CNNs, RNNs and Transformers in human action recognition: a survey and a hybrid model

Khaled Alomar[1] · Halil Ibrahim Aysel[2] · Xiaohao Cai[2]

© The Author(s) 2025

Abstract

Human action recognition (HAR) encompasses the task of monitoring human activities across various domains, including but not limited to medical, educational, entertainment, visual surveillance, video retrieval, and the identification of anomalous activities. Over the past decade, the field of HAR has witnessed substantial progress by leveraging convolutional neural networks (CNNs) and recurrent neural networks (RNNs) to effectively extract and comprehend intricate information, thereby enhancing the overall performance of HAR systems. Recently, the domain of computer vision has witnessed the emergence of Vision Transformers (ViTs) as a potent solution. The efficacy of Transformer architecture has been validated beyond the confines of image analysis, extending their applicability to diverse video-related tasks. Notably, within this landscape, the research community has shown keen interest in HAR, acknowledging its manifold utility and widespread adoption across various domains. However, HAR remains a challenging task due to variations in human motion, occlusions, viewpoint differences, background clutter, and the need for efficient spatio-temporal feature extraction. Additionally, the trade-off between computational efficiency and recognition accuracy remains a significant obstacle, particularly with the adoption of deep learning models requiring extensive training data and resources. This article aims to present an encompassing survey that focuses on CNNs and the evolution of RNNs to ViTs given their importance in the domain of HAR. By conducting a thorough examination of existing literature and exploring emerging trends, this study undertakes a critical analysis and synthesis of the accumulated knowledge in this field. Additionally, it investigates the ongoing efforts to develop hybrid approaches. Following this direction, this article presents a novel hybrid model that seeks to integrate the inherent strengths of CNNs and ViTs.

Keywords Human action recognition · Convolutional neural networks · Recurrent neural networks · Vision transformers · Deep learning · Video classification

---

Extended author information available on the last page of the article

1 Introduction

Human action recognition (HAR) focuses on the classification of the specific actions exhibited within a given video. On the other hand, action detection and segmentation focus on the precise localisation or extraction of individual instances of actions from video content (Ulhaq et al. 2022). The capacity of deep learning models to effectively capture the spatial and temporal complexities inherent in video representations plays a vital role in the recognition and understanding of actions.

Over the preceding decade, a considerable amount of research has been dedicated to the thorough investigation of action recognition, resulting in an extensive collection of review articles and survey papers addressing the topic (Pareek and Thakkar 2021; Sun et al. 2022; Kong and Fu 2022). However, it is worth noting that a predominant focus of these scholarly works has been placed on the examination and evaluation of convolutional neural networks (CNNs) and traditional machine learning models within the realm of action recognition.

The advent of Transformer architecture (Vaswani et al. 2017) has sparked a paradigm shift in deep learning. By employing a multi-head self-attention layer, the Transformer model computes sequence representations by effectively aligning words within the sequence with other words in the same sequence (Ulhaq et al. 2022). This approach outperforms traditional convolutional and recursive operations in terms of representation quality while utilizing fewer computational resources. As a consequence, the Transformer architecture diverges from conventional convolutional and recursive methods, favoring a more focused utilization of multiple processing nodes. The incorporation of multi-head attention allows the Transformer model to collectively learn a range of representations from diverse perspectives through the collaboration of multiple attention layers. Inspired by Transformers, many natural language processing (NLP) tasks have achieved remarkable performance, reaching human-level capabilities, as exemplified by models such as GPT (Brown et al. 2020) and BERT (Devlin et al. 2019).

The remarkable achievements of Transformers in handling sequential data, particularly in the domain of NLP, have prompted the exploration and advancement of Vision Transformer (ViT) (Dosovitskiy et al. 2020) (a special Transformer for computer vision tasks). ViTs have demonstrated comparable or even superior performance compared to CNNs in the context of image recognition tasks, especially when operating on vast datasets such as ImageNet (Han et al. 2022; Lin et al. 2022; Khan et al. 2022). This observation signifies a noteworthy shift in the field, wherein ViTs possess the potential to supplant the established dominance of CNNs in computer vision, mirroring the displacement witnessed in the case of recurrent neural networks (RNNs) (Ulhaq et al. 2022). The achievements of Transformer models have engendered considerable scholarly interest within the computer vision research community, prompting rigorous exploration of their efficacy in pure computer vision tasks.

The natural progression in the advancement of ViTs has led to the logical exploration of video recognition tasks. Unlike image recognition, video recognition focuses on the complex challenge of identifying and understanding events within video sequences, including the recognition of human actions. Consequently, there is a compelling need for a recent review that comprehensively examines the state-of-the-art research including ViTs and hybrid models in addition to CNNs and RNNs for HAR. Such a review would serve as a crucial guiding resource to shape the future research directions with Transformer and CNN-

Transformer hybrid architectures beside CNNs which previously were seen as unique and influential models for HAR. The main contributions of this paper are as follows.

- We present a thorough review of the CNNs, RNNs and ViTs. This review examines the evolution from traditional methods to the latest advancements in neural network architectures.
- We present an extensive examination of existing literature related to HAR.
- We propose a novel hybrid model integrating the strengths of CNNs and ViTs. In addition, we provide a detailed performance comparison of the proposed hybrid model against existing models. The analysis highlights the model's efficacy in handling complex HAR tasks with improved accuracy and efficiency.
- We also discuss emerging trends and the future direction of HAR technologies, emphasizing the importance of hybrid models in enhancing the interpretability and robustness of HAR systems.

These contributions enrich the understanding of the current state and future prospects of HAR, proposing innovative approaches and highlighting the importance of integrating different neural network architectures to advance the field.

The paper is structured as follows. Section 2 delves into the background, covering foundational concepts and technologies crucial to HAR, including CNNs, RNNs and ViTs, highlighting the chronological evolution of HAR deep learning technologies. Section 3 thoroughly reviews related HAR works with a brief discussion. A novel hybrid model combining CNNs and ViTs is proposed in Sect. 4, including the details of the experimental setup and the results. Section 5 discusses the challenges and their implications for future directions in HAR. Finally, Sect. 6 concludes the paper.

2 Background

This section provides a chronological and technical overview of three fundamental types of neural networks: CNNs, RNNs, and Transformers. CNNs, introduced in the late 1980 s, revolutionized image processing by leveraging local connectivity and shared weights to efficiently detect spatial hierarchies in data. As the field progressed, RNNs emerged in the 1990 s, addressing the need for modeling sequential data through their ability to maintain temporal dependencies across sequences. The advent of Transformers in 2017 marked a paradigm shift by utilizing self-attention mechanisms to capture global relationships in data more effectively, thereby enhancing performance in a wide array of tasks beyond sequential data. This background section will delve into the technical intricacies and evolutionary trajectory of these architectures, highlighting their contributions and transitions in the realm of deep learning.

2.1 CNNs

The evolution of CNNs has been remarkable since their introduction in the 1980 s. Originally, CNNs were designed to process static images, primarily focusing on spatial recognition tasks such as object and pattern recognition. The initial idea was to build layers of

convolutional filters that would apply various operations to the image to extract features like edges, textures, and shapes. This structure proved highly effective for tasks like image classification, object detection, image segmentation and more in computer vision.

The Neocognitron (Fukushima 1980), developed by Kunihiko Fukushima, presented an early example of neural networks incorporating convolutional operations for image processing, setting the foundations for subsequent progress. Later, Yann LeCun and collaborators introduced LeNet-5 (LeCun et al. 1998), a key architecture designed for handwritten digit recognition, showcasing the effectiveness of convolutional layers in pattern recognition tasks. The progress of CNNs reached a turning point in the mid-2010 s with the introduction of models like AlexNet (Krizhevsky et al. 2012), showcasing their potential in image classification tasks. Alongside architectural innovations, this milestone was achieved thanks to access to large datasets, notably, ImageNet (Deng et al. 2009), and computational improvements, including the rise of graphics processing units (GPUs) for parallel computing. Large-scale datasets provided the diversity and complexity necessary for training deep networks, while enhanced computational power accelerated the training of sophisticated CNN architectures.

The architectural enhancements, large datasets, and increased computational capabilities helped CNNs to be a cornerstone in deep learning methodologies, extending their applications beyond image processing to various domains. Notable architectures like VGGNet (Simonyan and Zisserman 2014a), distinguished by its uniform design and small convolutional filters, GoogLeNet (Szegedy et al. 2015), with its inception modules for capturing features at different scales efficiently, and ResNet (He et al. 2016), which introduced residual learning for training very deep networks, have further enriched the landscape of CNNs.

2.1.1 Spatio-temporal CNNs

As CNNs excelled in spatial tasks, researchers began exploring their potential in handling temporal data, such as video and time-series analysis. The challenge was to incorporate the dimension of time into the inherently spatial architecture of CNNs. To address this task, spatio-temporal CNNs were developed. These networks extend traditional CNN architectures by adding a temporal component to analyze dynamic behaviors across time frames. Several approaches have been utilized and main types are as follows.

3D convolution involves extending the 2D kernels to 3D, allowing the network to perform convolution across both spatial and temporal dimensions. This approach is directly applied to video data where the third dimension represents time (Hara et al. 2018; Tran et al. 2015). The two-stream CNNs involve running two parallel CNN streams: one for spatial processing of individual frames and another for temporal processing, usually of optical flow, which captures motion between frames (Simonyan and Zisserman 2014a; Feichtenhofer et al. 2016). RNNs with CNNs aim to combine CNNs for spatial processing with RNNs like long short-term memory (LSTM) or gated recurrent unit (GRU) to handle temporal dependencies. This hybrid model leverages CNNs' ability to extract spatial features and RNNs' capacity to manage temporal sequences effectively (Yue-Hei Ng et al. 2015; Donahue et al. 2015).

2.2 From vanilla RNN to attention-based transformers

This section explores the evolution from RNNs to the Transformers, highlighting the progression in handling time series and sequence data. Initially, RNNs were the go-to deep learning technique for managing temporal tasks, effectively capturing sequential dependencies. However, the development of Transformers marked a significant leap forward, driven by a series of iterative improvements and optimizations that built upon the limitations of RNNs. Transformers, with their focus on NLP, introduced a novel attention mechanism that allows for more efficient and scalable processing of sequential data. By examining the foundational RNN techniques and the subsequent enhancements leading to the Transformer architecture, this section elucidates the transformative journey from traditional RNN models to the sophisticated attention-based frameworks that now dominate the field.

We firstly establish common notations for RNN architectures including vanilla RNNs, LSTM and GRU to streamline discussions in subsequent sections. In these architectures, each iteration involves a cell that sequentially processes an input embedding $\boldsymbol{x}_t \in \mathbb{R}^{n_x}$ and retains information from the previous sequence through the hidden state $\boldsymbol{h}_{t-1} \in \mathbb{R}^{n_h}$ using weight matrices $\boldsymbol{W} \in \mathbb{R}^{n_h \times n_h}$ and $\boldsymbol{U} \in \mathbb{R}^{n_h \times n_x}$. The $\boldsymbol{W}$-like matrices encompass all weights related to hidden-to-hidden connections, while $\boldsymbol{U}$-like matrices encompass all weight matrices related to input-to-hidden connections. Additionally, bias terms are represented by $\boldsymbol{b}$-like vectors. Each cell produces a new hidden state $\boldsymbol{h}_t \in \mathbb{R}^{n_h}$ as its output. More details about symbols and variables used in this section are given in Table 1.

2.2.1 Vanilla RNNs

Vanilla RNNs (Rumelhart et al. 1985; Jordan 1986) lack the presence of a cell state, relying solely on the hidden states as the primary means of memory retention within the RNN framework. The hidden state $\boldsymbol{h}_t$ is subsequently updated and propagated to the subsequent cell, or alternatively, depending on the specific task at hand, it can be employed to generate a prediction. Figure 1a illustrates the internal mechanisms of an RNN and a mathematical description of it given as

$$\boldsymbol{h}_t = \tanh(\boldsymbol{W}\boldsymbol{h}_{t-1} + \boldsymbol{U}\boldsymbol{x}_t + \boldsymbol{b}), \tag{1}$$

where tanh is the activation function.

Vanilla RNNs effectively incorporate short-term dependencies of temporal order and past inputs in a meaningful manner. However, they are characterized by certain limitations. Firstly, due to their intrinsic sequential nature, RNNs pose challenges in parallelized computations (Graves et al. 2013). Consequently, this limitation can impose restrictions on the overall speed and scalability of the network. Secondly, when processing lengthy sequences, the issue of exploding or vanishing gradients may arise, thereby impeding the stable training of the network (Bengio et al. 1994).
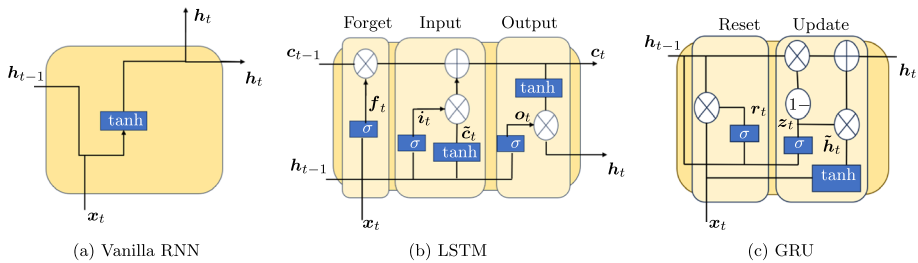
2.2.2 LSTM

Hochreiter and Schmidhuber (1997) introduced the LSTM cell as a solution to address the issue of long-term dependencies and to mitigate the challenge of interdependencies among

Table 1 List of mathematical symbols and variables used in Sect. 2

Symbol	Definition
$x_t \in \mathbb{R}^{n_x}$	Input embedding at time t
$h_t \in \mathbb{R}^{n_h}$	Hidden state at time t
$W \in \mathbb{R}^{n_h \times n_h}$	Weight matrix for hidden-to-hidden connections
$U \in \mathbb{R}^{n_h \times n_x}$	Weight matrix for input-to-hidden connections
$b \in \mathbb{R}^{n_h}$	Bias vector
$i_t \in \mathbb{R}^{n_h}$	Output of the sigmoid activation function at time t in the input gate in LSTM cell
$o_t \in \mathbb{R}^{n_h}$	Output of the output gate at time t in LSTM cell
$c_t \in \mathbb{R}^{n_h}$	Cell state at time t in LSTM cell
$\tilde{c}_t \in \mathbb{R}^{n_h}$	Candidate cell state at time t in LSTM cell
$z_t \in \mathbb{R}^{n_h}$	Output of the update gate in GRU at time t
$r_t \in \mathbb{R}^{n_h}$	Output of the reset gate in GRU at time t
$\tilde{h}_t \in \mathbb{R}^{n_h}$	Candidate hidden state in GRU at time t
$W_f \in \mathbb{R}^{n_h \times n_h}$	Weight matrix for forget gate in LSTM cell
$U_f \in \mathbb{R}^{n_h \times n_x}$	Weight matrix for forget gate input in LSTM cell
$b_f \in \mathbb{R}^{n_h}$	Bias for forget gate in LSTM cell
$W_i \in \mathbb{R}^{n_h \times n_h}$	Weight matrix for input gate in LSTM cell
$U_i \in \mathbb{R}^{n_h \times n_x}$	Weight matrix for input gate input in LSTM cell
$b_i \in \mathbb{R}^{n_h}$	Bias for input gate in LSTM cell
$W_o \in \mathbb{R}^{n_h \times n_h}$	Weight matrix for output gate in LSTM cell
$U_o \in \mathbb{R}^{n_h \times n_x}$	Weight matrix for output gate input in LSTM cell
$b_o \in \mathbb{R}^{n_h}$	Bias for output gate in LSTM cell
$W_{\tilde{c}} \in \mathbb{R}^{n_h \times n_h}$	Weight matrix for candidate cell state in GRU cell
$U_{\tilde{c}} \in \mathbb{R}^{n_h \times n_x}$	Weight matrix for candidate cell state input in GRU cell
$b_{\tilde{c}} \in \mathbb{R}^{n_h}$	Bias for candidate cell state in GRU cell
$W_z \in \mathbb{R}^{n_h \times n_h}$	Weight matrix for update gate in GRU cell
$U_z \in \mathbb{R}^{n_h \times n_x}$	Weight matrix for update gate input in GRU cell
$b_z \in \mathbb{R}^{n_h}$	Bias for update gate in GRU cell
$W_r \in \mathbb{R}^{n_h \times n_h}$	Weight matrix for reset gate in GRU cell
$U_r \in \mathbb{R}^{n_h \times n_x}$	Weight matrix for reset gate input in GRU cell
$b_r \in \mathbb{R}^{n_h}$	Bias for reset gate in GRU cell
$W_{\tilde{h}} \in \mathbb{R}^{n_h \times n_h}$	Weight matrix for candidate hidden state in GRU cell
$U_{\tilde{h}} \in \mathbb{R}^{n_h \times n_x}$	Weight matrix for candidate hidden state input in GRU cell
$b_{\tilde{h}} \in \mathbb{R}^{n_h}$	Bias for candidate hidden state in GRU cell
$d_k \in \mathbb{N}$	Dimension of the keys
$Q \in \mathbb{R}^{n_x \times d_k}$	A set of query vectors
$K \in \mathbb{R}^{n_x \times d_k}$	A set of key vectors
$V \in \mathbb{R}^{n_x \times d_k}$	A set of value vectors
$X \in \mathbb{R}^{n_x \times d_x}$	Input matrix (sequence of embeddings)
$W^Q \in \mathbb{R}^{d_x \times d_k}$	Weight matrix for queries
$W^K \in \mathbb{R}^{d_x \times d_k}$	Weight matrix for keys
$W^V \in \mathbb{R}^{d_x \times d_k}$	Weight matrix for values
$A \in \mathbb{R}^{n_x \times d_v}$	Attention output
$Q_i \in \mathbb{R}^{n_x \times d_k}$	Query matrix for the i-th attention head

