more accurate sub-videos capturing the complete 'catching' behavior.

4.2 Fine-grained finding for candidate video

The length of draft videos extracted from subtitles and static skeletons may be different from the videos prepared for training. The next step is to partition the draft videos such that the length of each video is identical to that of the training video. The partitioning of draft video encounters a problem in that the starting and end frames which contain the complete catching video are unknown. If the complete catching video is partitioned into two segments, the accuracy will be adversely affected. To cope with this problem, the partitioning of draft video adopts the fine-grained sliding window technology, which ensures that the complete 'catching' video remains intact in the partitioned segment.

Let Đ and $\mathbb{C}$ denote the length of the draft and candidate videos, respectively. Let π denote the basic unit of the sliding window. By applying the sliding window technology based on the basic unit π, the draft video can be divided into multiple segments. Consequently, numerous candidate videos can be created with the sliding window approach. Let $\mathbb{N}$ denote the created number of candidate videos. It can be calculated by Exp. (32).

$$\mathbb{N} = \left[\frac{\text{Đ} - \mathbb{C}}{\pi}\right] \tag{32}$$

As shown in Fig. 4, the draft video has a length of 25 s, while the training video has a length of 18 s, which is the input

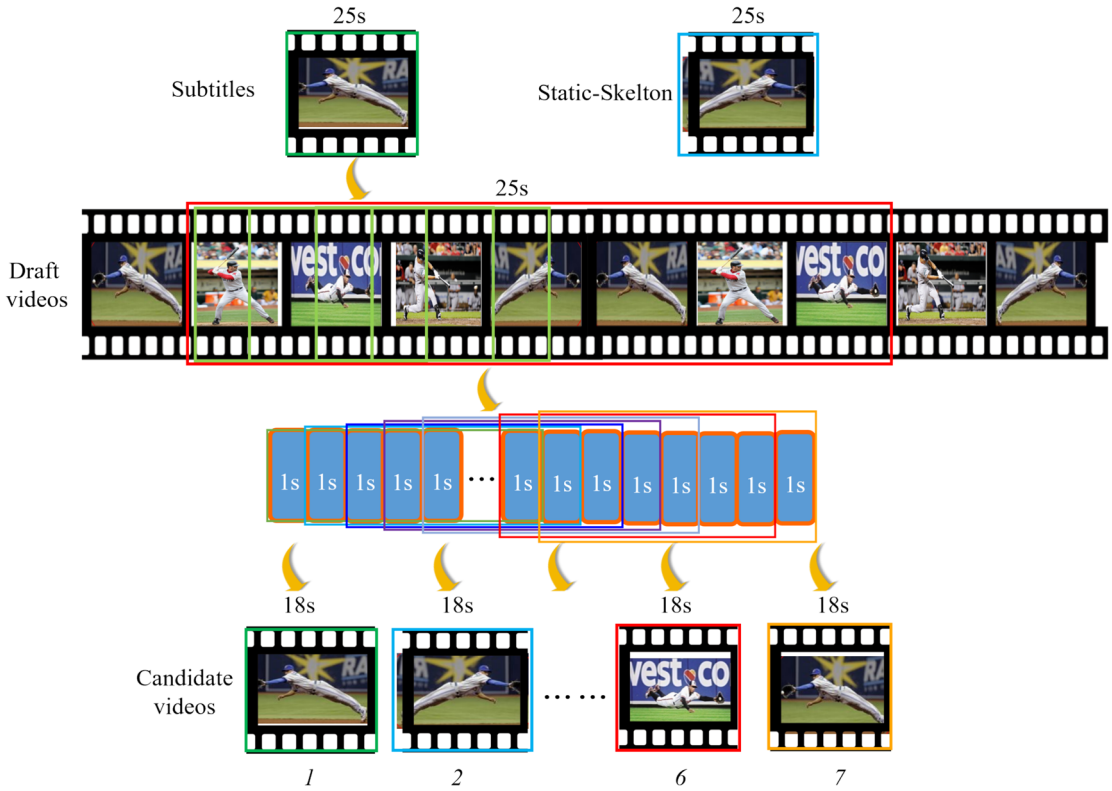


Fig.4 Capturing candidate videos from draft videos

length of the designed models. Assume that the basic sliding unit is 1 s. To guarantee that the candidate video contains the complete catching video, the mechanism divides the draft video into non-overlapped 25 partitions. Then, 18 of these partitions are selected to generate each candidate's videos. Then the starting frame shifts right one second on the draft video and continues to create the second candidate video by collecting 18 partitions. The shift right and draft video creation operations will be repeatedly performed until the right boundary of the draft video is reached. As a result, this phase has created many segments with the same length that is identical to the input length of the design model. The next phase aims to further examine which partition is the 'catching' video.

4.3 Behavior identification using heterogeneous features

Upon identifying the candidate video for 'catching' through the above two phases, this phase aims to design two models: classifiers and Siamese networks. They are constructed by 3DCNN [31–33] or LSTM based on heterogeneous features. The 3DCNN takes the RGB of the candidate video as the input while the LSTM takes the skeleton of the candidate video as its input.

The two models can extract the features of the input data and output a score for each candidate video, indicating the similarity of the features between the input data and the 'catching' video. These models are proficient in extracting heterogeneous features and operate simultaneously, complementing each other. The reason is that some candidate videos may exhibit more prominent RGB features, while other videos may have complete skeleton features that are easier to identify. Consequently, the proposed mechanism uses both RGB and skeleton features to identify 'catching' behavior and selects the most captivating video from the pool of candidate videos.

