A Survey of Deep Learning-Based Video Action Recognition Techniques: Postprint

Authors: Li Chen, He Ming, Wang Yong, Luo Ling, Han Wei

Date: 2022-05-10T11:22:58+00:00

Abstract

Action Recognition (AR) represents a research hotspot in the field of computer vision, with extensive application prospects in security surveillance, autonomous driving, production safety, and other domains. First, the connotation and extension of action recognition are analyzed, and the technical challenges encountered are identified. Second, the working principles of action recognition are analyzed and compared from three perspectives: temporal feature extraction, high-efficiency optimization, and long-term feature capture. Third, the performance characteristics of 43 benchmark AR methods from the past decade are compared on the UCF101, HMDB51, Something-Something, and Kinetics400 datasets, facilitating the selection of appropriate AR models for different application scenarios. Finally, future development directions for the action recognition field are indicated, and the research findings can provide theoretical references and technical support for video feature extraction and visual content understanding.

Full Text

Preamble

Review of Video Action Recognition Technology Based on Deep Learning

Li Chen, He Ming†, Wang Yong, Luo Ling, Han Wei
(Command & Control Engineering College, Army Engineering University of PLA, Nanjing 210007, China)

Abstract: Action recognition (AR) is a hot research area in computer vision with extensive application prospects in security monitoring, autonomous driving, production safety, and other domains. This paper first analyzes the connotation and denotation of AR and identifies key technical challenges. Second, it analyzes and compares the working principles of AR from three perspectives: temporal feature extraction, efficient optimization, and long-term feature capture. Third, to facilitate selection of appropriate AR models for different application scenarios, it compares the performance characteristics of 43 benchmark AR methods from the past decade on UCF101, HMDB51, Something-Something, and Kinetics400 datasets. Finally, the paper points out future development directions in the AR field, with research results providing theoretical reference and technical support for video feature extraction and visual content understanding.

Key words: action recognition; deep learning; convolutional neural networks; Transformer; RGB video

0 Introduction

Video data has become an important form of information presentation and is widely used across various industries. Consequently, video understanding technology—enabling computers to comprehend video content—has gradually emerged as a research hotspot. In 2017, the Computer Vision and Pattern Recognition Conference (CVPR) defined video understanding as comprising five subtasks: Untrimmed Video Classification, Trimmed Action Recognition, Temporal Action Proposal, Temporal Action Localization, and Dense-Captioning Events [1]. The action recognition (AR) discussed in this paper falls within the category of trimmed action recognition.

The focus of AR varies depending on the specific action categories and tasks. When "Action" represents single-person movements (such as abstract events like jumping, walking, or climbing), the action granularity is finer, requiring classification models with strong temporal modeling capabilities. When "Action" represents activities involving one or more persons (such as scene/object events like eating bread or playing soccer), recognition models can rely more on scene identification with less demanding temporal reasoning. "Recognition"carries two meanings: (a) classification, which involves categorizing behaviors in trimmed video clips; and (b) detection, which involves first localizing the start and end times of actions in untrimmed videos before classification. Additionally, input data exists in various forms including RGB video frames, skeleton diagrams, and depth maps. While AR research encompasses combinations of these concepts, this paper focuses primarily on action classification in trimmed RGB videos.

Feature extraction and classification constitute the core challenges of AR. Since video is a sequence of image frames over time, AR models must consider temporal information when extracting spatial features. Currently, two feature extraction approaches exist: First, manually designed features based on human sensitivity to various characteristics, creating feature extractors with physical meaning. While highly targeted, this approach neglects implicit information in data and suffers from poor generalization. Second, deep features extracted from data through deep learning, which designs model structures based on cortical visual theory and trains feature extractors using datasets and backpropagation algorithms. This approach is applicable to various data types but offers limited feature interpretability [2, 3]. While academia has not reached a definitive conclusion on the superiority of handcrafted versus deep features, this paper focuses on deep learning-based AR models, given that current classification tasks predominantly employ deep learning.