Table 1 (continued)

Symbol	Definition
$\boldsymbol{K}_i \in \mathbb{R}^{n_x \times d_k}$	Key matrix for the i-th attention head
$\boldsymbol{V}_i \in \mathbb{R}^{n_x \times d_k}$	Value matrix for the i-th attention head
$\boldsymbol{W}_i^Q \in \mathbb{R}^{d_x \times d_k}$	Weight matrix for queries in the i-th attention head
$\boldsymbol{W}_i^K \in \mathbb{R}^{d_x \times d_k}$	Weight matrix for keys in the i-th attention head
$\boldsymbol{W}_i^V \in \mathbb{R}^{d_x \times d_k}$	Weight matrix for values in the i-th attention head
$\boldsymbol{A}_i \in \mathbb{R}^{n_x \times d_v}$	Attention output for the i-th attention head

(a) Vanilla RNN   (b) LSTM   (c) GRU

Fig. 1 Various types of RNN cells

successive steps (Hochreiter and Schmidhuber 1997). LSTM architecture incorporates a distinct component known as the cell state $\boldsymbol{c}_t \in \mathbb{R}^{n_h}$, illustrated in Fig. 1b. Analogous to a freeway, this cell state facilitates the smooth flow of information, ensuring that it can readily traverse without undergoing significant alterations.

Gers et al. (2000) made modifications to the initial LSTM architecture by incorporating a forget gate within the cell structure. The mathematical expressions describing this modified LSTM cell are derived from its inner connections. Hence, the LSTM cell can be formally represented based on the depicted interconnections as follows.

- Forget gate decides what information should be thrown away or kept from the cell state with the equation

$$\boldsymbol{f}_t = \sigma(\boldsymbol{W}_f \boldsymbol{h}_{t-1} + \boldsymbol{U}_f \boldsymbol{x}_t + \boldsymbol{b}_f), \tag{2}$$

where $\boldsymbol{f}_t \in \mathbb{R}^{n_h}$ is the output of the forget gate and σ is the sigmoid activation function.

- Input gate determines which new information is added to the cell state with two activation functions defined as

$$\boldsymbol{i}_t = \sigma(\boldsymbol{W}_i \boldsymbol{h}_{t-1} + \boldsymbol{U}_i \boldsymbol{x}_t + \boldsymbol{b}_i), \tag{3}$$

where $\boldsymbol{i}_t \in \mathbb{R}^{n_h}$ is the output of the sigmoid activation function; and

$$\tilde{\boldsymbol{c}}_t = \tanh(\boldsymbol{W}_{\tilde{c}} \boldsymbol{h}_{t-1} + \boldsymbol{U}_{\tilde{c}} \boldsymbol{x}_t + \boldsymbol{b}_{\tilde{c}}), \tag{4}$$

where $\tilde{c}_t \in \mathbb{R}^{n_h}$ is known as candidate value. After obtaining i_t and $\tilde{c}_t$, we can update the cell state with

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t, \tag{5}$$

where $c_{t-1} \in \mathbb{R}^{n_h}$ is the previous cell state and $\odot$ is the Hadamard operator.

- Output gate determines the next hidden state based on the cell state and output gate's activity

$$o_t = \sigma(W_o h_{t-1} + U_o x_t + b_o), \tag{6}$$

where $o_t \in \mathbb{R}^{n_h}$ is the output of the output gate. Finally the updated hidden state,

$$h_t = \tanh(c_t) \odot o_t \tag{7}$$

is fed to the next iteration.

To enable selective information retention, LSTM employs three distinct gates. The first gate, known as the forget gate, examines the previous hidden state h_{t-1} and the current input x_t. It generates a vector f_t containing values between 0 and 1, determining the portion of information to discard from the previous cell state c_{t-1}. The second gate, referred to as the input gate, follows a similar process to the forget gate. However, instead of discarding information, it utilizes the output i_t to determine the new information to be stored in the cell state based on a candidate cell state $\tilde{c}_t$. Lastly, the output gate employs the output o_t to filter the updated cell state c_t, thereby transforming it into the new hidden state h_t. The LSTM cell exhibits superior performance in retaining both long-term and short-term memory compared to the vanilla RNN cell. However, this advantage comes at the expense of increased complexity.

2.2.3 GRU

The LSTM cell surpasses the learning capability of the conventional recurrent cell, yet the additional number of parameters escalates the computational load. Consequently, to address this concern, Chung et al. (2014) introduced the GRU, see Fig. 1c. GRU demonstrates comparable performance to LSTM while offering a more computationally efficient design with fewer weights. This is achieved by merging the cell state and the hidden state into "reset state" resulting in a simplified architecture. Furthermore, GRU combines the forget and input gates into an "update gate", contributing to a more streamlined computational process. For further elaboration, GRU cell incorporates two essential gates. The first gate is the reset gate, which examines the previous hidden state h_{t-1} and the current input x_t. It generates a vector r_t containing values between 0 and 1, determining the extent to which past information in h_{t-1} should be disregarded. The second gate is the update gate, which governs the selection of information to either retain or discard when updating the new hidden state h_t, based on the value of r_t.

Based on the depicted information in Fig. 1c, the mathematical expressions governing the behavior of the GRU cell can be expressed as follows.

- Update gate decides how much of the past information needs to be passed along with

$$z_t = \sigma(W_z h_{t-1} + U_z x_t + b_z),\tag{8}$$

where $z_t \in \mathbb{R}^{n_h}$ is the output of the update gate. The output of the reset gate $r_t \in \mathbb{R}^{n_h}$ is obtained by

$$r_t = \sigma(W_r h_{t-1} + U_r x_t + b_r).\tag{9}$$

A candidate activation for the subsequent step is

$$\tilde{h}_t = \tanh(W_{\tilde{h}}(r_t \odot h_{t-1}) + U_{\tilde{h}} x_t + b_{\tilde{h}})\tag{10}$$

where $\tilde{h}_t \in \mathbb{R}^{n_h}$.

- The final activation is a blend of the previous hidden state and the candidate activation, weighted by the update gate, i.e.,

$$h_t = z_t \odot \tilde{h}_t + (1 - z_t) \odot h_{t-1}\tag{11}$$

where $h_t \in \mathbb{R}^{n_h}$ is the updated hidden state. This mechanism allows the GRU to effectively retain or replace old information with new information.

2.2.4 Types of RNNs

RNNs were created with an internal memory mechanism that allows them to store and use information from previous outputs. This unique trait enables RNNs to retain important contextual information over time, enabling reasoned decision-making based on past results. There are four types of popular RNN variants that each serve different purposes across a variety of applications, see Fig. 2. For simplicity, x_i and y_i respectively represent the input and output with $i = 1, \ldots, t$ in Fig. 2.

The one-to-one is considered the simplest form of RNNs, where a single input corresponds to a single output. It operates with fixed input and output sizes, functioning similarly to a standard neural network. One-to-many represents a specific category of RNNs that is characterized by its ability to produce multiple outputs based on a single input provided to the model. This type of RNN is particularly useful in applications like image captioning, where a fixed input size results in a series of data outputs. Many-to-one RNNs merge a
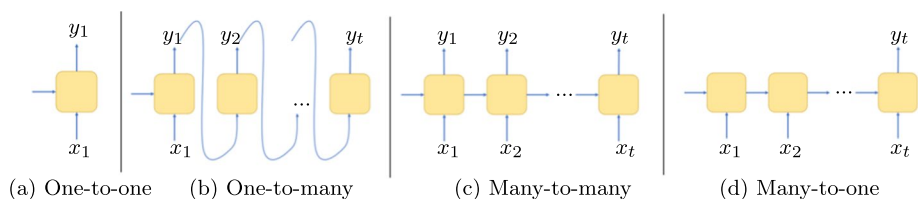

(a) One-to-one   (b) One-to-many   (c) Many-to-many   (d) Many-to-one

Fig. 2 Types of RNN structures based on input–output pairs. Here x_i and y_i, $i = 1, \ldots, t$, represent the input and output, respectively

sequence of inputs into a single output through a series of hidden layers that learn relevant features. An illustrative instance of this RNN type is sentiment analysis, where the model analyzes a sequence of text inputs and produces a single output indicating the sentiment expressed in the text.

Many-to-many RNNs are employed to generate a sequence of output data from a sequence of input units. It can be categorized into two subcategories: equal size and unequal size. In the equal size subcategory, the input and output layers have the same size, see many-to-many architecture in Fig. 2c. Several research efforts have emerged to tackle the limitation of the fixed-size input–output sequences in machine translation tasks, as they fail to adequately represent real-world requirements. The unequal size subcategory can handle different sizes of inputs and outputs. A practical application of the unequal size subcategory can be observed in machine translation. In this scenario, the model generates a sequence of translated text outputs based on a sequence of input sentences. Unequal size subcategory employs an encoder-decoder architecture, where the encoder adopts the many-to-one architecture, and the decoder adopts the one-to-many architecture. One notable contribution in this area was made by Kalchbrenner and Blunsom (2013), who pioneered the approach of mapping the entire input sentence to a vector. This work is related to the study conducted by Cho et al. (2014), although the latter was specifically utilized to refine hypotheses generated by a phrase-based system (Sutskever et al. 2014). In this architecture, the encoder component plays a crucial role in transforming the inputs into a singular vector, commonly referred to as the context. This context vector, typically with a length of 256, 512 or 1024, encapsulates all the pertinent information detected by the encoder from the input sentence, which serves as the translation target, see Fig. 3a. Subsequently, this vector is passed on to the decoder, which generates the corresponding output sequence. It is important to note that both the encoder and decoder components in this architecture are RNNs. Different from Fig. 3a, b gives the encoder-decoder architecture with attention which will be introduced in the next section.

2.2.5 Attention

The evolution of attention mechanisms in neural networks represents a significant advancement in the field of deep learning, particularly in tasks related to NLP and machine translation. Initially introduced by Graves (2013), the concept of attention mechanisms was designed to enhance the model's ability to focus on specific parts of the input sequence when generating an output, mimicking the human ability to concentrate on particular aspects of a
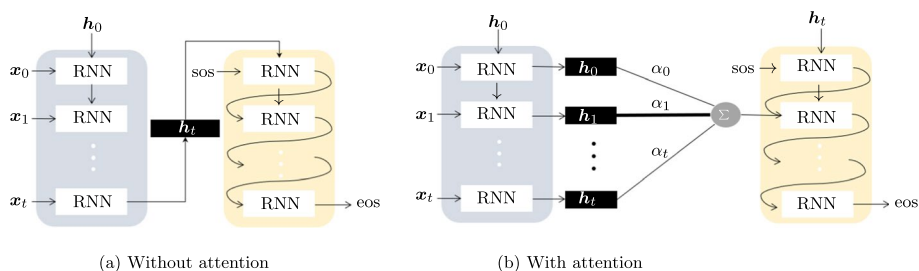

(a) Without attention   (b) With attention

Fig. 3 Sequence-to-sequence RNN with and without the attention mechanism. Here $\alpha_i, i = 1, \ldots, t$, are the attention weights

task. This foundational work laid the groundwork for subsequent developments in attention mechanisms, providing a mechanism for models to dynamically assign importance to different parts of the input data.

Building on Graves' initial concept, Bahdanau et al. (2014) introduced the additive attention mechanism, which was specifically designed to improve machine translation. This approach computes the attention weights through a feed-forward neural network, allowing the model to consider the entire input sequence and determine the relevance of each part when translating a segment. This additive form of attention significantly improved the performance of sequence-to-sequence models by enabling a more nuanced understanding and alignment between the input and output sequences (Sutskever et al. 2014). Following this, Luong et al. (2015) proposed the multiplicative attention mechanism, also known as dot-product attention, which simplifies the computation of attention weights by calculating the dot product between the query and all keys. This method not only streamlined the attention mechanism but also offered improvements in computational efficiency and performance in various NLP tasks, marking a pivotal moment in the evolution of attention mechanisms from their inception to more sophisticated and efficient variants.

The central idea of the attention mechanism is to shift focus from the task of learning a single vector representation for each sentence. Instead, it adopts a strategy of selectively attending to particular input vectors in the input sequence, guided by assigned attention weights. This strategy enables the model to dynamically allocate its attention resources to the most pertinent segments of the sequence, thereby improving its capacity to process and comprehend the information more efficiently (Brauwers and Frasincar 2021).

One possible explanation for the improvement is that the attention layer created memories associated with the context pattern rather than memories associated with the input itself, relieving pressure on the RNN model structure's weights and causing the model memory to be devoted to remembering the input rather than the context pattern (Hu et al. 2018).

2.2.6 Self-attention

To this point, attention mechanisms in sequence-transformation models have primarily relied on complex RNNs, featuring an encoder and a decoder, the most successful models in language translation yet. However, Vaswani et al. (2017) introduced a simple network architecture known as the Transformer, see Fig. 4, which exclusively utilized attention mechanism, eliminating the need for RNNs. They introduced a novel attention mechanism called self-attention, which is also known as KQV-attention (Key, Query, and Value). This attention mechanism subsequently gained prominence as a central component within the Transformer architecture. The attention mechanism stands out due to its ability to provide Transformers with an extensive long-term memory. In the Transformer model, it becomes possible to focus on all previously generated tokens.

The embedding layer in a Transformer model is the initial stage where input tokens are transformed into dense vectors, capturing semantic information about each token's meaning and context within the text. These embeddings serve as the foundation for subsequent layers to process and understand the relationships between words in the input sequence (Dar et al. 2022).