1) RGB feature extracted by a 3DCNN classifier

The RGB features are processed using the 3DCNN classifier to identify the highlights. The 3DCNN classifier comprises multiple layers, including convolution, pooling, and fully connected layers. In this approach, 3DCNN is achieved by applying a 3D kernel with the cuboid formed from stacking consecutive frames together. The notation is in line with the terminology described in [42]. However, it employs Rectified Linear Units (ReLUs) [43] instead of *tanh* as an activation function to expedite the training process. The value of a unit at position (x, y, z) in the j-th feature map of the i-th layer, is denoted by $v_{i,j}^{xyz}$, where x denotes height, y denotes width, and z denotes temporal dimension. The value of $v_{i,j}^{xyz}$ can be calculated by Exp. (33).

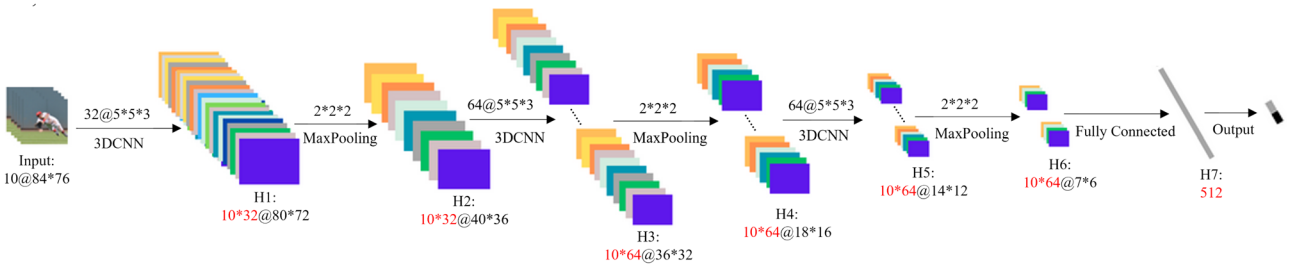


Fig. 5 A 3DCNN classifier for 'catching' behavior recognition

$$v_{i,j}^{xyz} = max\left(0, \left(b_{i,j} + \sum_{m} \sum_{p=0}^{P_i-1} \sum_{q=0}^{Q_i-1} \sum_{r=0}^{R_i-1} w_{i,j,m}^{p,q,r} v_{(i-1),m}^{(x+p),(y+q),(t+r)}\right)\right) \tag{33}$$

where $b_{i,j}$ denotes the bias for j-th feature map, $w_{i,j,m}^{p,q,r}$ denotes the value at the position (p, q, r) of the kernel connected to the $m-th$ feature map in the $(i-1)$-th layer. The dimensions of the kernel are denoted as P_i, Q_i and R_i representing height, width, and temporal dimensions, respectively. In the subsampling layers, the feature map resolution is reduced by performing pooling operations over a local neighborhood with the feature maps obtained from the previous layer. This process enhances the network's robustness to input distortions by capturing important information while reducing sensitivity to minor variations.

Figure 5 illustrates the finalized 3DCNN classifier. The mechanism utilizes a set of 10 frames, each with a size of 84*76, centered on the candidate videos as input for the 3DCNN. At the beginning of the process, the input data is passed through a 3DCNN layer with a kernel size of 5*5 and 32 feature maps, resulting in 32 feature maps of size 80*72 in layer H1. Subsequently, layer H1 applies max pooling with strides (2, 2, 2), resulting in the same number of feature maps with a reduced spatial resolution in layer H2. The output feature maps from layer H2 are then fed into layer H3, where 3DCNN with a kernel size of 5*5 is applied to the 64 feature maps. Following that, layer H4 is obtained by applying max pooling with strides (2, 2, 2), yielding 64 feature maps of size 18*16. Layer H4 further applies 3DCNN with a kernel size of 5*5 to generate layer H5, which comprises 64 feature maps of size 14*12. Continuing with the architecture, layer H5 applies max pooling with strides (2, 2, 2) to generate layer H6, consisting of 64 feature maps with a size of 7*6. Finally, all outputs from layer H6 are flattened and fed into one fully connected layer, denoted as H7, with a size of 512.

2) Skeleton feature extracted by a multi-layer LSTM classifier

The skeleton features can be processed by a multi-layer LSTM classifier to identify the highlight. Assume that there are K candidate videos with complete skeleton features. Let

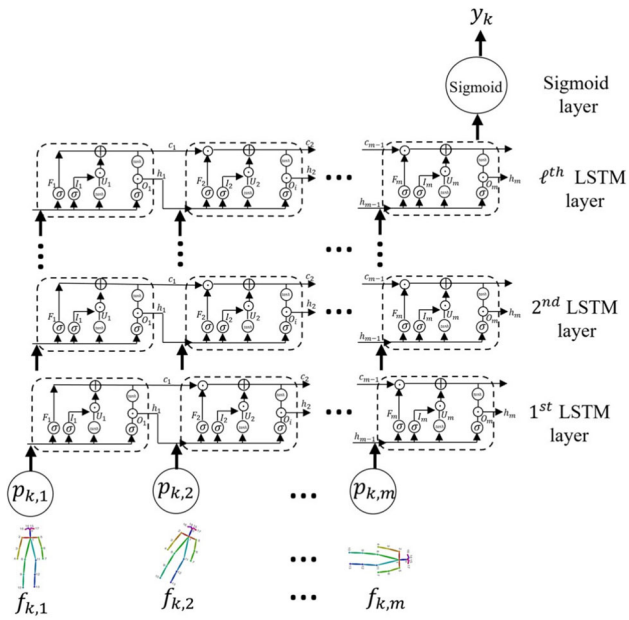


Fig. 6 The process of multi-layer LSTM to identify the skeleton features

$f_{k,1}$, $f_{k,2}$, ..., $f_{k,m}$ denote m continuous frames of $k-th$ candidate video. It is well known that each frame $f_{k,i}$ consists of several skeletal coordinates, representing the locations of the joints. Let $p_{k,i}$ denote the set of continuous skeletal coordinates of $f_{k,i}$.