Although deep learning-based image recognition models have moved from labs to practical applications, AR—as a temporal extension of image classification—requires additional temporal feature extraction, hindering its real-world deployment. In summary, AR faces the following technical challenges:

a) **Difficulty in video dataset creation.** Improving recognition accuracy requires training on large-scale annotated datasets, but video data annotation and action localization are extremely time-consuming, constraining both dataset scale and AR model development.

b) **Low model training efficiency.** Video data volume grows exponentially compared to images, making the training process for AR models to fit spatiotemporal features highly demanding in terms of hardware configuration and time.

c) **High intra-class variance and low inter-class variance.** AR encompasses diverse behaviors with significant variation within the same action category while different categories may appear similar, imposing more refined requirements on feature extractors.

d) **Insufficient real-time performance.** Current AR models prioritize high accuracy over lightweight design, and most operate in offline environments with pre-trimmed videos, making online streaming video recognition difficult.

Research Status: Liu et al. [4] described the application workflow of action recognition in smart homes. Liu et al. [5] discussed skeleton-based action recognition methods using deep learning. Zhang et al. [6] analyzed abnormal behavior discrimination from both recognition and detection perspectives. Pei et al. [7] compared traditional methods with deep models. Distinguishing itself from these studies, this paper categorizes AR models from three perspectives—temporal feature extraction, efficient optimization, and long-term feature capture—while summarizing public video datasets and performance comparisons of mainstream and state-of-the-art models, as illustrated in Fig. 1.

1 Deep Models for Spatiotemporal Feature Extraction

In the early stages of AR development, handcrafted methods such as improved dense trajectories (iDT) [8] dominated. After Hinton et al. [9] discussed the principles and advantages of deep learning in 2015, AR research based on deep learning gradually expanded. Karpathy et al. [10] employed convolutional neural networks (CNN) to learn spatiotemporal features from stacked video frames for end-to-end action classification, evaluating various 2D CNN connection methods including late fusion, early fusion, and slow fusion. However, recognition accuracy remained far below traditional handcrafted methods, demonstrating that simple frame fusion cannot effectively extract temporal features.

Unlike image recognition, AR must focus on spatiotemporal features that include the temporal domain to understand motion information. This section analyzes and compares AR spatiotemporal feature extraction work across three strategies: two-stream convolution, 3D convolution, and temporal modeling.

1.1 Two-Stream Convolutional Models

When humans observe moving objects, continuous images flow across the retina, with pixel movements called optical flow [11, 12]. Optical flow carries motion information by representing image changes, making it an effective method for temporal feature extraction. Simonyan et al. [13] designed a two-stream network based on optical flow, where two 2D CNN pathways extract spatial and temporal features from video frames and stacked optical flow images, respectively. The two-stream network achieved performance comparable to iDT, validating the effectiveness of optical flow for AR.

Feichtenhofer et al. [14] explored various fusion methods based on the two-stream network. With the popularization of residual networks (ResNet), they connected two streams using ResNet in subsequent work [15, 16] to achieve residual interaction of spatiotemporal information. Building on the two-stream approach, Wang et al. [17] increased network depth based on the VGGNet-16 architecture and mitigated overfitting from deeper networks using small learning rates and restricted cropping regions.

Given the strong performance of two-stream networks, Wang et al. [18] placed deep features from two-stream networks at the center of iDT trajectories, constructing Trajectory-Pooled Deep-Convolutional Descriptors (TDD). TDD shared handcrafted and deep features, achieving higher discriminative power and automatic learning, making this fusion approach an effective method for improving AR accuracy. Ding et al. [19] improved the two-stream network architecture by introducing BN-Inception and ResNet, establishing a spatiotemporal heterogeneous two-stream network that validated the effectiveness of heterogeneous design.

In summary, two-stream networks significantly elevated the status of deep learning methods in video action recognition and gradually developed into an important branch of AR.

1.2 3D Convolutional Models

While optical flow can extract temporal features, it is sensitive to lighting changes, demands high storage and computational resources, and its small displacement characteristics struggle with high-speed actions. Since 2D convolution achieved excellent results in image recognition, researchers extended it directly to extract spatiotemporal features from videos.

Ji et al. [20] used 3D convolutional kernels to learn spatiotemporal features, proving the effectiveness of 3D convolution for AR. However, without detailed 3D CNN design, recognition accuracy remained inferior to two-stream networks and handcrafted methods. Later, C3D [21] achieved good recognition results using 3$×3×$3 3D convolutional kernels based on the VGG-16 architecture from image recognition. However, C3D' s accuracy still lagged behind two-stream networks, with large parameter counts that led to long training cycles and overfitting given the lack of large-scale datasets at the time. Additionally, gradient vanishing/explosion issues limited C3D' s depth extension.