Self-attention is a mechanism that allows an input sequence to process itself in a way that each position in the sequence can attend to all positions within the same sequence.
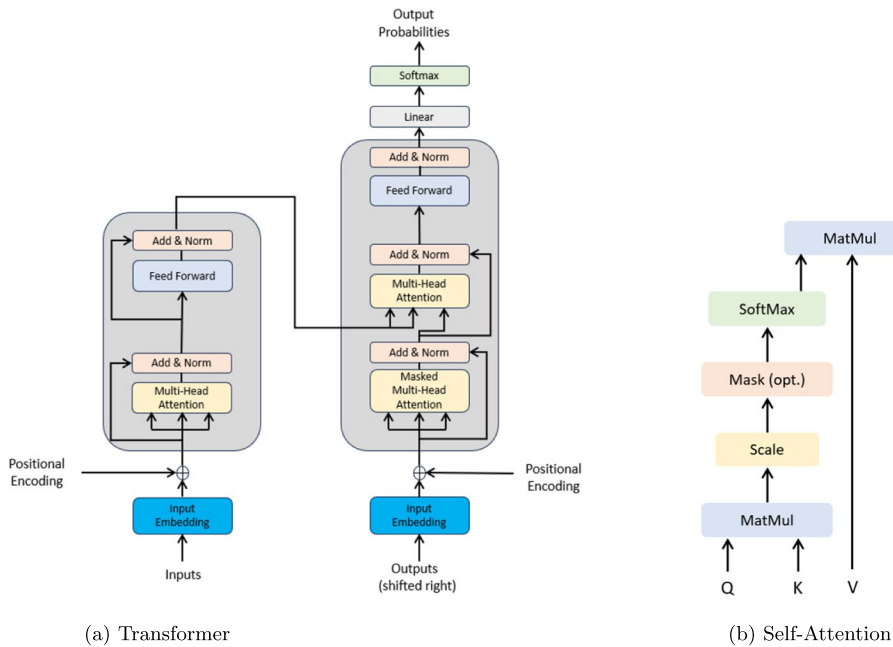
Fig. 4 Transformer architecture and its self-attention mechanism (adapted from Vaswani et al. 2017)

This mechanism is a cornerstone of the Transformer architecture, which has revolutionized NLP and beyond by enabling models to efficiently handle sequences of data with complex dependencies. The purpose of self-attention is to compute a representation of each element in a sequence by considering the entire sequence, thereby capturing the contextual relationships between elements regardless of their positional distance from each other. This ability to capture both local and global dependencies makes self-attention particularly powerful for tasks such as machine translation, text summarization, and sequence prediction, where understanding the context and the relationship between words or elements in a sequence is crucial (Vaswani et al. 2017).

The mathematical formulation of self-attention involves several key steps. First, a set of query vectors $\boldsymbol{Q} = \boldsymbol{X}\boldsymbol{W}^Q$, a set of key vectors $\boldsymbol{K} = \boldsymbol{X}\boldsymbol{W}^K$, and a set of value vectors $\boldsymbol{V} = \boldsymbol{X}\boldsymbol{W}^V$ are calculated through linear transformations of the input sequence, where $\boldsymbol{X}$ is the input matrix representing embeddings of tokens in a sequence, and $\boldsymbol{W}^Q, \boldsymbol{W}^K$, and $\boldsymbol{W}^V$ are weight matrices for queries, keys, and values, respectively. The attention scores are then calculated by taking the dot product of the query vector with all key vectors, followed by scaling the result by the inverse square root of the dimension of the keys (say $\sqrt{d_k}$) to avoid overly large values. These scores are then passed through a softmax function to obtain the attention weights, which represent the importance of each element's contribution to the output. Finally, the output say $\boldsymbol{A}$ is computed as a weighted sum of the value vectors, i.e.,

$$A(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = \text{softmax}(\frac{\boldsymbol{Q}\boldsymbol{K}^\top}{\sqrt{d_k}})\boldsymbol{V}. \tag{12}$$

This process allows the model to dynamically focus on different parts of the input sequence, enabling the extraction of rich contextual information from the sequence.

2.2.7 Multi-head-attention

Multi-head attention is an extension of the self-attention mechanism designed to allow the model to jointly attend the information from different representation subspaces at different positions (Vaswani et al. 2017). Instead of performing a single attention function, it runs the attention mechanism multiple times in parallel. The outputs of these independent attention computations are then concatenated and linearly transformed into the expected dimension. The mathematical formulation of the multi-head attention can be described in the following steps. First, for the i-th self-attention head, find

$$\boldsymbol{Q}_i = \boldsymbol{X}\boldsymbol{W}_i^Q, \quad \boldsymbol{K}_i = \boldsymbol{X}\boldsymbol{W}_i^K, \quad \boldsymbol{V}_i = \boldsymbol{X}\boldsymbol{W}_i^V, \tag{13}$$

and then compute

$$\boldsymbol{A}_i(\boldsymbol{Q}_i, \boldsymbol{K}_i, \boldsymbol{V}_i) = \text{softmax}\left(\frac{\boldsymbol{Q}_i \boldsymbol{K}_i^\top}{\sqrt{d_k}}\right) \boldsymbol{V}_i. \tag{14}$$

The multi-head attention is obtained by concatenating all $\boldsymbol{A}_i(\boldsymbol{Q}_i, \boldsymbol{K}_i, \boldsymbol{V}_i)$.

The multi-head attention mechanism enables the model to capture different types of information from different positions of the input sequence. By processing the sequence through multiple attention "heads", the model can focus on different aspects of the sequence, such as syntactic and semantic features, simultaneously. This capability enhances the model's ability to understand and represent complex data, making multi-head attention a powerful component of Transformer-based architectures (Devlin et al. 2019).

2.3 From transformer to vision transformer

The journey from the inception of the Transformer model to the development of the ViT marks a pivotal advancement in deep learning, showcasing the adaptability of models initially designed for sequence data processing to the realm of image analysis. This transition underscores a significant shift in approach, from conventional image processing techniques to more sophisticated sequence-based methodologies.

Introduced by Vaswani et al. (2017) through the seminal paper "Attention Is All You Need", the Transformer model revolutionized NLP by leveraging self-attention mechanisms. This innovation allowed for the processing of sequences of data without the reliance on recurrent layers, facilitating unprecedented parallelization and significantly reducing training times for large datasets. The Transformer's success in NLP sparked curiosity about its potential applicability across different types of data, including images, setting the stage for a transformative adaptation.

The adaptation of Transformers for image data pivoted on a novel concept: treating images not as traditional 2D arrays of pixels but as sequences of smaller and discrete image patches. This approach, however, faced computational challenges due to the self-attention mechanism's quadratic complexity with respect to input length. The breakthrough came

with the introduction of the ViT by Dosovitskiy et al. (2020), which applied the Transformer architecture directly to images, see Fig. 5. By dividing an image into fixed-size patches and processing these patches as if they were tokens in a text sequence, ViT was able to capture complex relationships between different parts of an image using the Transformer's encoder.

The operational mechanics of ViT begin with the division of an input image into fixed-size patches, each of which is flattened and linearly transformed into a vector, effectively converting the 2D image into a 1D sequence of embeddings. To account for the lack of inherent positional awareness within the Transformer architecture, positional embeddings are added to these patch embeddings, ensuring the model retains spatial information. The sequence of embeddings is then processed through the Transformer encoder, which consists of layers of multi-head self-attention and feed-forward neural networks, allowing the model to dynamically weigh the importance of each patch relative to others for a given task.

For tasks like image classification, the output from the Transformer encoder is passed through a classification head, often utilizing a learnable "class token" appended to the sequence of patch embeddings for this purpose. The model is trained on large datasets using backpropagation and, during inference, processes images through these steps to predict their classes.

The ViT not only demonstrates exceptional performance on image classification tasks, often surpassing CNNs when trained on extensive datasets, but also highlights the Transformer architecture's capacity to capture the global context within images. Despite its advantages, ViT's reliance on substantial computational resources for training and its need for large datasets to achieve optimal performance present challenges. Nonetheless, the development of ViT signifies a significant milestone in the application of sequence processing models to the field of computer vision, opening new avenues for research and practical applications.

The original ViT, designed for static image processing, divides images into patches and interprets these as sequences, leveraging the Transformer's self-attention mechanism to understand complex spatial relationships. Extending this model to action recognition
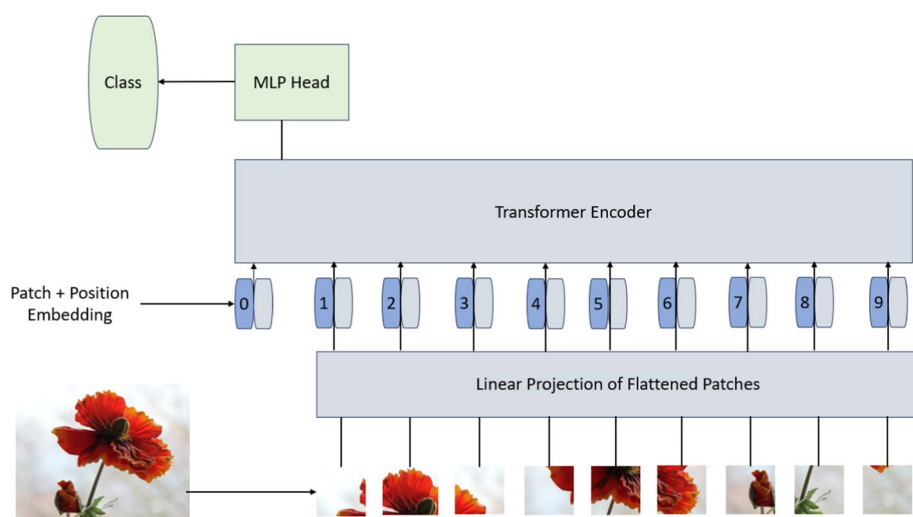

Fig. 5 The ViT architecture (adapted from Dosovitskiy et al. 2020)

involves adapting it to analyze video frames sequentially to capture both spatial and temporal relationships. Several works attempted to adapt ViT in action recognition task using different methods as below.

Temporal dimension integration. The integration of the temporal dimension is a fundamental step in adapting ViT for action recognition. Traditional ViT models process images as a series of patches, treating them essentially as sequences for the self-attention mechanism to analyze spatial relationships. By extending this concept to include the temporal dimension, the models can now treat videos as sequences of frame patches over time. This allows the models to capture the evolution of actions across frames. The work by Bertasius et al. (2021) highlights the potential of incorporating temporal information into Transformers, marking a significant advancement in video analysis capabilities.

Spatio-temporal embeddings. To effectively capture the dynamics of actions within videos, adapted ViT models generate spatiotemporal embeddings. This involves extending the traditional positional embeddings used in ViTs to also include temporal positions, thereby creating embeddings that account for both spatial and temporal information within video sequences. The discussion by Arnab et al. (2021) on the creation of these spatio-temporal embeddings showcases the method's effectiveness in enhancing the model's understanding of action dynamics across both space and time.

Multi-head self-attention across time. The extension of self-attention mechanisms to analyze relationships between patches not just within individual frames but also across different frames is crucial for recognizing actions over time. This approach enables the model to identify relevant features and changes across the video sequences, facilitating a deeper understanding of motion and the progression of actions. The exploration by Bertasius et al. (2021) of this concept demonstrates how Transformers can be effectively adapted to capture the temporal dynamics of actions, a key aspect of video analysis.

2.3.1 Autoencoders in HAR

Autoencoders, particularly sequence-to-sequence architectures, have been instrumental in capturing complex temporal dynamics in HAR tasks. In the context of RNN-based models, autoencoders have been utilized to effectively learn compressed representations of input sequences through the encoder-decoder framework. By mapping input sequences to a latent representation and reconstructing them from that compressed state, autoencoders are capable of learning essential features while discarding irrelevant information. This ability has been particularly valuable for HAR applications involving noise reduction, anomaly detection, and feature extraction. As Transformer-based models gained popularity, autoencoders evolved to leverage attention mechanisms for improved performance. Transformer-based encoders, such as ViTs and their derivatives, apply self-attention mechanisms within the encoder framework to enhance the extraction of spatial and temporal features. Unlike traditional RNN-based autoencoders, which process sequences sequentially, Transformer-based encoders can simultaneously process all input elements, enabling them to capture long-range dependencies more effectively. This parallel processing capability significantly improves training efficiency and enhances the ability to model complex temporal relationships. The transition from RNN-based to ViTs has marked a substantial improvement in HAR performance, particularly in handling large-scale datasets and learning rich hierarchical representations.

3 Literature review

This section briefly recalls the most commonly used deep learning-based HAR approaches.

3.1 CNN-based approaches in HAR

This section recalls the most prominent CNN-based approaches in HAR based on the model type (i.e., the two-stream CNN, 3D CNN, and RNNs with CNNs), organized chronologically.

Deep learning was still in its early stages in 2012, and CNNs or RNNs had not yet gained significant popularity in the field of HAR. The focus was primarily on traditional machine learning approaches, such as support vector machines (Cortes and Vapnik 1995), and hand-crafted features, such as histogram of oriented gradients (Dalal and Triggs 2005) and histogram of optical flow (Barron et al. 1994). A few studies did, nevertheless, start looking into neural networks for action recognition.

In 2014, the use of CNNs in action recognition was at a pivotal stage, marking a shift from hand-crafted feature-based methods to deep learning approaches. The key points of the use of CNNs in action recognition at that period of time are the following. (I) *Emergence of deep learning:* deep learning, particularly CNNs, had started to dominate image classification tasks, thanks to their ability to learn feature representations directly from raw pixel data. This success in static images paved the way for applying CNNs to video data for action recognition. (II) *Challenges in video data:* unlike 2D images, videos incorporate a third dimension which represents the temporal patterns, making action recognition more complex. CNNs had to be adapted to not only recognize spatial patterns but also capture motion information over time dimension. (III) *Datasets and benchmarks:* the adoption of large-scale video datasets like UCF-101 (Soomro et al. 2012) and HMDB-51 (Kuehne et al. 2011) became more common. These datasets provided diverse sets of actions and were large enough to train deep networks. The performance on these benchmarks has been becoming a key measure of progress for action recognition models. (IV) *Transfer learning:* due to the computational expense of training CNNs from scratch and the relatively smaller size of video datasets compared to image datasets, transfer learning became a popular strategy. Networks pre-trained on large image datasets like ImageNet (Deng et al. 2009) were fine-tuned on video frames for action recognition tasks. (V) *Computational constraints:* despite the promise of CNNs, computational constraints were a significant challenge. Training deep networks required significant GPU power, and processing video data with CNNs was resource-intensive. This limited the complexity of the models that could be trained and the size of the datasets that could be used.

3.1.1 Two-stream CNNs

Simonyan and Zisserman (2014a) presented an innovative approach to recognize actions in video sequences by using a two-stream CNN architecture. This approach divides the task into two distinct problems: recognizing spatial features from single frames and capturing temporal features across frames. The spatial stream CNN processes static visual information, while the temporal stream CNN handles motion by analyzing optical flow. The model was tested on benchmark datasets like UCF-101 and HMDB-51, where it achieved state-of-the-art results, showcasing the effectiveness of this two-stream method. The novelty of

this work lies in the separation of motion and appearance features, which allows for more specialized networks that can better capture the complexities of video-based action recognition. The success of this model has made a significant impact on the field, influencing many future research directions in video understanding. Consequently, numerous methods have been proposed to enhance the the two-stream model (Wang et al. 2015; Feichtenhofer et al. 2016; Wang et al. 2016; Peng et al. 2018; Wang et al. 2017).

In 2016, building on the the two-stream CNN, Feichtenhofer et al. (2016) focused on improving the two-stream CNN by exploring various fusion strategies for combining spatial and temporal streams, resulting in better performance on the UCF-101 and HMDB-51 datasets. By enhancing fusion techniques, this work addressed the limitations of the initial two-stream model, leading to more effective integration of spatial and temporal information. Wang et al. (2016) introduced temporal segment networks (TSN). This work aimed to capture long-range temporal structures for action recognition, achieving significant improvements on the UCF-101 and HMDB-51 datasets by dividing videos into segments for comprehensive analysis. The introduction of TSN extended the temporal analysis capabilities of the two-stream CNN, enabling the capture of long-range dependencies.

In 2017, derived from the two-stream CNN, Cosmin Duta et al. (2017) proposed a three-stream method by using spatio-temporal vectors, with locally max-pooled features to enhance performance. Tested on the UCF-101 and HMDB-51 datasets, the approach demonstrated improved recognition accuracy by efficiently capturing spatio-temporal dynamics. In 2018, the efficient convolutional network for online video understanding (ECO) was introduced by Zolfaghari et al. (2018), combining the two-stream CNN approach with lightweight 3D CNNs, and focusing on efficiency and real-time processing, with high efficiency and competitive accuracy demonstrated on the Kinetics and UCF-101 datasets.