As shown in Fig. 6, the 1^{st} LSTM layer comprises m LSTM cells, enabling the LSTM to take m continuous frames as its inputs, where m is the length of the time sequence. The upper-layer LSTM takes the output of the lower-layer LSTM as its input. This LSTM variation allows the higher LSTM layer to capture longer-term dependencies of the input skeleton sequences. To provide a confidence score y_k, the sigmoid function is adopted as the activation function of the last LSTM cell. The highlight, referred to as the 'catching' video, can be derived from Exp. (34).

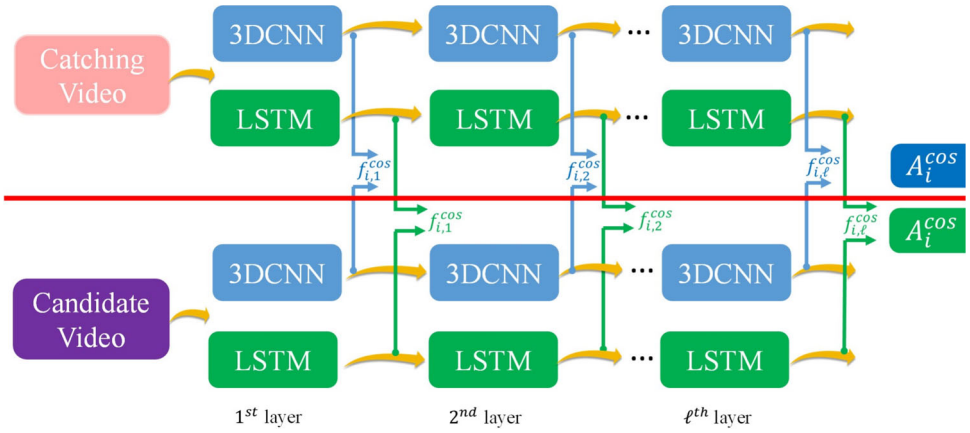


Fig. 7 The process of the Siamese model

Table 2 Parameter settings of the proposed MMDL

Parameter	Specification
CPU	i7-12,700
GPU	NVIDIA A100 GPU (80G)
CUDA	v11.7
cuDNN	v8.9.1.23
PyTorch	V2.0.0

$$arg \max_{\forall k} y_k (1 \leq k \leq K) \tag{34}$$

3) Siamese networks

Siamese networks are deep learning architectures utilized to determine the similarity degree of feature representations extracted from different candidate videos, which have the same length.

As shown in Fig. 7, the Siamese model extracts features from the 'catching' and candidate videos. Subsequently, the mechanism adopts the cosine similarity method to calculate the similarity between the candidate videos and the 'catching' video at each layer. Finally, the mechanism selects the video with the highest average cosine similarity as the 'catching' highlight. Let $V_{i,j}^C$ and V_j^L denote the one-dimensional vector of i-th candidate video at j^{th} layer and catching video, respectively. Let $f_{i,j}^{cos}$ denote cosine similarity between $V_{i,j}^C$ and V_j^L. The value of $f_{i,j}^{cos}$ can be calculated by Exp. (35).

$$f_{i,j}^{cos} = \frac{V_{i,j}^C \cdot V_j^L}{\left|\left|V_{i,j}^C\right|\right| * \left|\left|V_j^L\right|\right|} \tag{35}$$

where $(\bullet)$ denotes the vector dot product, $||V_{i,j}^C||$ and $||V_j^L||$ denote the norm of the vector $V_{i,j}^C$ and V_j^L, respectively.

Assume that there are ℓ layers in the Siamese network. Let A_i^{cos} denote the average of cosine similarity between the i-th 'catching' and candidate videos. The value of A_i^{cos} can be derived by Exp. (36).

$$A_i^{cos} = \frac{\sum_{j=1}^{\ell} f_{i,j}^{cos}}{\ell} \tag{36}$$

By selecting the candidate video with the maximum average cosine similarity, the MMDL mechanism identifies the most similar candidate video as the 'catching' highlight. It can be derived by Exp. (37).

$$arg \max_{\forall i} A_i^{cos} \tag{37}$$

5 Performance evaluation

This section aims to evaluate the performance improvement of the proposed MMDL compared with the existing related studies Neural Projection Layer (NPL) [12] and Inflated 3D (I3D) [44]. The NPL utilized a geometric convolutional layer to learn viewpoint-invariant representations from 3D representations, while I3D inflated 2D ConvNet filters and pooling kernels into 3D to capture both spatial and temporal features of videos. However, both NPL and I3D primarily focused on extracting features from RGB, which may be sensitive to view-point variations and background clutter,

potentially affecting their performance in behavior identification. In contrast, the proposed MMDL employs two modes: MMDL-LSTM (utilizing static skeleton data, multi-layer LSTM, and Siamese architecture) and MMDL-3DCNN (utilizing BERT, multi-layer 3DCNN, and Siamese architecture). This dual-modal approach enables the proposed MMDL to effectively extract both spatial and temporal features from videos. The following presents the datasets and simulation results.

Fig. 8 Sample videos from the ELTA dataset

5.1 Parameter settings and datasets

Table 2 presents the key hardware and software configurations used for executing the proposed MMDL. The implementation was carried out on a high-performance computing platform equipped with an Intel i7-12,700 CPU and an NVIDIA A100 GPU with 80 GB memory, ensuring efficient handling of computationally intensive tasks. The environment utilized CUDA version 11.7, and cuDNN version 8.9.1.23 for optimized GPU acceleration. PyTorch version 2.0.0 was utilized as the primary deep learning framework.