Since ResNet can mitigate degradation problems in deep networks, Tran et al. [22] designed 3D Residual Networks (R3D), extending ResNet' s 2D convolutions to 3D and reducing parameters by nearly 50% compared to C3D. Hara et al. [23] further improved recognition accuracy through deeper extensions based on R3D. T3D [24] also improved upon C3D but used the DenseNet architecture, halving parameters compared to R3D though dense connections increased computational load.

Early 3D CNNs consistently underperformed optical flow-based two-stream networks until I3D [25] broke this impasse in 2017. Carreira et al. [25] reasoned that duplicating a single image from an image dataset multiple times could generate a "static video" for training 3D CNNs. Similarly, parameters from 2D convolutional kernels in image dataset pre-trained 2D CNNs could be duplicated along the temporal axis to initialize 3D CNNs, facilitating the use of mature architectures from image recognition for AR. They applied this idea to the 2D convolutional pathways of two-stream networks and further pre-trained on the Kinetics dataset for the first time, resulting in the Inflated 3D ConvNet (I3D). Deeper than C3D yet with fewer parameters, I3D became a benchmark method for AR.

In summary, 3D CNNs gradually surpassed optical flow-based two-stream networks to become another important branch of AR.

1.3 Temporal Convolutional Models

Two-stream networks and 3D CNNs generally involve high computational costs, hindering real-time applications, and exhibit weak temporal reasoning capabilities. Since AR models need to understand action information over time, some research focuses on designing temporal modules with temporal modeling mechanisms and low computational complexity.

Temporal Relation Network (TRN) [26] learns inter-frame temporal relations at multiple scales and can be plugged into CNN architectures, but suffers from training difficulties when input frames are numerous due to excessive modules. Based on the decomposability of 3D convolution into shift operations and multiply-accumulate operations, Temporal Shift Module (TSM) [27] shifts partial channels along the temporal axis to extract inter-frame information. TSM modules can be embedded into various 2D CNN recognition models, enabling efficient recognition without additional computation. TSM' s extension, TIN [28], performs shift operations on the channel dimension and automatically learns shift direction and activation, achieving slight accuracy improvements over TSM. TEI [29] modules separate channel correlation and temporal interaction modeling, while TAM [30] uses dynamic temporal convolutional kernels to adaptively aggregate temporal information. Temporal Excitation and Aggregation module (TEA) [31] proposes ME and MAT modules based on STM [32] to process short-range and long-range features.

Luo et al. [33] designed Spatial Convolutional Attention (SCA) and Temporal Convolutional Attention (TCA) modules. SCA uses self-attention to capture spatial feature relationships and 1D convolution to extract temporal features, while TCA obtains temporal features through self-attention and learns spatial features using 2D convolution. Wu et al. [34] proposed a channel-temporal module that preserves more effective channel and temporal information by adjusting the order of pooling and convolutional layers.

In summary, temporal convolutional methods can integrate spatiotemporal and motion features into 2D CNNs without requiring optical flow or 3D convolution, offering temporal modeling capabilities while reducing computational overhead.

2 Efficiency Optimization in Deep Models

While temporal convolutional models demonstrate temporal modeling with efficiency advantages, efficiency remains a crucial metric for AR models. Optical flow in two-stream CNNs is expensive in terms of storage and computation, and 3D CNNs involve large parameter counts and computational loads, prompting research into efficiency optimization for AR.

2.1 Input Data Optimization

Wang et al. noted that not all video frames contain useful information. Based on two-stream CNNs, they proposed Temporal Segment Networks (TSN) [35] using uniform sampling of video frames to improve efficiency. TSN reduces information redundancy and enables end-to-end learning at lower cost. The key frame mining framework [36] abandons random strategies by scoring frames to sample key frames, though gains are not significant. References [37, 38] considered all frames beneficial for classification tasks, clustering forward outputs from all frames to improve efficiency.

To address the computational difficulty of optical flow, FlowNet [39] and FlowNet2.0 [40] predict optical flow fields from images using neural networks. Piergiovanni et al. [41] proposed representation flow based on TV-L1 optical flow to simulate iterative optical flow parameters through end-to-end learning of flow convolutional layers. The hidden two-stream network [42] connects MotionNet, which can generate optical flow-like features from video frames, with temporal stream CNNs to alleviate optical flow computation overhead. Motion-Augmented RGB Stream (MARS) [43] uses trained optical flow to teach neural networks to learn optical flow performance. Zhang et al. [44] eliminated optical flow dependency through small displacements of motion boundaries.