Feichtenhofer et al. (2019) introduced the SlowFast network which processes video data at varying frame rates to capture both spatial semantics and motion dynamics, achieving state-of-the-art results on the Kinetics-400 and Charades datasets. By introducing different temporal resolutions, this work innovated on the two-stream concept, capturing fine and coarse temporal details. Wang et al. (2018a) expanded on their previous work with TSN, developing a multi-stream approach that incorporated RGB, optical flow, and warped optical flow streams to model long-range temporal structures more effectively. This approach achieved state-of-the-art results by capturing both spatial and temporal information across various time scales. In 2021, temporal difference networks (TDN) were introduced by Wang et al. (2021a), leveraging the multi-stream CNN with a focus on capturing motion dynamics efficiently. Using the UCF-101 and HMDB-51 datasets, TDN achieved notable improvements by effectively modeling temporal differences. By emphasizing temporal differences, this work advanced the ability of the two-stream CNN to capture motion dynamics more effectively. Hussain et al. (2023) presented a novel approach for HAR in low-light and complex environments. It integrates a lightweight CNN for enhancing low-light frames, YOLOv7-Tiny for human detection, and a dual-stream network combining CNN and transformer features. These features are processed using a custom Optimized Parallel Sequential Temporal Network (OPSTN) with squeeze-and-excitation attention for efficient recognition. The model achieved superior performance on HMDB51, UCF50, and YouTube Action datasets, outperforming many existing state-of-the-art methods.

Table 2 presents the works discussed in this section that utilized two or more stream CNNs approaches.

3.1.2 3D CNN-based approaches

The foundational work conducted by Ji et al. (2012) introduced 3D CNNs for HAR, demonstrating their effectiveness in capturing spatio-temporal features on the KTH and UCF-101 datasets and outperforming traditional 2D CNNs. The work paved the way for further research on enhancing 3D convolutional models. Tran et al. (2015) introduced C3D, a generic 3D CNN for spatio-temporal feature learning, achieving state-of-the-art performance on the Sports-1 M and UCF-101 datasets and highlighting the scalability and effectiveness of 3D convolutions. Building on the work by Ji et al. (2012), C3D demonstrated

Table 2 Two-stream CNN-based approaches in HAR

Paper	Model	Dataset	Novelty
Simonyan and Zisserman (2014a)	Two-stream CNN	UCF-101, HMDB-51	Introduced the two-stream architecture separating spatial and temporal streams for effective action recognition
Feichtenhofer et al. (2016)	Two-stream CNN	UCF-101, HMDB-51	Explored various fusion strategies to combine spatial and temporal streams, and improved performance
Wang et al. (2016)	Two-stream CNN + TSN	UCF-101, HMDB-51	Introduced TSN to capture long-range temporal structures by dividing videos into segments
Cosmin Duta et al. (2017)	Three-Stream CNN	UCF-101, HMDB-51	Proposed a three-stream method using spatio-temporal vectors with locally max-pooled features for enhanced performance
Zolfaghari et al. (2018)	Two-stream CNN + 3D CNN	Kinetics, UCF-101	Combined the two-stream CNN with lightweight 3D CNNs for efficient real-time processing
Feichtenhofer et al. (2019)	Two-stream CNN + SlowFast	Kinetics-400, Charades	Introduced SlowFast networks processing video data at varying frame rates to capture both spatial and motion dynamics
Wang et al. (2018a)	CNN-RNN, (Multi-stream TSN)	UCF101, HMDB51	Expanded on TSN by developing a multi-stream approach that incorporated RGB, optical flow, and warped optical flow streams to model long-range temporal structures more effectively
Wang et al. (2021a)	Multi-stream CNN + TDN	Something-Something V1 and V2	Introduced TDN focusing on capturing motion dynamics efficiently

the potential of 3D CNNs across diverse datasets, influencing subsequent research in 3D CNNs. Varol et al. (2017) introduced long-term temporal convolutions to capture extended motion patterns. This work improved the accuracy on the UCF-101 and HMDB-51 datasets and emphasized the importance of long-term motion information. Moreover, this study extended the temporal scope of 3D CNNs, highlighting the need for capturing long-term motion for accurate action recognition. In the same year, Qiu et al. (2017) proposed pseudo-3D residual networks (P3D), which combined 2D and 3D convolutions to balance the accuracy and computational complexity. This work achieved competitive performance on the Kinetics and UCF-101 datasets. Moreover, P3D networks offered a more efficient approach by blending 2D and 3D convolutions, further refining the capabilities of 3D CNNs. Additionally, Carreira and Zisserman (2017) introduced I3D by inflating 2D convolutions to 3D, achieving significant improvements on the Kinetics dataset by leveraging ImageNet pre-training, thereby setting new performance benchmarks. I3D bridged the gap between 2D and 3D CNNs, demonstrating the benefits of transfer learning in 3D convolutional models.

Hara et al. (2018) evaluated the scalability of 3D CNNs with increased data and model sizes, demonstrating that deeper 3D CNNs can achieve better performance on the Kinetics and UCF-101 datasets, paralleling the success of 2D CNNs on ImageNet. This study emphasized the need for larger datasets and deeper models in 3D convolutional research, highlighting the potential of 3D CNNs to retrace the historical success of 2D CNNs. Building on these insights, Diba et al. (2017) introduced a new temporal 3D ConvNet architecture with enhanced transfer learning capabilities, demonstrating superior performance on the UCF-101 and HMDB-51 datasets through architectural innovations and effective transfer learning. This work underscored the importance of architectural innovation and transfer learning, pushing the boundaries of 3D CNN performance and further advancing the field of action recognition. Tran et al. (2018) further contributed by conducting a comprehensive analysis of spatio-temporal convolutions, highlighting the benefits of factorizing 3D convolutions into separate spatial and temporal components, achieving state-of-the-art results on the Kinetics and UCF-101 datasets. This dissection provided insights that informed subsequent model designs and optimizations. In the same year, Xie et al. (2018) explored the trade-offs between speed and accuracy in spatio-temporal feature learning, proposing efficient 3D CNN variants that balance computational cost and recognition performance on the Kinetics and UCF-101 datasets. Their work highlighted the practical considerations of deploying 3D CNNs, emphasizing the need to balance speed and accuracy, thereby refining the approach to spatio-temporal feature learning. Additionally, Wang et al. (2018b) introduced non-local neural networks to capture long-range dependencies, demonstrating that non-local operations significantly improve the modeling of complex temporal relationships and enhance action recognition performance on the Kinetics and Something-Something datasets. By integrating non-local operations, this study advanced the ability of 3D CNNs to capture complex temporal patterns, further pushing the boundaries of spatio-temporal modeling.

Feichtenhofer et al. (2019) introduced SlowFast Networks, a novel approach that processes video at different frame rates to capture both slow and fast motion dynamics, and achieved state-of-the-art results on the Kinetics-400 and Charades datasets. This innovation highlighted the importance of capturing varied motion dynamics for improved video recognition. In the same year, Tran et al. (2019) presented channel-separated convolutional networks (CSN), which reduced computational complexity by separating convolu-

tions by channel, demonstrating efficiency without sacrificing accuracy on the Kinetics and Sports-1 M datasets. This approach contributed to the development of more computationally feasible models. Concurrently, Ghadiyaram et al. (2019) leveraged large-scale weakly-supervised pre-training on video data, significantly boosting performance on the IG-65 M and Kinetics datasets and underscoring the potential of massive datasets in enhancing 3D CNN capabilities. Additionally, Kopuklu et al. (2019) proposed resource-efficient 3D CNNs using depthwise separable convolutions and achieved competitive accuracy with significantly reduced computational requirements on the Kinetics-400 and UCF-101 datasets. This work emphasized the importance of optimizing 3D CNNs for computational efficiency, further advancing the field of action recognition.

Feichtenhofer (2020) proposed X3D, a family of efficient video models by expanding architectures along multiple axes. It achieved state-of-the-art performance with reduced model complexity on the Kinetics-400 and Charades datasets. X3D highlighted the significance of model efficiency in balancing performance and computational demands. In the same year, Li et al. (2020) introduced an efficient 3D CNN with a temporal attention mechanism and achieved high accuracy with efficient computation by focusing on salient temporal features on the Kinetics-400 and UCF-101 datasets. This work demonstrated the potential of selectively focusing on important temporal features to enhance the efficiency and accuracy of 3D CNNs, further advancing the field of action recognition.

Table 3 presents the works discussed in this section that utilized 3D CNN approaches.

3.1.3 CNN-RNN-based approaches

The integration of CNNs and RNNs for HAR was significantly advanced by the work of Donahue et al. (2015), who introduced long-term recurrent convolutional networks (LRCN). This approach effectively combined the spatial feature extraction capabilities of CNNs with the temporal dynamics modeling of LSTMs, demonstrating substantial improvements in action recognition tasks on datasets like UCF-101 and HMDB-51. Building on this foundation, Yue-Hei Ng et al. (2015) extended the application of deep networks to video classification by integrating deep CNNs with LSTMs to handle longer video sequences. Their method, tested on the Sports-1 M and UCF-101 datasets, highlighted the importance of capturing extended temporal dependencies for improved performance in complex video classification tasks. Further pushing the boundaries, Srivastava et al. (2015) explored unsupervised learning of video representations using LSTMs. By leveraging LSTMs to learn spatio-temporal features without labeled data, their approach demonstrated effective video representation learning on the UCF-101 dataset, showcasing the versatility and potential of CNN-RNN architectures in both supervised and unsupervised learning scenarios for HAR.

The development of CNN-RNN architectures for HAR saw significant advancements in 2016. Wu et al. (2015) proposed a hybrid deep learning framework that modeled spatial-temporal clues by combining CNNs for spatial feature extraction with RNNs for temporal sequence modeling. Their approach, tested on the UCF-101 and HMDB-51 datasets, demonstrated substantial improvements in video classification accuracy. Additionally, Li et al. (2016) expanded the application of CNN-RNN architectures to real-time scenarios with their approach for online human action detection using joint classification-regression RNNs. Combining CNNs for spatial features and RNNs for temporal dynamics, their method, tested

Table 3 3D CNN-based approaches in HAR

Paper	Model	Dataset	Novelty
Ji et al. (2012)	3D CNN	UCF-101, HMDB-51	Introduced 3D CNNs for HAR, effectively capturing spatio-temporal features and outperforming 2D CNNs
Tran et al. (2015)	3D CNN	Sports-1 M, UCF-101	Introduced C3D, a generic 3D CNN for spatio-temporal feature learning, and achieved state-of-the-art performance
Varol et al. (2017)	3D CNN	UCF-101, HMDB-51	Introduced long-term temporal convolutions to capture extended motion patterns, and improved accuracy
Qiu et al. (2017)	3D CNN	Kinetics, UCF-101	Proposed P3D networks combining 2D and 3D convolutions, balancing accuracy and computational complexity
Carreira and Zisserman (2017)	3D CNN	Kinetics	Introduced I3D by inflating 2D convolutions to 3D, leveraging ImageNet pre-training for significant improvements
Hara et al. (2018)	3D CNN	Kinetics, UCF-101	Evaluated the scalability of 3D CNNs with increased data and model sizes, and showed parallels to 2D CNN success
Diba et al. (2017)	3D CNN	UCF-101, HMDB-51	Introduced a new temporal 3D ConvNet architecture with enhanced transfer learning capabilities
Tran et al. (2018)	3D CNN	Kinetics, UCF-101	Conducted a comprehensive analysis of spatio-temporal convolutions, and highlighted the benefits of factorizing 3D convolutions
Xie et al. (2018)	3D CNN	Kinetics, UCF-101	Explored speed-accuracy trade-offs in spatio-temporal feature learning, and proposed efficient 3D CNN variants
Wang et al. (2018b)	3D CNN	Kinetics, Something-Something	Introduced non-local operations to capture long-range dependencies, and improved modeling of complex temporal relationships
Feichtenhofer et al. (2019)	3D CNN	Kinetics-400, Charades	Proposed SlowFast networks to process video at different frame rates, capturing both slow and fast motion dynamics
Tran et al. (2019)	3D CNN	Kinetics, Sports-1 M	Introduced CSN to reduce computational complexity without sacrificing accuracy
Ghadiyaram et al. (2019)	3D CNN	IG-65 M, Kinetics	Leveraged large-scale weakly-supervised pre-training on video data, and significantly boosted performance
Kopuklu et al. (2019)	3D CNN	Kinetics-400, UCF-101	Proposed resource-efficient 3D CNNs using depthwise separable convolutions, and achieved competitive accuracy with reduced computational requirements
Feichtenhofer (2020)	3D CNN	Kinetics-400, Charades	Proposed X3D, a family of efficient video models by expanding architectures along multiple axes
Li et al. (2020)	3D CNN	Kinetics-400, UCF-101	Introduced a temporal attention mechanism to enhance efficiency and accuracy in 3D CNNs

on the J-HMDB and UCF-101 datasets, achieved notable improvements in accuracy and efficiency, showcasing the practicality of CNN-RNN models in real-time action detection.

Building on these advancements, 2017 and 2018 witnessed further refinements and innovations in CNN-RNN architectures for HAR. Li et al. (2018) introduced VideoLSTM, integrating convolutions, attention mechanisms and optical flow within a recurrent framework, and demonstrating improved performance on the UCF101 and HMDB51 datasets. Carreira and Zisserman (2017) made a significant contribution with the two-stream Inflated 3D ConvNet (I3D), which inflated 2D CNN architectures into 3D and combined them with RNNs for temporal modeling. The model was evaluated on the Kinetics dataset, as well as

UCF101 and HMDB51. Ullah et al. (2017) proposed a novel architecture combining CNNs with bi-directional LSTMs, effectively utilizing both spatial and temporal information from video sequences and showing superior performance on the UCF-101 and HMDB-51 datasets. In 2020, in the realm of human activity recognition using sensor data, Xia et al. (2020) proposed an LSTM-CNN architecture that effectively captured both temporal dependencies and local feature patterns, showing improved accuracy on the WISDM, UCI HAR, and OPPORTUNITY datasets. Similarly, Mutegeki and Han (2020) developed a CNN-LSTM approach for smartphone sensor-based activity recognition, demonstrating high accuracy on the UCI HAR dataset and further validating the effectiveness of combining CNNs and RNNs for processing time-series data in activity recognition tasks.

Recent advancements in HAR have leveraged sophisticated CNN-RNN architectures to enhance performance and reduce computational complexity. Muhammad et al. (2021) introduced an attention-based LSTM network combined with dilated CNN features, and significantly improved the recognition accuracy on the UCF-101 and HMDB-51 datasets by capturing essential spatial features through dilated convolutions and temporal patterns with attention mechanisms. Building on this, Malik et al. (2023) focused on multiview HAR; utilizing a CNN-LSTM architecture to cascade pose features, they achieved high accuracy (94.4% on the MCAD dataset and 91.67% on the IXMAS dataset) while reducing the computational load by targeting pose data rather than entire images.

In 2024, Hussain et al. (2024c) proposed a Human-Centric Attention with Deep Multi-scale Feature Fusion Framework aimed at improving HAR within the Internet of Medical Things (IoMT). Their approach involves a combination of MobileNetV3 and a redesigned bidirectional LSTM integrated with Sequential Multihead Attention to enhance long-range temporal dependencies. Their model achieved impressive accuracy on both healthcare and general HAR datasets, outperforming existing methods. Another work done by Hussain et al. (2024a) developed an AI-driven behavior biometrics framework utilizing a Dynamic Attention Fusion Unit (DAFU) and Temporal-Spatial Fusion (TSF) network. Their approach integrated Echo-ConvLSTM to enhance accuracy and robustness in recognizing complex activities. This model demonstrated superior performance across multiple public HAR datasets.

Table 4 presents the works discussed in this section that utilized CNN-RNN approaches.