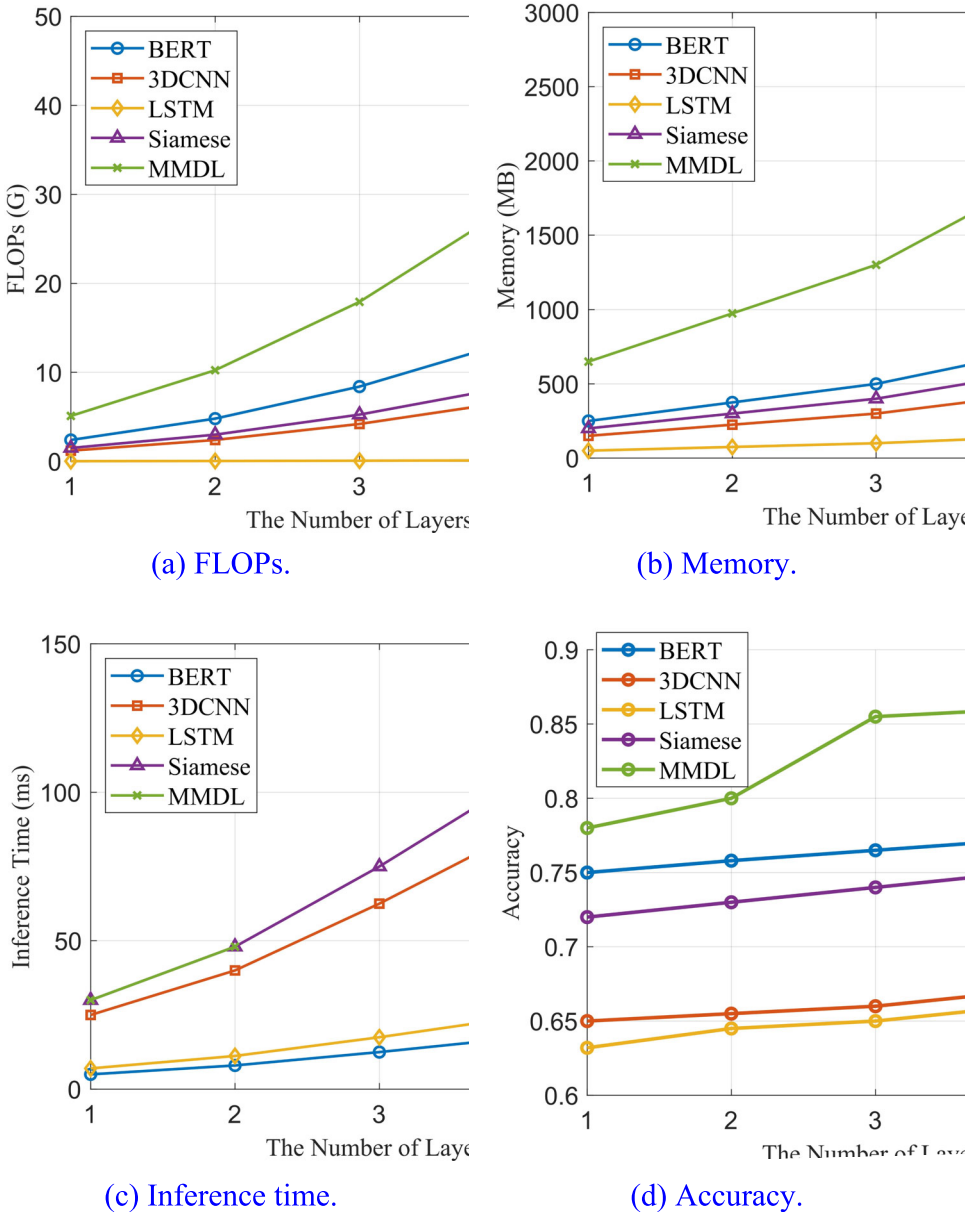
The evaluation is conducted on two datasets: MLB-YouTube, and ELTA. The MLB-YouTube dataset consists of 500 video clips centered around various baseball activities. The ELTA dataset is a custom dataset collected in collaboration with ELTA, with video clips sourced from the ELTA platform. Figure 8 showcases sample videos from this dataset. It contains approximately 800 video clips specifically collected from ELTA, with a focus on baseball activities. The use of these diverse datasets allows the proposed MVES to be thoroughly tested across various scenarios, ensuring its robustness and effectiveness in different action recognition tasks.

5.2 Simulation results

Figure 9 compares different machine learning models (BERT, 3DCNN, LSTM, Siamese, and the proposed MMDL) across varying numbers of layers in terms of Floating-Point Operations Per Second (FLOPs), memory usage, inference time, and accuracy. As the number of layers increases, all models exhibit growth in FLOPs, memory usage, and inference time, with MMDL showing a more pronounced increase compared to the others. In contrast, LSTM consistently demonstrates the lowest FLOPs, reflecting its high computational efficiency. While the proposed MMDL requires more computation and has higher inference time, it is designed to offer significant advantages in terms of accuracy in complex scenarios. The accuracy of all models shows an upward trend with the increasing number of layers, reflecting the general principle that deeper architectures tend to capture more complex features, thus improving performance.

As shown in Fig. 9, the proposed MMDL demonstrates the most significant accuracy gains across layers, outperforming all other models, particularly in scenarios requiring more complex feature representations. Although the proposed MMDL demands more computational resources and has a longer inference time, it offers substantial accuracy advantages in complex scenarios. This is particularly pertinent as the video processing is offline, where sensitivity to computation time is minimal. The proposed MMDL effectively balances FLOPs, memory usage, inference time, and accuracy to achieve state-of-the-art performance. Based on the experimental results, a three-layer configuration was selected, offering an optimal trade-off between computational efficiency and performance.