2.2 Spatiotemporal Decomposition of 3D Convolution

Compared to 2D convolution, 3D convolution involves significantly larger parameter counts and computational loads. A 3D convolutional kernel has dimensions FC×FT×FH×FW, where FC represents the number of channels, FH×FW the spatial receptive field, and FT the temporal receptive field. Excluding channels FC, the spatiotemporal decomposition approach factorizes a 3D kernel of size FT×FH×FW into the outer product of a 2D spatial kernel (1×FH×FW) and a 1D temporal kernel (FT$×1×$1), as illustrated in Fig. 4.

Based on this decomposition, P3D [45] uses 1$×3×32*Dconvolutionand*3×1×11*Dconvolutiontosimulate*3×3×$3 3D convolution, significantly reducing parameters compared to C3D while enabling initialization from 2D CNNs. Tran et al. [46] proposed R(2+1)D, a similar structure to P3D-A with 2D convolution followed by 1D convolution. However, R(2+1)D leverages efficiency gains to increase channel numbers, improving accuracy over R3D. S3D [47] adopts a Top-heavy approach to simplify feature volume and optimize efficiency. Recently, Sudhakaran et al. [48] proposed Gate-Shift Modules (GSM) for 3D spatiotemporal decomposition, which can adaptively find and combine features over time with minimal additional parameters.

While spatiotemporal decomposition offers efficiency optimization, this rigid separability can affect optimal iteration of AR models and consequently impact accuracy.

2.3 Depthwise Separable 3D Convolution

Unlike spatiotemporal decomposition, depthwise separable convolution splits the convolutional kernel into different depth groups. As shown in Fig. 5, depthwise separable convolution decomposes a 3D kernel of dimension FC×FT×FH×FW into two parts: a depthwise convolution kernel (1×FT×FH×FW) and a pointwise convolution kernel (FC$×1×1×$1). The pointwise kernel performs weighted combination of features in the depth direction, with both parts working together within a Bottleneck structure to optimize model efficiency.

MFNet [49] applies depthwise separable convolution to ResNet, splitting ResNet modules into multi-fiber ResNet modules. Experiments demonstrate that MFNet reduces computation by 9× and 13× compared to I3D and R(2+1)D, respectively. Channel-Separated Convolutional Networks (CSN) [50] design three Bottleneck structures based on depthwise separable convolution on 3D ResNet modules, reducing computation by 2-3× compared to R(2+1)D. Grouped Spatial-Temporal Aggregation (GST) [51] improves upon P3D using depthwise separation to perform spatial and temporal operations on different channels for efficiency gains.

Depthwise separable convolution reduces parameters, but its depthwise convolution lacks cross-channel information, resulting in insufficient spatial correlation and hindering spatiotemporal feature extraction for AR models.

2.4 Hybrid 2D and 3D Convolution

Given the impact of convolutional decomposition on recognition performance, methods combining 2D and 3D convolution attempt to optimize efficiency while maintaining accuracy. MiCT [52] extends depth by concatenating 2D CNN after 3D convolution, while parallel 2D CNN prevents gradient vanishing and training errors from increased depth, effectively controlling 3D CNN complexity. Conversely, Efficient Convolutional Network (ECO) [53] connects 3D CNN after 2D CNN to obtain feature maps for classification, supporting fast processing capable of classifying 230 video segments within one second.

ARTnet [54] adopts a two-stream approach with 2D and 3D convolutions in separate streams to extract spatial and temporal features. The SlowFast [55] network resembles ARTnet’ s two-stream pathways but designs slow and fast pathways. As shown in Fig. 6, the slow pathway focuses on spatial features using low frame rates and larger channel numbers, accounting for about 80% of computation. The fast pathway focuses on temporal features using high frame rates and smaller channel numbers, accounting for about 20% of model computation. However, action tempos vary, requiring SlowFast to set different frame rates, which is impractical to predefine. To address this, Temporal Pyramid Network (TPN) [56] extracts pyramid-shaped feature maps at different levels using a single frame rate to represent various tempos, while BQN [57] automatically separates slow and fast information for greater generality. Liu et al. [58] proposed temporal zero-padding convolutional networks to reduce 3D CNN parameters, first using 3D convolution without temporal padding to extract spatiotemporal information, then converting 3D convolution to 2D convolution through network reorganization for further feature extraction.