3.2 ViT-based approaches in HAR

In 2020, the ViT was conceptualized and introduced in the academic domain through the paper authored by Dosovitskiy et al. (2020). The ViT marked a paradigm shift in still image recognition methodologies, applying the Transformer model, predominantly known for its success in NLP, to the realm of computer vision. The application of ViTs in action recognition, a more specific and complex task within the field of computer vision, followed the initial introduction of ViT. Specifically, in 2021 and beyond, subsequent research and publications have explored and expanded the use of ViTs for action recognition tasks, demonstrating their efficacy in capturing spatial-temporal features within video data. They employ attention mechanisms to minimize redundant information and to model interactions over long distances in both space and time (Koot et al. 2021). The adaptation of ViT to action recognition signifies the model's versatility and its potential for broader applications in computer vision beyond static image analysis.

Table 4 CNN-RNN-based approaches in HAR

Paper	Model	Dataset	Novelty
Donahue et al. (2015)	CNN-RNN, (LRCN)	UCF-101, HMDB-51	Combined CNNs for spatial feature extraction with LSTMs for temporal dynamics
Yue-Hei Ng et al. (2015)	CNN-RNN	Sports-1 M, UCF-101	Integrated deep CNNs with LSTMs to handle longer video sequences, capturing extended temporal dependencies
Srivastava et al. (2015)	CNN-RNN, (Unsupervised LSTM)	UCF-101	Explored unsupervised learning of video representations using LSTMs, leveraging spatiotemporal features
Wu et al. (2015)	CNN-RNN	UCF-101, HMDB-51	Modeled spatial-temporal clues by combining CNNs for spatial features with RNNs for temporal sequence modeling
Li et al. (2016)	CNN-RNN	J-HMDB, UCF-101	Applied CNN-RNN architectures to real-time scenarios for online human action detection
Li et al. (2018)	CNN-RNN (VideoLSTM)	UCF-101, HMDB-51	Integrated convolutions, attention mechanisms, and optical flow within a recurrent framework
Carreira and Zisserman (2017)	3D CNN-RNN	Kinetics, UCF101, HMDB51	Inflated 2D CNN architectures into 3D, and combined them with RNNs for temporal modeling
Ullah et al. (2017)	CNN-RNN, (CNN-BiLSTM)	UCF101, HMDB51	Combined CNNs with bi-directional LSTMs to utilize both spatial and temporal information
Xia et al. (2020)	CNN-RNN	WISDM, UCI, OPPORTUNITY	Captured both temporal dependencies and local feature patterns for human activity recognition using sensor data
Mutegeki and Han (2020)	CNN-RNN	UCI	Developed a CNN-LSTM approach for smartphone sensor-based activity recognition, and demonstrated high accuracy
Muhammad et al. (2021)	CNN-RNN, (CNN-Attention-LSTM)	UCF-101, HMDB-51	Improved recognition accuracy with attention-based LSTM network combined with dilated CNN features
Malik et al. (2023)	CNN-RNN	MCAD, IXMAS	Achieved high accuracy in multiview HAR by cascading pose features using a CNN-LSTM architecture

Recent advancements in action recognition have seen a significant shift towards ViT, highlighting their efficacy in video understanding tasks. Arnab et al. (2021) introduced ViViT, extending the vision Transformer architecture to handle video sequences. They demonstrated its potential on datasets like Kinetics-400 and Something-Something-V2, marking a substantial improvement in video action recognition capabilities. Building on this, Bertasius et al. (2021) proposed a space-time Transformer that models temporal information innovatively, and achieved competitive results on similar datasets. The efficiency of multi-scale ViTs was further illustrated by Fan et al. (2021), who showed that such architectures could effectively capture fine-grained video details and enhance classification performance on comprehensive video datasets. Moreover, Liu et al. (2022) presented the Swin Transformer, utilizing a shifted window mechanism to model long-range dependencies more efficiently, and leading to significant improvements in action recognition accuracy. Together,

these works underscore the transformative impact of ViTs in advancing the field of HAR. Additionally, Wang et al. (2021b) introduced ActionCLIP, leveraging the CLIP model for enhanced video action recognition on multiple standard video datasets, including Kinetics-400 and HMDB-51. This novel approach integrated visual and linguistic representations.

Chen and Ho (2022) introduced Mm-ViT, a multi-modal video Transformer designed for compressed video action recognition, and demonstrated high performance by leveraging multi-modal inputs on compressed video datasets such as HACS and UCF101. Sharir et al. (2021) explored the extension of ViT to video data, showing its potential in capturing temporal dynamics effectively across several standard video datasets, including Kinetics-400 and HMDB-51. Furthermore, Xing et al. (2023) developed SVFormer, a semi-supervised video Transformer that leverages both labelled and unlabeled data to bridge the gap between supervised and unsupervised learning, and achieved significant improvements in action recognition tasks on various standard HAR datasets such as Kinetics-400 and UCF101. Together, these works underscore the transformative impact of ViTs in advancing the field of HAR.

In 2024, Hussain et al. (2024e) proposed a Hybrid Transformer Framework aimed at efficient HAR on consumer electronics devices. Their method involves using a MobileNetV3 model for extracting salient contextual features from each video frame. These features are then passed to a Sequential Residual Transformer Network (SRTN) designed to effectively learn long-range temporal dependencies. The SRTN's temporal multi-head self-attention module and residual strategy enhance the extraction of discriminative features while discarding irrelevant information. This approach achieved high accuracy on challenging HAR datasets, e.g. 96.64% on UCF101, 76.14% on HMDB51, and 97.31% on UCF50.

While our study primarily focuses on the ViT model due to its significant impact on HAR, we acknowledge that several other Transformer architectures have been developed and applied to HAR and related tasks. Notable examples include Swin Transformers (Chen and Mo 2023) (which introduce hierarchical feature representations using shifted windows to enhance computational efficiency), Timesformer (Bertasius et al. 2021) (which factorizes spatio-temporal attention for video understanding), and Motion Transformers (Shi et al. 2022) (which explicitly model motion dynamics). Furthermore, hybrid transformer models, such as CNN-Transformer architectures (Reda et al. 2023; Leong et al. 2022), have been explored to balance local feature extraction with global context modelling. In contrast, our model proposes a hybrid CNN-ViT architecture, focusing on combining CNNs for spatial features with ViTs for spatiotemporal reasoning. Leong et al. (2022) integrated a 3D CNN with a Transformer Encoder and optionally a cross-modal video-text attention module to enhance fine-grained action recognition using textual class descriptions. The model ConViViT (Reda et al. 2023) used a CNN to preprocess RGB videos into a 128-channel representation, followed by a Transformer with factorized self-attention to model spatial and temporal features efficiently.

HAR plays a pivotal role in advancing various domains beyond conventional activity recognition. For instance, the work in Hussain et al. (2024d) introduced a novel framework utilising attention-inspired sequential temporal convolution networks (STCN) to enhance HAR in industrial surveillance systems. This framework demonstrated the application of HAR in monitoring worker interactions with industrial devices, thereby contributing to improved safety, operational efficiency, and anomaly detection within industrial settings. Furthermore, in the domain of Sports Action Recognition (SAR), the study in Hussain et al.

(2024f) presented the Cricket Excited Actions (CEA) dataset, providing a comprehensive benchmark for analyzing complex multi-person activities in cricket matches. By enabling accurate recognition of cricket-specific actions, this dataset supports advancements in both sports analytics and commercial entertainment, bridging the gap between academia and real-world applications. These contributions highlight the broader applicability of HAR techniques in enhancing efficiency and safety across diverse fields.

More notable works incorporated include a novel slow-fast tubelet processing framework for efficient action recognition in infrared-based scenarios, demonstrating superior performance on benchmark datasets NTURGB-D 120 and InfAR (Munsif et al. 2024b). Additionally, a contextual-motion coalescence network (CMCNet) was proposed for robust action representation in dark environments by synergistically integrating contextual visual features and temporal optical flow learning modules (Munsif et al. 2024a). Hussain et al. (2022) proposed a CNN-free approach combining ViT (ViT-Base-16) for spatial feature extraction with a multilayer LSTM network to capture long-range temporal dependencies. This method addresses challenges like occlusion, low resolution, and varying viewpoints in surveillance environments. Evaluated on UCF50 and HMDB51 datasets, the model achieved strong performance with 96.14% accuracy on UCF50 and 73.71% on HMDB51, surpassing many state-of-the-art methods.

Table 5 presents the works discussed in this section that utilized ViTs.

3.3 CNN-ViT hybrid architectures

The integration of ViTs with CNNs has significantly advanced HAR tasks. Yin and Yin (2024) proposed a two-stream hybrid CNN-Transformer network (THCT-Net), which demonstrated enhanced generalization ability and convergence speed on the NTU RGB+D dataset by combining CNNs for low-level context sensitivity and Transformers for capturing global information. Following this, Shan et al. (2021) applied a similar hybrid model to driver action recognition, leveraging multi-view data to achieve high accuracy through the integration of CNNs for spatial feature extraction and Transformers for temporal dependencies. Mazzeo et al. (2022) extended this approach by integrating 3D CNNs with Transformers for late temporal modeling, and achieved substantial improvements in action recognition accuracy on the HMDB-51 and UCF101 datasets. Moreover, Chen and Mo (2023) proposed Swin-Fusion, which combines Swin Transformers with CNN-based feature fusion to achieve state-of-the-art performance on datasets like Kinetics-400 and Something-Something-V2, demonstrating the robustness and superior performance of hybrid models in HAR tasks.

Djenouri and Belbachir (2023) proposed a hybrid visual Transformer model that integrates CNNs and Transformers for efficient and accurate human activity recognition. They demonstrated its capability on datasets like Kinetics-400 and UCF101, and showed that the hybrid approach leverages the local feature extraction of CNNs with the global context modeling of Transformers. Following this, Surek et al. (2023) provided a comprehensive review of deep learning approaches for video-based human activity recognition, emphasizing the potential of hybrid models. This review underscored the effectiveness of such hybrid models in capturing both spatial and temporal features from video data, and evaluated on various human activity datasets including NTU RGB+D and UTD-MHAD. Ahmadabadi et al. (2023) explored the use of knowledge distillation techniques to enhance the perfor-

Table 5 ViT-based approaches in HARs

Paper	Model	Dataset	Novelty
Arnab et al. (2021)	ViViT	Kinetics-400, Something-Something-V2	Extended ViT to video sequences
Bertasius et al. (2021)	Space-Time Transformer	Kinetics-400	Innovative temporal information modeling
Fan et al. (2021)	Multiscale ViT	Kinetics-400, Something-Something-V2	Efficient capture of fine-grained video details
Liu et al. (2022)	Swin Transformer	Kinetics-400, Something-Something-V2	Shifted window mechanism for long-range dependency modeling
Wang et al. (2021b)	ActionCLIP	Kinetics-400, HMDB-51	Leveraged CLIP for enhanced video action recognition
Chen and Ho (2022)	Mm-ViT	HACS, UCF101	Multi-modal inputs for compressed video action recognition
Sharir et al. (2021)	ViT	Kinetics-400, HMDB-51	Applied ViT to video data
Xing et al. (2023)	SVFormer	Kinetics-400, UCF101	Semi-supervised learning for action recognition

mance of hybrid CNN-Transformer models. Their approach was validated on datasets such as HMDB-51 and Kinetics-400, showing significant improvements in HAR by effectively transferring knowledge from complex models to more efficient ones. Together, these works highlight the evolving landscape of hybrid models in human activity recognition, showcasing their robustness and efficiency in handling complex video data. Hussain et al. (2024b) proposed a dual-stream framework for robust HAR in surveillance videos. It used a shot segmentation module trained on a custom lowlight dataset to filter and enhance frames. Spatial and motion features are extracted using ViT-B16 and FlowNet2, then processed with a Parallel BiLSTM and dual stream multi-head attention. The model achieved high accuracy

on benchmark datasets: 78.63% (HMDB51), 96.02% (UCF101), and 98.88% (YouTube Actions) (Table 6).

Table 3 presents the works discussed in this section that utilized CNN-ViT approaches.

3.4 Discussion

In the field of HAR, the choice of models – whether CNN-based, ViT-based, or a hybrid of CNN and ViT – significantly influences the outcome and efficiency of the task. CNN-based models are particularly adept at extracting local features due to their convolutional nature (LeCun et al. 2015), making them highly effective in pattern recognition within images and videos. Their computational efficiency is a boon for real-time applications (Howard et al. 2017), and their robustness to input variations is notable (Simonyan and Zisserman 2014b). However, CNNs often struggle with global contextual understanding (Szegedy et al. 2015) and are prone to overfitting. Moreover, their ability to model long-range temporal dependencies (Karpathy et al. 2014), which is crucial in action recognition, is somewhat limited.

ViT-based models, in contrast, excel in capturing global dependencies (Carion et al. 2020; Dosovitskiy et al. 2020), thanks to their self-attention mechanism. This attribute makes them particularly suited for understanding complex actions that require a broader view beyond local features. ViTs are scalable with data, benefiting significantly from larger datasets, and are flexible in processing inputs of various sizes (Touvron et al. 2021). The adaptability in processing various input sizes is a byproduct of the patch-based approach and the global receptive field of the ViTs. However, these models are computationally more intensive and require substantial training data to achieve optimal performance (Khan et al. 2022). Unlike CNNs, ViTs are not as efficient in extracting detailed local features, which can be a critical drawback in certain action recognition scenarios. For more clarification, a detailed explanation of the ViViT model including its capabilities and limitations was introduced by Arnab et al. (2021). ViViT is a pure Transformer-based model designed specifically for video recognition tasks. It applies a spatio-temporal attention mechanism to process video frames as a sequence of patches, enabling it to capture both spatial and temporal information effectively. While ViViT has demonstrated impressive performance on datasets like Kinetics-400 and Something-Something-V2, it faces challenges related to computational complexity and the requirement for large-scale training data.

Hybrid models that combine CNNs and ViTs aim to harness the strengths of both architectures. They offer the local feature extraction capabilities of CNNs along with the global context awareness of ViTs, potentially providing a more balanced approach to action recognition. These models can be more efficient and versatile, adapting well to a range of tasks. However, this combination brings its own challenges, including increased architectural complexity, higher resource demands, and the need for careful tuning to balance the contributions of both CNN and ViT components. The choice among these models depends on the specific requirements of the action recognition task, such as the available computational resources, the nature and size of the dataset, and the types of actions that need to be recognized.

For a summary of the advantages and disadvantages of these three architectural variations, see Table 7.

Table 6 CNN-ViT hybrid approaches in HARs

Paper	Model	Datase	Novelty
Yin and Yin (2024)	The two-stream hybrid CNN-Transformer network (THCT-Net)	NTU RGB+D	Combined CNNs and Transformers for improved generalization and convergence speed
Shan et al. (2021)	Multi-view vision Transformer	Custom driver action datasets	Leveraged multi-view data for spatial and temporal feature integration
Mazzeo et al. (2022)	3D CNN-Transformer	HMDB-51, UCF101	Integrated 3D CNNs with Transformers for late temporal modeling
Chen and Mo (2023)	Swin-Fusion	Kinetics-400, Something-Something-V2	Combined Swin Transformers with CNN-based feature fusion for state-of-the-art performance
Dj-enouri and Bel-bachir (2023)	Hybrid visual Transformer	Kinetics-400, UCF101	Efficient and accurate human activity recognition leveraging strengths of CNNs and Transformers
Surek et al. (2023)	Various deep learning models including hybrid models	NTU RGB+D, UTD-MHAD	Comprehensive review highlighting the potential of hybrid models
Ah-mad-abadi et al. (2023)	Hybrid CNN-Transformer	HMDB-51, Kinetics-400	Knowledge distillation from CNN-Transformer models for enhanced performance

4 Proposed CNN-ViT hybrid architecture

In this section, we present our proposed CNN-ViT architecture for HAR, leveraging the benefits of both approaches described in previous sections, see Fig. 6. The architecture incorporates a TimeDistributed layer with a CNN backbone, followed by a ViT model to classify actions in video sequences.

Spatial component. Let $\mathcal{X}$ be a collection of N frames, i.e., $\mathcal{X} = \{X_i\}_{i=1}^{N}$. The CNN backbone (i.e. MobileNet in (Howard et al. 2017)) in the TimeDistributed layer (see Fig. 6) processes the indifivual frames X_i and outputs the spatial features vector $v_i = p_\theta(X_i) \in \mathbb{R}^L$, where p_θ is the CNN model (e.g. MobileNet or VGG16) with parameters in θ wrapped by the TimeDistributed layer.