Figure 10 illustrates the F1-score variation across different thresholds for three models (I3D, NPL, and MMDL) to evaluate their classification performance. The threshold ranges from 0.1 to 1. Since the activation function sigmoid is applied in the last layer, the output value of the model is a single number ranging between 0 and 1. The output value is considered either positive or negative depending on a threshold. When the output value exceeds the threshold,



(a) FLOPs. (b) Memory. (c) Inference time. (d) Accuracy.

Fig. 9 Comparative analysis of FLOPs, memory, and inference time across different models at various number of layers

the prediction is regarded as a highlight; otherwise, it is considered a non-highlight segment. Therefore, the threshold will affect the F1-score. As shown in Fig. 10, the F1-score initially increases with the threshold in the range of 0.1 to 0.6 but gradually decreases as the threshold continues to rise. This phenomenon can be attributed to the interaction between stricter classification criteria and the distribution of confidence scores for highlight and non-highlight segments. At higher thresholds, more true highlight segments are misclassified as non-highlights, leading to an increase in false negatives. This will reduce the recall and consequently the F1-score. Among all the mechanisms compared, the proposed MMDL demonstrates significant superiority across all threshold ranges. Its outstanding performance can be attributed to its multi-modal feature fusion capability, which effectively captures both spatial and temporal features in the data, allowing it to better balance precision and recall.

Figures 11a, b, and c compare the proposed MMDL with the related work NPL and I3D in terms of precision, recall, and F1-score on the ELTA dataset. The value of the threshold varies ranging from 0.2 to 1, and the size of the dataset ranges from 100 to 500. All the compared mechanisms have the common trend that the precision is increased with the threshold. The reason is that higher thresholds make the model more conservative, resulting in fewer false positives. Interestingly, recall and F1-score initially rise but subsequently decline as the threshold increases beyond 0.6. The reason is that higher thresholds impose stricter classification criteria, which leads

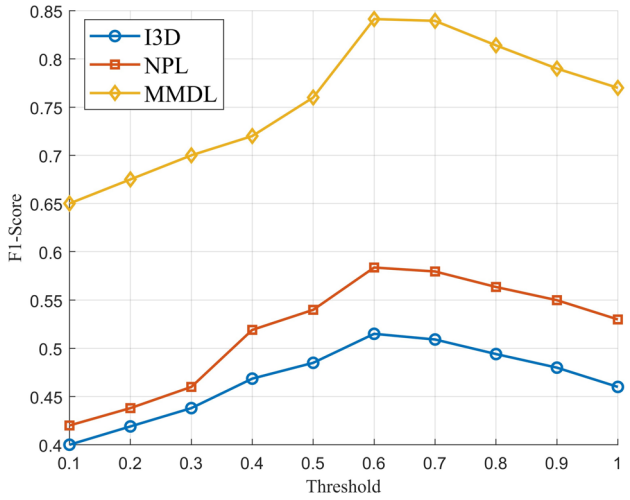


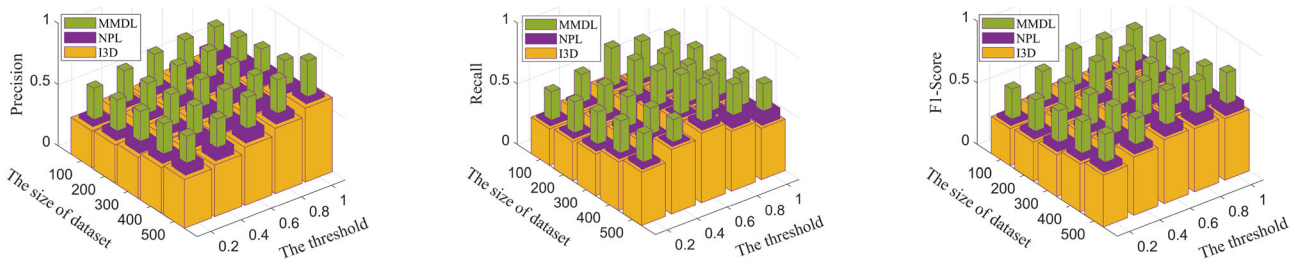
Fig. 10 F1-score variation across thresholds

to an increase in false negatives due to the overlapping distribution of confidence scores between highlight and non-highlight segments. This overlap illustrates the impact of data distribution on classification performance, where certain threshold settings exacerbate trade-offs between precision and recall.

On the other hand, the precision, recall, and F1-score are increased with the amount of data. This is because more training data helps the models to better learn and capture the features present in the dataset. In comparison, the proposed MMDL outperforms the NPL and I3D. The reason is that the MMDL considers the skeleton and RGB modalities simultaneously, which allows it to capture both spatial and temporal features from the videos. This dual-modal approach enhances the model's ability to understand and distinguish behaviors in the videos.

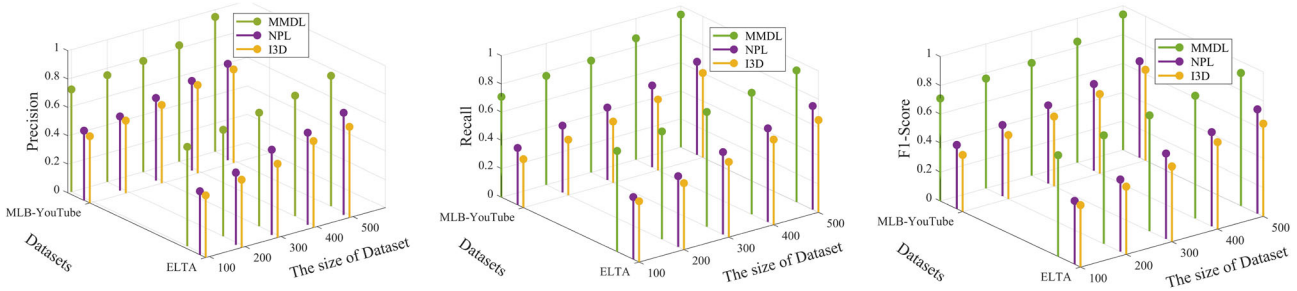
Figures 12a, b, and c compare the proposed MMDL and the related work including NPL and I3D in terms of precision, recall, and F1-Score on the MLB-YouTube and ELTA datasets. The size of the datasets ranges from 100 to 500. Across the various datasets, precision, recall, and F1-Score all show an increasing trend with the size of the dataset. This indicates that the larger number of samples in the training set allows the models to gain a better understanding of the data distribution and extract more meaningful features. Consequently, the performance of MMDL improves with increasing dataset size.