Temporal component. In the proposed hybrid CNN-ViT model, iT is engineered to process the sequence of the N spatial features vectors, i.e., $\{v_i\}_{i=1}^{N}$, where each v_i represents a distinct frame of the input video clip, see Fig. 6. Afterwards, the ViT block outputs a final representation z, which is then fed into the softmax layer to classify the action in the video. In detail, the Transformer encoder is designed to process a sequence of vectors, each representing one frame, and aggregate information into a single vector for classification.

In the proposed ViT-only model in Fig. 7 for the purpose of comparison, each vector represents a distinct patch. These vectors are first linearly projected into a high-dimensional

Table 7 Capability comparison between Transformer-based, CNN-based, and hybrid models in HARs

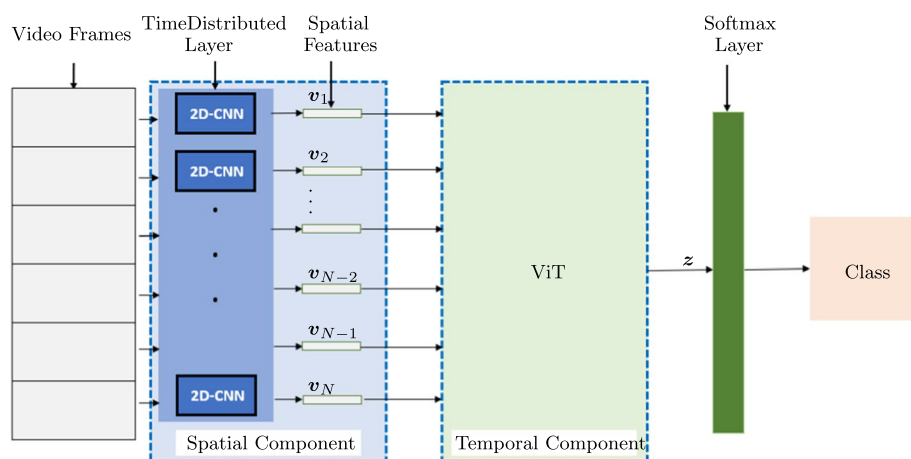| Criteria	ViT-based	CNN-based	Hybrid Models
Advantages			
Excel at capturing global dependencies	✓		✓
Scalable with data	✓		✓
Flexible in processing various input sizes	✓		✓
Adept at extracting local features		✓	✓
Computationally efficient		✓	
Robust to input variations		✓	✓
Efficient and versatile			✓
Adapts well to a range of tasks			✓
Disadvantages			
Computationally intensive	✓		✓
Requires substantial training data	✓		✓
Limited global contextual understanding		✓	
Prone to overfitting		✓	
Limited in modeling long-range dependencies		✓	
Architectural complexity			✓
Higher resource demands			✓
Need for careful tuning			✓
Balancing contributions of both components can be challenging			✓

Fig. 6 The hybrid CNN-ViT architecture for HARs

space, facilitating the model's ability to learn complex patterns within the data. To ensure the model captures the sequential nature of the input, positional encodings are added to these embeddings. The core of the ViT consists of two layers, each comprising a multi-head self-attention mechanism and a feed-forward network. The self-attention mechanism allows the model to weigh the importance of different patches relative to each other, while the feed-forward network, utilizing an exponential linear unit (ELU) activation function, processes each position independently to capture global context. The ViT is designed to aggregate the information from all vectors and positional encodings into a single [CLS] token, which is prepended to the input sequence. The output vector associated with this [CLS] token, after propagation through the Transformer layers, serves as a comprehensive representation of the entire input, suitable for downstream classification tasks.

4.1 Experiments

The goal of the presented experiments is not necessarily to produce a model that outperforms the state-of-the-art models in the HAR field. Rather, the aim is to conduct a comparison among the CNN, ViT-only, and hybrid models to give further insights.

The Royal Institute of Technology in 2004 unveiled the KTH dataset, a significant and publicly accessible dataset for action recognition (Schuldt et al. 2004). The KTH dataset was chosen here for its balanced representation of spatial and temporal features. Renowned as a benchmark dataset, it encompasses six types of actions: walking, jogging, running, boxing, hand-waving, and hand-clapping. The dataset features performances by 25 different individuals, introducing a diversity in execution. Additionally, the environment for each participant's actions was deliberately altered, including settings such as outdoors, outdoors with scale changes, outdoors with clothing variations, and indoors. The KTH dataset comprises 2,391 video sequences, all recorded at 25 frames per second using a stationary camera against uniform backgrounds.

Six experiments were conducted, with each of the aforementioned models trained on three different lengths of frame sequences. Care was taken to avoid pre-training in order
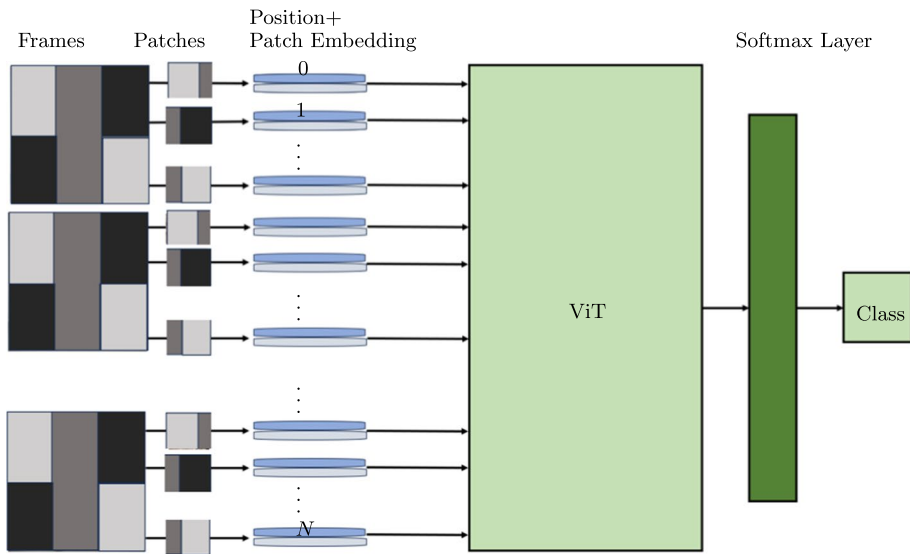
Fig. 7 The ViT-only architecture for HARs

to ensure the neutrality of the results. The TransNet model by Alomar and Cai (2023) was adopted to represent the CNN model, and the ViT model was depicted in Fig. 7. For the spatial component of the hybrid model, we employed the spatial component of TransNet; and for the temporal component, we employed the same ViT model that we used in the ViT-only model. We constructed our model utilizing Python 3.6, incorporating the Keras deep learning framework, OpenCV for image processing, matplotlib, and the scikit-learn library. The training and test were performed on a computer equipped with an Intel Core i7 processor, an NVidia RTX 2070 graphics card, and 64GB of RAM.

4.1.1 Results and discussion

Table 8 presents the quantitative results of the three distinct models, i.e., CNN, ViT-only, and a hybrid model on the KTH dataset, focusing on three different context lengths, i.e., short (12 frames), medium (18 frames), and long (24 frames). The results from these experiments provide insightful revelations into the efficacy of each model under different temporal contexts. More details are given below.

The CNN model exhibited a decrease in accuracy as the frame length increased, recording 94.35% for 12 frames, 93.91% for 18 frames, and 93.49% for 24 frames. This descending trend suggests that CNN may struggle with processing longer sequences where temporal dynamics become more complex, potentially leading to challenges such as overfitting or difficulties in temporal feature retention over extended durations.

In contrast, the ViT model demonstrated an improvement in performance with longer sequences, achieving accuracy of 92.44% for 12 frames, 92.82% for 18 frames, and 93.69% for 24 frames. This ascending pattern supports the notion that ViT architectures, with their inherent self-attention mechanisms, are well-suited to managing longer sequences. The abil-

Table 8 Experimental results of different models on the KTH Dataset using three different context lengths

Context length	CNN-based	ViT-only	Hybrid	Hybrid$_{pre}$
12 frames	94.35 ± 0.41	92.44 ± 0.16	94.12 ± 0.05	96.34 ± 0.03
18 frames	93.91 ± 0.32	92.82 ± 0.07	94.56 ± 0.10	97.13 ± 0.04
24 frames	93.49 ± 0.24	93.69 ± 0.08	95.78 ± 0.60	97.89 ± 0.05

In particular, the hybrid model was trained without pre-training, whereas Hybrid$_{pre}$ is for the hybrid model pre-trained on ImageNet. Every experiment was repeated over five runs to ensure robust statistical evaluation

ity of ViTs to assign varying degrees of importance to different parts of the sequence likely contributes to their enhanced performance on longer input frames.

The hybrid CNN-ViT model showcased the highest and continuously improving accuracy rates across all frame lengths: 94.12% for 12 frames, 94.56% for 18 frames, and an impressive 95.78% for 24 frames. Moreover, the pre-trained hybrid model showcased the same trend, with the best accuracy achieved. This type of model synergistically combines CNN's robust spatial feature extraction capabilities with ViT's efficient handling of temporal relationships via self-attention. The results from this model indicate that such a hybrid approach is particularly effective in capturing the complexities of action recognition tasks in video sequences, especially as the sequence length increases.

These findings underscore the potential advantages of hybrid neural network architectures in video-based action recognition tasks, particularly for handling longer sequences with complex interactions. The superior performance of the hybrid CNN-ViT model suggests that integrating the spatial acuity of CNNs with the temporal finesse of ViTs can lead to more accurate and reliable recognition systems. Future work could explore the scalability of these models to other datasets, their computational efficiency, and their robustness against variations in video quality and scene dynamics. Additionally, further research might investigate the optimal balance of CNN and ViT components within hybrid models to maximize both performance and efficiency.

To complete the comparison, Table 9 shows that the impressive 97.89% accuracy achieved by the presented CNN-ViT hybrid model on the KTH dataset places it prominently among state-of-the-art models for HAR. This performance is notably superior when compared to earlier benchmarks reported in the literature such as Geng and Song (2016) with 92.49% and Arunnehru et al. (2018) with 94.90%. Our model utilizes an ImageNet-pre-trained MobileNet (Howard et al. 2017) as the CNN backbone in the spatial component, which enhances its robust feature extraction capabilities. Combined with the dynamic attention mechanisms of ViT, it can thereby enhance both the spatial and temporal processing of video sequences. Furthermore, our hybrid model not only surpasses other contemporary approaches like Liu et al. (2020) (91.93%) and Lee et al. (2021) (89.40%), but also shows competitive/superior performance against some of the highest accuracy in the field, such as Jaouedi et al. (2020) (96.30%) and Basha et al. (2022) (96.53%). Even in comparison to the high benchmark set by Sahoo et al. (2020) (97.67%), our hybrid model demonstrates a marginal but significant improvement, underscoring the efficacy of integrating CNN with ViT. This integration not only facilitates more nuanced feature extraction across both spatial and sequential dimensions but also adapts more dynamically to the varied contexts inherent in video data, making it a potent solution for realistic action recognition scenarios.

On the whole, the integration of CNN with ViT is particularly advantageous for enhancing feature extraction capabilities and focusing on relevant segments dynamically through

Table 9 Comparison of the proposed hybrid model with the state-of-the-art models on the KTH dataset

Methods	Venue	Accuracy
Geng and Song (2016)	ICCSAE '16	92.49
Arunnehru et al. (2018)	RoSMa '18	94.90
Abdelbaky and Aly (2020)	ITCE '20	87.52
Jaouedi et al. (2020)	KSUCI journal '20	96.30
Liu et al. (2020)	JAIHC '20	91.93
Sahoo et al. (2020)	TETCI '20	97.67
Lee et al. (2021)	CVF '21	89.40
Basha et al. (2022)	MTA journal '22	96.53
Ye and Bilodeau (2023)	CVF '23	90.90
Ours	-	97.89

Table 10 The t-statistic values for the models in Table 8 across the three contexts

Context	CNN-based	Vit-only	Hybrid	Hybrid$_{pre}$
12 vs. 18 frames	6.52	19.75	51.12	70.19
12 vs. 24 frames	14.22	37.67	134.33	141.14
18 vs. 24 frames	6.59	58.14	301.23	73.14

the attention mechanisms of ViTs. This not only helps in improving accuracy but also in making the model more adaptable to varied video contexts, a key requirement for action recognition in realistic scenarios. This comparative advantage suggests that hybrid models are paving the way for future explorations in HAR, combining the best of convolutional and ViT-based architectures for improved performance and efficiency.

4.1.2 Statistical significance analysis

In this section, we present the statistical significance analysis used to evaluate the performance of the proposed model in comparison with benchmark models. The analysis here employs two statistical methods: the *paired-samples t-test* (see Algorithm 1 in Appendix) and the *one-sample t-test* (see Algorithm 2 in Appendix) (Montgomery and Runger 2020; Devore 2000). The symbols and variables used in Algorithms 1 and 2 are summarized in Table 12 in Appendix.

The paired-samples t-test algorithm evaluates different individual models in Table 8 whether there is a significant difference of the performance of a model between two related contexts among the total three contexts (i.e., 12, 18, and 24 frames). Applying Algorithm 1 on the quantitative results in Table 8, we obtain the t-statistic values t_{sp} given in Table 10 and the p-values p_p given in Table 11 for each model on paired contexts. For the paired-samples t-test, the null hypothesis (H_{p0}) posited that no significant difference exists between two contexts for each model. The alternative hypothesis (H_{p1}) suggested that a significant difference existed between two contexts for each model. The results in Table 11 demonstrates statistically significant differences across all contexts for each model, with p-values lower than the adjusted significance level α_a using the Bonferroni correction.

The one-sample t-test algorithm is used here to evaluate whether there is a significant difference between the performance of the proposed model and the mean performance of the benchmark models in Table 9. In this test, the null hypothesis (H_{o0}) assumes that the performance of the proposed model is not significantly different from the mean performance of

Context	CNN-based	ViT-only	Hybrid	Hybrid$_{pre}$
12 vs. 18 frames	2.95×10^{-3}	3.94×10^{-5}	8.85×10^{-7}	2.47×10^{-7}
12 vs. 24 frames	1.45×10^{-4}	2.97×10^{-6}	1.84×10^{-8}	1.51×10^{-8}
18 vs. 24 frames	2.76×10^{-3}	5.24×10^{-7}	7.31×10^{-10}	2.09×10^{-7}

Table 11 The two-tailed 5% p-value for the models in Table 8 across the three contexts

the benchmark models in Table 9. The alternative hypothesis (H_{o1}) posits that a significant difference does exist. By applying Algorithm 2 on the data in Table 9, we obtain a p-value of $p_o = 0.0034$, which is significantly lower than the commonly accepted significance level of 0.05. As a result, we reject the null hypothesis. This finding indicates that the performance difference between the proposed model and the state-of-the-art models is statistically significant. Consequently, we can conclude with 95% confidence that the proposed model outperforms the current state-of-the-art models for the HAR task under consideration. This result highlights the effectiveness of the proposed model in advancing the field.

5 Challenges and future directions

The field of HAR faces several formidable challenges that stem from the inherent complexity of interpreting human movements within diverse and dynamic environments. One of the primary obstacles is the variability in human actions themselves, which can differ significantly in speed, scale, and execution from one individual to another (Pareek and Thakkar 2021). This variability necessitates the development of sophisticated models capable of generalizing across a wide range of actions without sacrificing accuracy (Nayak et al. 2021). Additionally, the presence of complex backgrounds and environments further complicates the task of HAR. Systems must be adept at isolating and recognizing human actions against a backdrop of potentially distracting or obstructive elements, which can vary from the bustling activity of a city street to the unpredictable conditions of outdoor settings (Wang and Schmid 2013; He et al. 2016).