The standard BERT model does not possess the capability to classify baseball highlights. This limitation is primarily due to the lack of sufficient labeled data. To overcome this, the proposed MMDL employs a subtitle generation mechanism that expands the training dataset by automatically generating varied sentence labels. This approach enables the generation of a large quantity of labeled data with minimal manual annotation, facilitating more effective training of the BERT model for behavior classification in the context of baseball



(a) Precision (b) Recall (c) F1-Score

Fig. 11 Comparison of I3D, NPL, and the proposed MMDL by varying the threshold and dataset size



(a) Precision (b) Recall (c) F1-Score

Fig. 12 Comparison of I3D, NPL, and the proposed MMDL by varying the datasets and dataset size

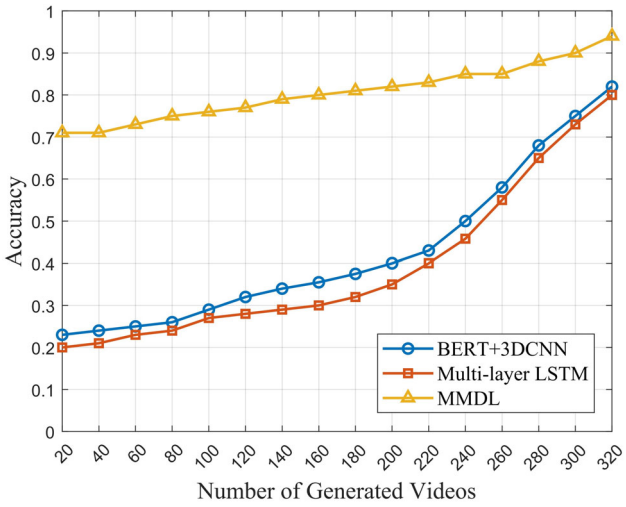


Fig. 13 The accuracy of BERT + 3DCNN, Multi-layer LSTM, and the proposed MMDL varying the number of generated videos

Table 3 Comparison of different methods on ELTA

Methods	Precision	Recall	F1-Score
3DCNN	0.445	0.435	0.44
+ LSTM	0.632	0.618	0.62
+ BERT	0.763	0.746	0.75
+ Siamese 3DCNN	0.867	0.843	0.85
+ Siamese LSTM	0.898	0.878	0.89

highlights. Figure 13 illustrates the comparison of accuracy improvement for the BERT + 3DCNN, Multi-layer LSTM, and the proposed MMDL. The number of generated videos ranges from 20 to 320. Notably, the accuracy exhibits an upward trend with an increase in the number of paired videos. The underlying reason for this trend lies in the concept that a larger quantity of training samples leads to improved performance of the model. As the number of paired videos generated increases, the model is exposed to a more extensive and diverse set of training data. This greater diversity enhances the model's ability to recognize patterns, generalize to new examples, and make accurate predictions. On the other hand, the proposed MMDL achieves high accuracy even with limited training samples, which means they can be trained with a small number of labeled pairs without the need for a large amount of labeled data. The reason is that the proposed MMDL utilizes the integration of a Siamese network and multi-modal features, which significantly enhance the model's robustness. It effectively reduces the dependency on large-scale labeled datasets, showcasing the efficiency and adaptability of the proposed MMDL.

Table 3 presents the performance results obtained from various models, including 3DCNN, LSTM, BERT, and Siamese, as well as their combinations. The evaluation is conducted based on the ELTA dataset. In comparison, the Siamese models outperform the non-Siamese models. The Siamese networks are mainly based on similarity comparison which learns the relative relationships between behaviors, not just classifying individual behaviors. This makes them more suitable for handling highlights similarities and differences. Moreover, Siamese models demonstrate better generalization compared to pure 3DCNN or LSTM, as they may perform better in recognizing new and unseen behaviors.

Table 4 The ablation study of the proposed MMDL

Mechanisms	Precision	Recall	F1-score
MMDL without Subtitle	0.63	0.62	0.62
MMDL without Skeleton	0.76	0.72	0.74
MMDL without Siamese	0.79	0.76	0.77
MMDL (Full Model)	0.89	0.88	0.88

As shown in Table 4, the ablation study demonstrates the importance of each component in improving the performance of the proposed MMDL model. Subtitle-based predictions and skeleton-based recognition are both crucial for detecting specific behaviors, while Siamese contrastive learning further enhances the model's ability to distinguish between similar behaviors. This comprehensive feature integration leads to significant improvements in model performance, as reflected in the higher precision, recall, and F1-score achieved by the full model.

Table 5 compares the proposed approach to other approaches in terms of accuracy for catching recognition. The experimental results show that the proposed approach outperforms the existing mechanisms [45] and [12]. This occurs because the proposed MMDL adopts the course-grained and fine-grained phases. In the course-grain phase, it utilizes BERT for subtitle analysis and LSTM for sub-video detection. In addition, the fine-grain phase further identifies whether the candidate videos are 'catching' highlights. To achieve this, the Siamese LSTM and 3DCNN are designed to determine the similarity of the two videos.