HAR systems furthermore must navigate the fine line between inter-class similarity and intra-class variability, where actions that are similar to each other (such as running versus jogging) require nuanced differentiation, while the same action can appear markedly different when performed by different individuals or under varying circumstances (Gong et al. 2020; Zhu and Yang 2018). The challenge of temporal segmentation adds another layer of complexity, as accurately determining the start and end of an action within a continuous video stream is crucial for effective recognition (Zolfaghari et al. 2018). Coupled with the need for computational efficiency to process video data in real-time and the difficulties associated with obtaining large, accurately annotated datasets, these challenges underscore the multifaceted nature of HAR (Caba Heilbron et al. 2015). Addressing these issues is critical for advancing the field and enhancing the practical applicability of HAR systems in real-world applications, from surveillance and security to healthcare and entertainment.

The motivation behind this work has been driven by the compelling need to bridge the existing gaps between the spatial feature extraction capabilities inherent in CNNs and the dynamic temporal processing strengths found in ViTs (Arnab et al. 2021). Through the introduction of a novel hybrid model, an attempt has been made to leverage the synergistic

potential of these technologies, thereby enhancing the accuracy and efficiency of HAR systems in capturing the complex spatial-temporal dynamics of human actions.

Looking forward, a promising future for HAR is envisioned, particularly through the development of hybrid and integrated models. It is believed that the potential of these models extends beyond immediate performance improvements, inspiring new directions for research within the field. It is anticipated that future studies will focus on optimizing these hybrid architectures, aiming to make them more scalable and adaptable to real-world applications across various domains such as surveillance, healthcare, and interactive media. Furthermore, the exploration of self-attention mechanisms and the adaptation of large-scale pre-training strategies from ViTs are seen as exciting prospects for HAR. These approaches are expected to lead to the development of more sophisticated models capable of understanding and interpreting human actions with unprecedented accuracy and nuance.

The integration of CNNs and ViTs into hybrid CNN-ViT models presents a promising avenue for overcoming the challenges faced by HAR systems. These hybrid models capitalize on the strengths of both architectures: the local feature extraction capabilities of CNNs and the global context understanding of ViTs. Future developments could focus on enhancing model adaptability to generalize across diverse actions, improving the isolation of human actions from complex backgrounds through advanced attention mechanisms, and developing nuanced differentiation techniques for closely related actions (Carion et al. 2020). Innovations in model architecture, alongside the application of transfer learning and few-shot learning techniques, could significantly reduce the variability challenge in human actions.

Moreover, addressing the temporal segmentation challenge requires the integration of specialized temporal modules and sequence-to-sequence models to accurately determine the start and end of an action within continuous video streams. Computational efficiency remains paramount for real-time processing, necessitating ongoing efforts in model optimization and the exploration of synthetic data generation to mitigate the difficulties associated with obtaining large and accurately annotated datasets. Customizable hybrid CNN-ViT models that can be tailored for specific applications, from surveillance to healthcare, will ensure that these advancements not only push the boundaries of academic research but also enhance practical applicability in real-world scenarios. Through these concerted efforts, hybrid CNN-ViT models are poised to make significant contributions to the field of HAR, offering innovative solutions to its multifaceted challenges.

This work has highlighted the importance of continued innovation and cross-disciplinary collaboration in the advancement of HAR technologies. By integrating insights from computer vision, machine learning, and domain-specific knowledge, it is hoped that HAR systems will not only become more efficient and accurate but also more responsive to the complexities and variances of human behavior in natural environments. As the field moves forward, the focus is set on pushing the boundaries of what is possible in HAR, with the aim of creating systems that enhance human-computer interaction and contribute positively to society through various applications.

6 Conclusions

This survey provides a comprehensive overview of the current state of HAR by examining the roles and advancements of CNNs, RNNs, and ViTs. It delves into the evolution of these architectures, emphasizing their individual contributions to the field. The introduction of a hybrid model that combines the spatial processing capabilities of CNNs with the temporal understanding of ViTs represents a methodological advancement in HAR. This model aims to address the limitations of each architecture when used in isolation, proposing a unified approach that potentially enhances the accuracy and efficiency of action recognition tasks. The paper identifies key challenges and opportunities within HAR, such as the need for models that can effectively integrate spatial and temporal information from video data. The exploration of hybrid models, as suggested, offers a pathway for future research, particularly in improving model performance on complex video datasets. The discussion encourages further investigation into optimizing these hybrid architectures and exploring their applicability across various domains. This work sets a foundation for future studies to build upon, aiming to push the boundaries of what is currently achievable in HAR and to explore new applications of these technologies in real-world scenarios.

	Symbol	Definition
Table 12 List of symbols and variables used in the paired-samples t-test (i.e., Algorithm 1) and one-sample t-test (i.e., Algorithm 2)	c_{1i}	Performance of a model in the first context (i.e., 12 frames) of the i-th run in the paired-samples test
	c_{2i}	Performance of a model in the second context (i.e., 18 frames) of the i-th run in the paired-samples test
	c_{3i}	Performance of a model in the third context (i.e., 24 frames) of the i-th run in the paired-samples test
	n_p	Number of paired observations (i.e., the number of runs) in the paired-samples t-test
	d_i	Differences between paired observations $(c_{1i} - c_{2i})$ in the paired-samples t-test
	$\bar{d}$	Mean of the differences between paired observations in the paired-samples t-test
	s_{dp}	Standard deviation of the differences in the paired-samples t-test
	t_{sp}	t-statistic value for the paired-samples t-test
	d_{fp}	Degrees of freedom for the paired-samples t-test, calculated as $n_p - 1$
	p_p	Two-tailed p-value for the paired-samples t-test
	n_c	Number of comparisons for the paired-samples t-test (i.e., 12 vs. 18, 12 vs. 24, and 18 vs. 24 frames)
	m_i	Performance of the state-of-the-art i-th model used in the one-sample t-test
	n_o	Population size, i.e., the number of state-of-the-art models
	μ_o	Mean performance of the state-of-the-art models
	s_{do}	Standard deviation of the performance of the state-of-the-art models
	t_{so}	t-statistic value for the one-sample t-test
	d_{fo}	Degrees of freedom for the one-sample t-test, calculated as $n_o - 1$
	p_o	Two-tailed p-value for the one-sample t-test
	m_p	Observed performance of the proposed model in the one-sample t-test
	α	Significance level for hypothesis testing, typically set at 0.05
	α_a	Adjusted significance level using the Bonferroni correction

Appendix A: Statistical significance analysis methods

This appendix presents two statistical significance analysis methods: the paired-samples t-test and the one-sample t-test (Montgomery and Runger 2020; Devore 2000) in Algorithms 1 and 2, respectively. The symbols and variables used in Algorithms 1 and 2 are summarized in Table 12.

Algorithm 1 processes the data in Table 8 from five experimental runs for each model across two contexts out of the total three contexts. It pairs the results from the first run of each context, followed by pairing the results from the second run of each context, continuing in this manner until all five runs have been paired. The algorithm then computes the two-tailed p-value, denoted as p_p, for the paired-samples t-test. Algorithm 2 utilizes the performance results of the state-of-the-art models along with the performance result of the proposed model from Table 9. The algorithm then computes the two-tailed p-value, denoted as p_o, for the one-sample t-test.

1: **Input:** The model performance on two different frame contexts: (c_{1i}, c_{2i}), where $i = 1, 2, \ldots, n_p$ with $n_p = 5$ (experimental runs); the number of comparisons $n_c = 3$; and the significance level $\alpha = 0.05$.

2: **Output:** Two-tailed p-value p_p.

3: Calculate the differences between the paired observations:

$$d_i = c_{1i} - c_{2i}, \quad i = 1, 2, \ldots, n_p.$$

4: Compute the mean of the differences:

$$\bar{d} = \frac{\sum_{i=1}^{n_p} d_i}{n_p}.$$

5: Compute the standard deviation of the differences:

$$s_{dp} = \sqrt{\frac{\sum_{i=1}^{n_p}(d_i - \bar{d})^2}{n_p - 1}}.$$

6: Calculate the t-statistic value:

$$t_{sp} = \frac{\bar{d}}{s_{dp}/\sqrt{n_p}}.$$

7: Determine the degrees of freedom:

$$d_{fp} = n_p - 1.$$

8: Calculate the two-tailed p-value:

$$p_p = 2 \times f_{cp}(|t_{sp}|, d_{fp}),$$

where the f_{cp} function uses a statistical t-distribution table/software (e.g., the SciPy library and Python programming language) to find the critical t-value corresponding to the calculated t-statistic (t_{sp}) and degrees of freedom (d_{fp}).

9: Apply the Bonferroni correction:

$$\alpha_a = \alpha/n_c,$$

where n_c is the number of comparisons and $n_c = 3$ for the case in Table 8.

10: **if** $p_p < \alpha_a$ **then**

11:    Reject the null hypothesis (H_{p0}), i.e., there is a significant difference.

12: **else**

13:    Fail to reject the null hypothesis, i.e., there is not enough evidence to suggest a significant difference.

14: **end if**

Algorithm 1 Paired-samples t-test algorithm

1: **Input:** Performance results of state-of-the-art models: $m_1, m_2, \ldots, m_{n_o}$; the performance result of the proposed model: m_p; and the significance level $\alpha = 0.05$.

2: **Output:** Two-tailed p-value p_o.

3: Compute the population mean:
$$\mu_0 = \frac{\sum_{i=1}^{n_o} m_i}{n_o}.$$

4: Compute the population standard deviation:
$$s_{do} = \sqrt{\frac{\sum_{i=1}^{n_o} (m_i - \mu_0)^2}{n_o - 1}}.$$

5: Calculate the t-statistic:
$$t_{so} = \frac{m_p - \mu_0}{s_{do}/\sqrt{n_o}}.$$

6: Determine the degrees of freedom:
$$d_{fo} = n_o - 1.$$

7: Calculate the two-tailed p-value:
$$p_o = 2 \times f_{co}(|t_{so}|, d_{fo}),$$
where the f_{co} function uses a statistical t-distribution table/software (e.g., the SciPy library and Python programming language) to find the critical t-value corresponding to the calculated t-statistic (t_{so}) and degrees of freedom (d_{fo}).

8: **if** $p_o < \alpha$ **then**

9:     Reject the null hypothesis (H_{o0}), i.e., there is a significant difference.

10: **else**

11:     Fail to reject the null hypothesis, i.e., there is not enough evidence to suggest a significant difference.

12: **end if**

Algorithm 2 One-sample t-test algorithm

Declarations

Conflict of interest The authors declare no Conflict of interest.

References

Abdelbaky A, Aly S (2020) Human action recognition based on simple deep convolution network Pcanet. In 2020 International conference on innovative trends in communication and computer engineering (ITCE). IEEE, pp. 257–262

Ahmadabadi H, Manzari ON, Ayatollahi A (2023) Distilling knowledge from CNN-transformer models for enhanced human action recognition. In 2023 13th International conference on computer and knowledge engineering (ICCKE). IEEE, pp. 180–184

Alomar K, Cai X (2023) TransNet: A transfer learning-based network for human action recognition. In 2023 International conference on machine learning and applications (ICMLA). IEEE, pp. 1825–1832

Arnab A, Dehghani M, Heigold G, Sun C, Lučić M, Schmid C (2021) Vivit: a video vision transformer. In Proceedings of the IEEE/CVF international conference on computer vision, pp. 6836–6846

Arunnehru J, Chamundeeswari G, Bharathi SP (2018) Human action recognition using 3d convolutional neural networks with 3d motion cuboids in surveillance videos. Procedia Comput Sci 133:471–477

Bahdanau D, Cho K, Bengio Y (2014) Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473

Barron JL, Fleet DJ, Beauchemin SS (1994) Performance of optical flow techniques. Int J Comput Vis 12:43–77

Basha SS, Pulabaigari V, Mukherjee S (2022) An information-rich sampling technique over spatio-temporal cnn for classification of human actions in videos. Multimedia Tools Appl 81(28):40431–40449

Bengio Y, Simard P, Frasconi P (1994) Learning long-term dependencies with gradient descent is difficult. IEEE Trans Neural Netw 5(2):157–166

Bertasius G, Wang H, Torresani L (2021) Is space-time attention all you need for video understanding? ICML 2:4

Brauwers G, Frasincar F (2021) A general survey on attention mechanisms in deep learning. IEEE Trans Knowl Data Eng 35(4):3279–3298

Brown T, Mann B, Ryder N, Subbiah M, Kaplan JD, Dhariwal P, Neelakantan A, Shyam P, Sastry G, Askell A et al (2020) Language models are few-shot learners. Adv Neural Inf Process Syst 33:1877–1901

Caba Heilbron F, Escorcia V, Ghanem B, Carlos Niebles J (2015) Activitynet: a large-scale video benchmark for human activity understanding. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 961–970

Carion N, Massa F, Synnaeve G, Usunier N, Kirillov A, Zagoruyko S (2020) End-to-end object detection with transformers. In European conference on computer vision. Springer, pp. 213–229

Carreira J, Zisserman A (2017) Quo Vadis, action recognition? A new model and the kinetics dataset. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 6299–6308

Chen J, Ho CM (2022) MM-ViT: Multi-modal video transformer for compressed video action recognition. In Proceedings of the IEEE/CVF winter conference on applications of computer vision, pp. 1910–1921

Chen T, Mo L (2023) Swin-fusion: swin-transformer with feature fusion for human action recognition. Neural Process Lett 55(8):11109–11130

Cho K, Van Merriënboer B, Gulcehre C, Bahdanau D, Bougares F, Schwenk H, Bengio Y (2014) Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078

Chung J, Gulcehre C, Cho K, Bengio Y (2014) Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv preprint arXiv:1412.3555

Cortes C, Vapnik V (1995) Support-vector networks. Mach Learn 20:273–297

Cosmin Duta I, Ionescu B, Aizawa K, Sebe N (2017) Spatio-temporal vector of locally max pooled features for action recognition in videos. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 3097–3106

Dalal N, Triggs B (2005) Histograms of oriented gradients for human detection. In 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05). IEEE, Volume 1, pp. 886–893

Dar G, Geva M, Gupta A, Berant J (2022) Analyzing transformers in embedding space. arXiv preprint arXiv:2209.02535

Deng J, Dong W, Socher R, Li LJ, Li K, Fei-Fei L (2009) Imagenet: a large-scale hierarchical image database. In 2009 IEEE conference on computer vision and pattern recognition. IEEE, pp. 248–255

Devlin J, Chang MW, Lee K, Toutanova K (2019) BERT: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, Volume 1 (Long and Short Papers), pp. 4171–4186

Devore JL (2000) Probability and statistics. Brooks/Cole, Pacific Grove

Diba A, Fayyaz M, Sharma V, Karami AH, Arzani MM, Yousefzadeh R, Van Gool L (2017) Temporal 3d convnets: new architecture and transfer learning for video classification. arXiv preprint arXiv:1711.08200

Djenouri Y, Belbachir AN (2023) A hybrid visual transformer for efficient deep human activity recognition. In Proceedings of the IEEE/CVF international conference on computer vision, pp. 721–730

Donahue J, Anne Hendricks L, Guadarrama S, Rohrbach M, Venugopalan S, Saenko K, Darrell T (2015) Long-term recurrent convolutional networks for visual recognition and description. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2625–2634

Dosovitskiy A, Beyer L, Kolesnikov A, Weissenborn D, Zhai X, Unterthiner T, Dehghani M, Minderer M, Heigold G, Gelly S et al (2020) An image is worth 16x16 words: transformers for image recognition at scale. arXiv preprint arXiv:2010.11929

Fan H, Xiong B, Mangalam K, Li Y, Yan Z, Malik J, Feichtenhofer C (2021) Multiscale vision transformers. In Proceedings of the IEEE/CVF international conference on computer vision, pp. 6824–6835

Feichtenhofer C (2020) X3d: Expanding architectures for efficient video recognition. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 203–213

Feichtenhofer C, Pinz A, Zisserman A (2016) Convolutional two-stream network fusion for video action recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1933–1941

Feichtenhofer C, Fan H, Malik J, He K (2019) Slowfast networks for video recognition. In Proceedings of the IEEE/CVF international conference on computer vision, pp. 6202–6211

Fukushima K (1980) Neocognitron: a self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. Biol Cybern 36(4):193–202