6 Conclusions

This paper proposes a behavior identification mechanism called MMDL, which effectively captures key highlights in baseball videos. The proposed MMDL not only integrates existing models but is also based on specific design principles aimed at enhancing multimodal operations. These principles include the automatic generation of training data using the BERT model, hierarchical feature comparisons through the Siamese network, and the application of both coarse-grain

Table 5 Comparison of other mechanisms on MLB-YouTube, ELTA, and UCF101 datasets

Mechanisms	MLB-YouTube			ELTA			UCF101		
	precision	Recall	F1-Score	precision	Recall	F1-Score	Precision	Recall	F1-Score
IDT [45]	0.283	0.264	0.273	0.403	0.378	0.39	0.339	0.322	0.33
I3D [44]	0.618	0.593	0.605	0.678	0.651	0.664	0.692	0.67	0.681
NPL [12]	0.616	0.592	0.604	0.741	0.722	0.731	0.759	0.738	0.748
MMDL (Ours)	0.8	0.78	0.79	0.897	0.884	0.89	0.802	0.781	0.791

and fine-grain strategies to precisely segment highlight clips, effectively reducing the computational load of multimodal processing. These strategies provide a solid theoretical foundation for the integration of technologies and collaboratively enhance the detection of highlight segments. Additionally, the proposed MMDL is particularly suited for handling complex action comparisons and segmentation challenges in long videos, and it effectively recognizes high-speed and precise actions in various sports videos, such as basketball, football, and tennis. The simulation results demonstrate that the proposed MMDL outperforms existing models, highlighting its effectiveness in behavior recognition tasks. Future work will focus on identifying additional specific behaviors, exploring more advanced contrastive learning methods or integrating self-supervised learning techniques to enhance the model's ability to discern subtle differences between video segments.

Acknowledgements This work was supported in part by the Smart Home and Applied Industry Innovation Team, Higher Education Research Program Project under Grant Nos. 2022AH010067 and 2023AH051609, Anhui Outstanding Youth Fund Project under Grant No. 2022AH030109 and Anhui Provincial Natural Science Foundation under Grant No. 2408085MF177.

Author contributions Conceptualization: Qiaoyun Zhang, Chih-Yung Chang; Methodology: Qiaoyun Zhang, Chih-Yung Chang, Hsiang-Chuan Chang, Shih-Jung Wu; Formal analysis: Qiaoyun Zhang, Chih-Yung Chang; Writing—original draft preparation: Qiaoyun Zhang, Chih-Yung Chang; Writing—review and editing: Shih-Jung Wu, Diptendu Sinha Roy; Experiments: Qiaoyun Zhang, Chih-Yung Chang.

Data availability No datasets were generated or analysed during the current study.

Declarations

Conflict of interest The authors declare no competing interests.

Ethical approval This study does not involve either human subjects or animals.

Informed consent All participating authors have been informed.

References

1. Farrokhi A, Farahbakhsh R et al (2021) Application of Internet of Things and artificial intelligence for smart fitness: a survey. Comput Netw 189:107859
2. Qi F, Yang X, Xu C (2020) Emotion knowledge driven video highlight detection. IEEE Trans Multimed 23:3999–4013
3. Jiao L, Zhang R et al (2022) New generation deep learning for video object detection: a survey. IEEE Trans Neural Netw Learning Syst 33(8):3195–3215
4. Xu P, Liu K et al (2021) Fine-grained instance-level sketch-based video retrieval. IEEE Trans Circuits Syst Video Technol 31(5):1995–2007
5. Zhang X, Wang T et al (2022) Multi-attention convolutional neural network for video deblurring. IEEE Trans Circuits Syst Video Technol 32(4):1986–1997
6. Rongved OAN, Hicks SA et al. (2020) Real-time detection of events in soccer videos using 3D convolutional neural networks. In: IEEE International Symposium on Multimedia (ISM), pp. 135–144
7. Li B, Xu X (2021) Application of artificial intelligence in basketball sport. J Educ, Health Sport 11(7):54–67
8. Fenil E, Manogaran G et al (2019) Real-time violence detection framework for football stadium comprising of big data analysis and deep learning through bidirectional LSTM. Comput Netw 151:191–200
9. Liu K, Liu W et al (2021) A real-time action representation with temporal encoding and deep compression. IEEE Trans Circuits Syst Video Technol 31(2):647–660
10. Wang L, Zhang H, Yuan G (2021) Big data and deep learning-based video classification model for sports. Wirel Commun Mob Comput 2021:1–11
11. Piergiovanni AJ and Ryoo MS (2018) Fine-grained activity recognition in baseball videos. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, pp. 1740–1748
12. Piergiovanni AJ and Ryoo MS (2021) Recognizing actions in videos from unseen viewpoints. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4124–4132
13. Wang P et al (2018) RGB-D-based human motion recognition with deep learning: A survey. Comput Vis Image Underst 171:118–139
14. Duan H, Zhao Y et al. (2020) Omni-sourced webly-supervised learning for video recognition. In: European Conference on Computer Vision, pp. 670–688
15. Shi Y, Tian Y et al (2017) Sequential deep trajectory descriptor for action recognition with three-stream CNN. IEEE Trans Multimed 19(7):1510–1520
16. Liu J, Shahroudy A et al (2020) NTU RGB+D: A large-scale benchmark for 3D human activity understanding. IEEE Trans Pattern Anal Mach Intell 42(10):2684–2701
17. Zhou J, Lin KY, et al. (2021) Graph-based high order relation modeling for long-term action recognition. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 8984–8993

18. Yao A, Gall J, et al. (2011) Does human action recognition benefit from pose estimation. In: British Machine Vision Conference, pp. 67.1–67.11
19. Du Y, Wang W and Wang L. (2015) Hierarchical recurrent neural network for skeleton based action recognition. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1110–1118
20. Wang H, Wang L (2018) Beyond joints: Learning representations from primitive geometries for skeleton-based action recognition and detection. IEEE Trans Image Process 27(9):4382–4394
21. Du Y, Fu Y, Wang L (2016) Representation learning of temporal dynamics for skeleton-based action recognition. IEEE Trans Image Process 25(7):3010–3022
22. Zhang S, Yang Y et al (2018) Fusing geometric features for skeleton-based action recognition using multilayer LSTM networks. IEEE Trans Multimed 20(9):2330–2343
23. Liu J, Shahroudy A et al (2018) Skeleton-based action recognition using spatio-temporal LSTM network with trust gates. IEEE Trans Pattern Anal Mach Intell 40(12):3007–3021
24. Liu J, Wang G et al (2018) Skeleton-based human action recognition with global context-aware attention LSTM networks. IEEE Trans Image Process 27(4):1586–1599
25. Devlin J, Chang MW, et al. (2019) BERT: Pre-training of deep bidirectional transformers for language understanding. In: North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 4171–4186
26. Jin W, Zhao Z et al (2021) Adaptive spatio-temporal graph enhanced vision-language representation for video QA. IEEE Trans Image Process 30:5477–5489
27. Song X, Chen J et al (2021) Spatial-temporal graphs for cross-modal text2video retrieval. IEEE Trans Multimed 24:2914–2923
28. Malhotra P, Vig L et al. (2015) Long short term memory networks for anomaly detection in time series. In: European Symposium on Artificial Neural Networks Computational Intelligence and Machine Learning(ESANN), pp. 89–94
29. Song S, Lan C et al (2018) Spatio-temporal attention-based LSTM networks for 3D action recognition and detection. IEEE Trans Image Process 27(7):3459–3471
30. Xu S, Rao H et al (2021) Attention-based multilevel co-occurrence graph convolutional LSTM for 3-D action recognition. IEEE Internet Things J 8(21):15990–16001
31. Gao Q, Chen Y et al (2022) Dynamic hand gesture recognition based on 3D hand pose estimation for human–robot interaction. IEEE Sens J 22(18):17421–17430
32. Wu D, Pigou L et al (2016) Deep dynamic neural networks for multimodal gesture segmentation and recognition. IEEE Trans Pattern Anal Mach Intell 38(8):1583–1597
33. Zhu G, Zhang L et al (2019) Continuous gesture segmentation and recognition using 3DCNN and convolutional LSTM. IEEE Trans Multimed 21(4):1011–1021
34. Chandrakanth V, Murthy VSN, Channappayya SS (2023) Siamese cross-domain tracker design for seamless tracking of targets in RGB and thermal videos. IEEE Trans Artif Intell 4(1):161–172
35. Jain H, Harit G, Sharma A (2021) Action quality assessment using siamese network-based deep metric learning. IEEE Trans Circuits Syst Video Technol 31(6):2260–2273
36. Fan C, Yu H et al (2023) Siamon: Siamese occlusion-aware network for visual tracking. IEEE Trans Circuits Syst Video Technol 33(1):186–199
37. Cao C, Zhang Y et al (2018) Body joint guided 3-D deep convolutional descriptors for action recognition. IEEE Trans Cybern 48(3):1095–1108
38. Wang X, Gao L et al (2018) Two-stream 3-D convnet fusion for action recognition in videos with arbitrary size and length. IEEE Trans Multimed 20(3):634–644
39. Zhang P, Lan C et al (2019) View adaptive neural networks for high performance skeleton-based human action recognition. IEEE Trans Pattern Anal Mach Intell 41(8):1963–1978
40. Cao Z, Hidalgo G et al (2021) OpenPose: realtime multi-person 2D pose estimation using part affinity fields. IEEE Trans Pattern Anal Mach Intell 43(1):172–186
41. Lugaresi C, Tang J et al. (2019) Mediapipe: A framework for perceiving and processing reality. In: Third Workshop on Computer Vision for AR/VR at IEEE Computer Vision and Pattern Recognition (CVPR), pp. 1–4
42. Ji S, Xu W et al (2013) 3D convolutional neural networks for human action recognition. IEEE Trans Pattern Anal Mach Intell 35(1):221–231
43. Krizhevsky A, Sutskever I, Hinton GE (2017) Imagenet classification with deep convolutional neural networks. Commun ACM 60(6):84–90
44. Carreira J and Zisserman A. (2017) Quo vadis, action recognition? A new model and the kinetics dataset. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 6299–6308
45. Wang H and Schmid C. (2013) Action recognition with improved trajectories. In: IEEE International Conference on Computer Vision, pp. 3551–3558
46. Vaswani A, Shazeer N, Parmar N et al. (2017) Attention Is All You Need. In: 31st Conference on Neural Information Processing Systems, pp. 1–11
47. Lugaresi C, Tang J, et al. (2019) Mediapipe: A framework for perceiving and processing reality, In: IEEE Computer Vision and Pattern Recognition (CVPR), pp. 1–4
48. Chu F, Cao J, Shao Z et al. (2022) Illumination-guided transformer-based network for multispectral pedestrian detection. In: International Conference on Artificial Intelligence, pp. 343–355
49. Chen N, Xie J, Nie J, et al. (2023) Attentive alignment network for multispectral pedestrian detection. In: Proceedings of the 31st ACM International Conference on Multimedia, pp. 3787–3795
50. Gao A, Pang Y, Nie J et al (2024) Toward generalizable multispectral pedestrian detection. IEEE Trans Intell Transp Syst 25(5):3739–3750