Geng C, Song J (2016) Human action recognition based on convolutional neural networks with a convolutional auto-encoder. In 2015 5th international conference on computer sciences and automation engineering (ICCSAE 2015). Atlantis Press, pp. 933–938

Gers FA, Schmidhuber J, Cummins F (2000) Learning to forget: continual prediction with lstm. Neural Comput 12(10):2451–2471

Ghadiyaram D, Tran D, Mahajan D (2019) Large-scale weakly-supervised pre-training for video action recognition. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 12046–12055

Gong G, Wang X, Mu Y, Tian Q (2020) Learning temporal co-attention models for unsupervised video action localization. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 9819–9828

Graves A (2013) Generating sequences with recurrent neural networks. arXiv preprint arXiv:1308.0850

Graves A, Mohamed AR, Hinton G (2013) Speech recognition with deep recurrent neural networks. In 2013 IEEE international conference on acoustics, speech and signal processing. IEEE, pp. 6645–6649

Han K, Wang Y, Chen H, Chen X, Guo J, Liu Z, Tang Y, Xiao A, Xu C, Xu Y et al (2022) A survey on vision transformer. IEEE Trans Pattern Anal Mach Intell 45(1):87–110

Hara K, Kataoka H, Satoh Y (2018) Can spatiotemporal 3D CNNs retrace the history of 2D CNNs and ImageNet? In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 6546–6555

He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770–778

Hochreiter S, Schmidhuber J (1997) Long short-term memory. Neural Comput 9(8):1735–1780

Howard AG, Zhu M, Chen B, Kalenichenko D, Wang W, Weyand T, Andreetto M, Adam H (2017) Mobilenets: efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861

Hu Y, Wong Y, Wei W, Du Y, Kankanhalli M, Geng W (2018) A novel attention-based hybrid cnn-rnn architecture for semg-based gesture recognition. PLoS ONE 13(10):e0206049

Hussain A, Hussain T, Ullah W, Baik SW (2022) Vision transformer and deep sequence learning for human activity recognition in surveillance videos. Comput Intell Neurosci 2022(1):3454167

Hussain A, Khan SU, Khan N, Rida I, Alharbi M, Baik SW (2023) Low-light aware framework for human activity recognition via optimized dual stream parallel network. Alex Eng J 74:569–583

Hussain A, Khan SU, Khan N, Shabaz M, Baik SW (2024a) Ai-driven behavior biometrics framework for robust human activity recognition in surveillance systems. Eng Appl Artif Intell 127:107218

Hussain A, Khan SU, Khan N, Ullah W, Alkhayyat A, Alharbi M, Baik SW (2024b) Shots segmentation-based optimized dual-stream framework for robust human activity recognition in surveillance video. Alex Eng J 91:632–647

Hussain A, Khan SU, Rida I, Khan N, Baik SW (2024c) Human centric attention with deep multiscale feature fusion framework for activity recognition in internet of medical things. Inf Fusion 106:102211

Hussain A, Hussain T, Ullah W, Khan SU, Kim MJ, Muhammad K, Del Ser J, Baik SW (2024d) Big data analysis for industrial activity recognition using attention-inspired sequential temporal convolution network. IEEE Trans Big Data

Hussain A, Khan SU, Khan N, Bhatt MW, Farouk A, Bhola J, Baik SW (2024e) A hybrid transformer framework for efficient activity recognition using consumer electronics. IEEE Trans Consumer Electron

Hussain A, Khan N, Munsif M, Kim MJ, SW Baik (2024f) Medium scale benchmark for cricket excited actions understanding. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 3399–3409

Jaouedi N, Boujnah N, Bouhlel MS (2020) A new hybrid deep learning model for human action recognition. J King Saud Univ-Comput Inf Sci 32(4):447–453

Ji S, Xu W, Yang M, Yu K (2012) 3d convolutional neural networks for human action recognition. IEEE Trans Pattern Anal Mach Intell 35(1):221–231

Jordan M (1986) Serial order: a parallel distributed processing approach. Technical Report, June 1985–March 1986. Technical report, California Univ., San Diego, La Jolla (USA). Inst. for Cognitive Science

Kalchbrenner N, Blunsom P (2013) Recurrent continuous translation models. In Proceedings of the 2013 conference on empirical methods in natural language processing, pp. 1700–1709

Karpathy A, Toderici G, Shetty S, Leung T, Sukthankar R, Fei-Fei L (2014) Large-scale video classification with convolutional neural networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1725–1732

Khan S, Naseer M, Hayat M, Zamir SW, Khan FS, Shah M (2022) Transformers in vision: a survey. ACM Comput Surv (CSUR) 54(10s):1–41

Kong Y, Fu Y (2022) Human action recognition and prediction: a survey. Int J Comput Vis 130(5):1366–1401

Koot R, Hennerbichler M, Lu H (2021) Evaluating transformers for lightweight action recognition. arXiv preprint arXiv:2111.09641

Kopuklu O, Kose N, Gunduz A, Rigoll G (2019) Resource efficient 3D convolutional neural networks. In Proceedings of the IEEE/CVF international conference on computer vision workshops

Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. Adv Neural Inf Process Syst 25

Kuehne H, Jhuang H, Poggio E, Poggio T, Serre T (2011) HMDB: a large video database for human motion recognition. In 2011 International conference on computer vision. IEEE, pp. 2556–2563

LeCun Y, Bottou L, Bengio Y, Haffner P (1998) Gradient-based learning applied to document recognition. Proc IEEE 86(11):2278–2324

LeCun Y, Bengio Y, Hinton G (2015) Deep learning. Nature 521(7553):436–444

Lee S, Kim HG, Choi DH, Kim HI, Ro YM (2021) Video prediction recalling long-term motion context via memory alignment learning. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 3054–3063

Leong MC, Zhang H, Tan HL, Li L, Lim JH (2022) Combined CNN transformer encoder for enhanced fine-grained human action recognition. arXiv preprint arXiv:2208.01897

Li Y, Lan C, Xing J, Zeng W, Yuan C, Liu J (2016) Online human action detection using joint classification-regression recurrent neural networks. In Computer vision–ECCV 2016: 14th European conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part VII 14. Springer, pp. 203–220

Li Z, Gavrilyuk K, Gavves E, Jain M, Snoek CG (2018) Videolstm convolves, attends and flows for action recognition. Comput Vis Image Underst 166:41–50

Li J, Liu X, Zhang M, Wang D (2020) Spatio-temporal deformable 3d convnets with attention for action recognition. Pattern Recogn 98:107037

Lin T, Wang Y, Liu X, Qiu X (2022) A survey of transformers AI Open 3:111–132

Liu X, Qi DL, Xiao HB (2020) Construction and evaluation of the human behavior recognition model in kinematics under deep learning. J Ambient Intell Human Comput, 1–9

Liu Z, Ning J, Cao Y, Wei Y, Zhang Z, Lin S, Hu H (2022) Video swim transformer. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 3202–3211

Luong MT, Pham H, Manning CD (2015) Effective approaches to attention-based neural machine translation. arXiv preprint arXiv:1508.04025

Malik NUR, Abu-Bakar SAR, Sheikh UU, Channa A, Popescu N (2023) Cascading pose features with cnn-lstm for multiview human action recognition. Signals 4(1):40–55

Mazzeo PL, Spagnolo P, Fasano M, Distante C (2022) Human action recognition with transformers. In International conference on image analysis and processing. Springer, pp. 230–241

Montgomery DC, Runger GC (2020) Applied statistics and probability for engineers. Wiley

Muhammad K, Ullah A, Imran AS, Sajjad M, Kiran MS, Sannino G, Albuquerque VHC et al (2021) Human action recognition using attention based lstm network with dilated cnn features. Future Gener Comput Syst 125:820–830

Munsif M, Khan SU, Khan N, Hussain A, Kim MJ, Baik SW (2024a) Contextual visual and motion salient fusion framework for action recognition in dark environments. Knowl-Based Syst 304:112480

Munsif M, Khan N, Hussain A, Kim MJ, Baik SW (2024b) Darkness-adaptive action recognition: leveraging efficient Tubelet slow-fast network for industrial applications. IEEE Trans Ind Inf

Mutegeki R, Han DS (2020) A CNN-LSTM approach to human activity recognition. In 2020 International conference on artificial intelligence in information and communication (ICAIIC). IEEE, pp. 362–366

Nayak R, Pati UC, Das SK (2021) A comprehensive review on deep learning-based methods for video anomaly detection. Image Vis Comput 106:104078

Pareek P, Thakkar A (2021) A survey on video-based human action recognition: recent updates, datasets, challenges, and applications. Artif Intell Rev 54:2259–2322

Peng Y, Zhao Y, Zhang J (2018) Two-stream collaborative learning with spatial-temporal attention for video classification. IEEE Trans Circ Syst Video Technol 29(3):773–786

Qiu Z, Yao T, Mei T (2017) Learning Spatio-temporal representation with pseudo-3D residual networks. In Proceedings of the IEEE international conference on computer vision, pp. 5533–5541

Reda DR, Chaieb F, Drira H, Aberkane A (2023) ConViViT-A deep neural network combining convolutions and factorized self-attention for human activity recognition. In 2023 IEEE 25th international workshop on multimedia signal processing (MMSP). IEEE, pp. 1–6

Rumelhart DE, Hinton GE, Williams RJ (1985) Learning internal representations by error propagation. Technical report, California Univ San Diego La Jolla Inst for Cognitive Science

Sahoo SP, Ari S, Mahapatra K, Mohanty SP (2020) Har-depth: a novel framework for human action recognition using sequential learning and depth estimated history images. IEEE Trans Emerg Top Comput Intell 5(5):813–825

Schuldt C, Laptev I, Caputo B (2004) Recognizing human actions: a local SVM approach. In Proceedings of the 17th international conference on pattern recognition, 2004. ICPR 2004. IEEE, Volume 3, pp. 32–36

Shan G, Ji Q, Xie Y (2021) Multi-view vision transformer for driver action recognition. In International conference on intelligent transportation engineering. Springer, pp. 970–981

Sharir G, Noy A, Zelnik-Manor L (2021) An image is worth 16x16 words, what is a video worth? arXiv preprint arXiv:2103.13915

Shi S, Jiang L, Dai D, Schiele B (2022) Motion transformer with global intention localization and local movement refinement. Adv Neural Inf Process Syst 35:6531–6543

Simonyan K, Zisserman A (2014a) Two-stream convolutional networks for action recognition in videos. Adv Neural Inf Process Syst 27

Simonyan K, Zisserman A (2014b) Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556

Soomro K, Zamir AR, Shah M (2012) UCF101: A dataset of 101 human actions classes from videos in the wild. arXiv preprint arXiv:1212.0402

Srivastava N, Mansimov E, Salakhudinov R (2015) Unsupervised learning of video representations using LSTMs. In International conference on machine learning. PMLR, pp. 843–852

Sun Z, Ke Q, Rahmani H, Bennamoun M, Wang G, Liu J (2022) Human action recognition from various data modalities: a review. IEEE Trans Pattern Anal Mach Intell 45(3):3200–3225

Surek G, Seman L, Stefenon S, Mariani V, Coelho L (2023) Video-based human activity recognition using deep learning approaches. Sensors 23(14):6384

Sutskever I, Vinyals O, Le QV (2014) Sequence to sequence learning with neural networks. Adv Neural Inf Process Syst 27

Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A (2015) Going deeper with convolutions. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1–9

Touvron H, Cord M, Douze M, Massa F, Sablayrolles A, Jégou H (2021) Training data-efficient image transformers & distillation through attention. In International conference on machine learning. PMLR, pp. 10347–10357

Tran D, Bourdev L, Fergus R, Torresani L, Paluri M (2015) Learning spatiotemporal features with 3D convolutional networks. In Proceedings of the IEEE international conference on computer vision, pp. 4489–4497

Tran D, Wang H, Torresani L, Ray J, LeCun Y, Paluri M (2018) A closer look at spatiotemporal convolutions for action recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 6450–6459

Tran D, Wang H, Torresani L, Feiszli M (2019) Video classification with channel-separated convolutional networks. In Proceedings of the IEEE/CVF international conference on computer vision, pp. 5552–5561

Ullah A, Ahmad J, Muhammad K, Sajjad M, Baik SW (2017) Action recognition in video sequences using deep bi-directional lstm with cnn features. IEEE Access 6:1155–1166

Ulhaq A, Akhtar N, Pogrebna G, Mian A (2022) Vision transformers for action recognition: a survey. arXiv preprint arXiv:2209.05700

Varol G, Laptev I, Schmid C (2017) Long-term temporal convolutions for action recognition. IEEE Trans Pattern Anal Mach Intell 40(6):1510–1517

Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I (2017) Attention is all you need. Adv Neural Inf Process Syst 30

Wang H, Schmid C (2013) Action recognition with improved trajectories. In Proceedings of the IEEE international conference on computer vision, pp. 3551–3558

Wang L, Xiong Y, Wang Z, Qiao Y (2015) Towards good practices for very deep two-stream convnets. arXiv preprint arXiv:1507.02159

Wang L, Xiong Y, Wang Z, Qiao Y, Lin D, Tang X, Van Gool L (2016) Temporal segment networks: towards good practices for deep action recognition. In European conference on computer vision. Springer, pp. 20–36

Wang Y, Long M, Wang J, Yu PS (2017) Spatiotemporal pyramid network for video action recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1529–1538

Wang L, Xiong Y, Wang Z, Qiao Y, Lin D, Tang X, Van Gool L (2018a) Temporal segment networks for action recognition in videos. IEEE Trans Pattern Anal Mach Intell 41(11):2740–2755

Wang X, Girshick R, Gupta A, He K (2018b) Non-local neural networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 7794–7803

Wang L, Tong Z, Ji B, Wu G (2021a) Tdn: Temporal difference networks for efficient action recognition. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 1895–1904

Wang M, Xing J, Liu Y (2021b) Actionclip: a new paradigm for video action recognition. arXiv preprint arXiv:2109.08472

Wu Z, Wang X, Jiang YG, Ye H, Xue X (2015) Modeling spatial-temporal clues in a hybrid deep learning framework for video classification. In Proceedings of the 23rd ACM international conference on multimedia, pp. 461–470

Xia K, Huang J, Wang H (2020) Lstm-cnn architecture for human activity recognition. IEEE Access 8:56855–56866

Xie S, Sun C, Huang J, Tu Z, Murphy K (2018) Rethinking Spatiotemporal feature learning: speed-accuracy trade-offs in video classification. In Proceedings of the European conference on computer vision (ECCV), pp. 305–321

Xing Z, Dai Q, Hu H, Chen J, Wu Z, Jiang YG (2023) Svformer: semi-supervised video transformer for action recognition. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 18816–18826

Ye X, Bilodeau GA (2023) A unified model for continuous conditional video prediction. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 3604–3613

Yin R, Yin J (2024) A two-stream hybrid CNN-transformer network for skeleton-based human interaction recognition. In Chinese conference on pattern recognition and computer vision (PRCV). Springer, pp. 395–408

Yue-Hei Ng J, Hausknecht M, Vijayanarasimhan S, Vinyals O, Monga R, Toderici G (2015) Beyond short snippets: deep networks for video classification. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 4694–4702

Zhu L, Yang Y (2018) Compound memory networks for few-shot video classification. In Proceedings of the European conference on computer vision (ECCV), pp. 751–766

Zolfaghari M, Singh K, Brox T (2018) Eco: efficient convolutional network for online video understanding. In Proceedings of the European conference on computer vision (ECCV), pp. 695–712

Authors and Affiliations

Khaled Alomar[1] · Halil Ibrahim Aysel[2] · Xiaohao Cai[2]

✉ Khaled Alomar
  kaaalomar@imamu.edu.sa

✉ Halil Ibrahim Aysel
  hia1v20@soton.ac.uk

  Xiaohao Cai
  x.cai@soton.ac.uk

[1] Imam Mohammad Ibn Saud Islamic University (IMSIU), Riyadh, Saudi Arabia

[2] Electronics and Computer Science, University of Southampton, Southampton SO17 1BJ, UK