In summary, AR efficiency optimization involves expansion or compression across depth, space, time, channels, and sampling, but manual settings provide suboptimal balance between accuracy and efficiency. Recently, X3D [59] automatically performed progressive expansion across metrics with evaluation feedback, achieving excellent accuracy while greatly improving runtime efficiency. Unlike X3D’ s defined expansion, MoViNet [60] uses neural architecture search to generate efficient and diverse 3D CNNs, achieving an excellent efficiency-accuracy balance.

3 Deep Models for Long-Term Feature Capture

Previous models extract short-term action features, performing poorly on actions with long intervals between start and end (such as high jump and long jump). Long-Term Temporal Convolution (LTC) [61] stacks more video frames to enhance long-term feature performance, while FOF [39] and FCF [40] stack multiple representation flow layers to capture longer temporal features. However, these methods involve large computation and risk losing relationships between long-interval frames, prompting research on capturing long-term behavioral features.

3.1 Global Uniform Sampling

TSN [35] from Section 2 uses sparse sampling to fix computational cost while simultaneously obtaining globally sampled frames for long-term feature extraction. Consequently, sparse sampling strategies have been widely adopted in AR model data preprocessing.

However, TSN simply averages prediction scores from sampled frames, unable to compensate for false label losses. Lan et al. [62] aggregated features into global features and trained mapping functions on the same training data to map global features to global labels. ActionVLAD [63] pooled and aggregated two-stream spatiotemporal features to achieve global feature integration. Diba et al. [64] fused sampled features for temporal linear encoding (TLE) to capture long-term dynamic processes. Wang et al. [65] proposed Temporal Difference Networks (TDN) based on TSN, designing channel attention enhancement methods based on different features to strengthen inter-segment motion change information.

3.2 Long Short-Term Memory Networks

Long Short-Term Memory (LSTM) [66] demonstrates remarkable effectiveness in representing language sequences with strong long-term feature capture capabilities. Since video shares similar temporal contextual relationships with language, Srivastava et al. [67] considered LSTM an effective approach to promote AR models in learning long sequence relationships.

As shown in Fig. 7, Ng et al. [68] first used 2D CNN to extract spatial features, then input them into LSTM for fusion to achieve temporal feature extraction. Building on this, Long-term Recurrent Convolutional Networks (LRCNs) [69] optimized end-to-end training. TS-LSTM [70] divided feature matrices into multiple segments, applying average or max pooling before sequentially inputting into LSTM layers. I3D-LSTM [71] attempted to combine 3D CNN and LSTM based on I3D. Li et al. [72] modified LSTM’ s weight dot product into convolution operations, demonstrating that Conv-LSTM better facilitates attention mechanisms than standard LSTM.

While LSTM enhances CNN’ s long-term representation capabilities to some extent, LSTM itself is difficult to train, and the strict sequential iteration significantly impacts training efficiency.

3.3 Transformer

CNN and LSTM can only capture long-term dependencies through repeated stacking, yet features gradually decay with increasing distance while incurring substantial computational overhead. In 2017, Google proposed Transformer [73] for natural language processing. Transformer' s multi-head self-attention mechanism can directly focus on global information between any sequences regardless of distance, offering strong computational parallelism. Wang et al. [74] proposed Non-Local Neural Networks (NLNN) based on self-attention, which can calculate relationships between any two spatiotemporal positions to rapidly capture long-term features. Neimark et al. [75] proposed VTN, a CNN+Transformer-based AR model that uses 2D CNN for feature extraction followed by Transformer structures to focus on long-term information. UniFormer [76] uses spatiotemporal self-attention to learn local and global label similarities in shallow and deep CNN layers respectively, addressing spatiotemporal redundancy and dependency for better balance between computation and accuracy.

ViViT [77] completely abandons CNN based on ViT [78], using pure Transformer for AR tasks. As shown in Fig. 8, ViViT constructs video as a set of spatiotemporal tokens with spatiotemporal position encoding as Transformer input for classification. MViT [79] creates multi-scale feature pyramids based on ViT, first modeling low-level visual information at high resolution and later modeling complex high-dimensional features at low resolution. Li et al. [80] improved MViT by decomposing relative position embeddings and residual pooling connections. Due to significant local redundancy and complex global dependencies between video frames, VidTr [81] and STAM-32 [82] proposed separable attention performing spatial and temporal attention separately, inspired by convolutional decomposition, to reduce encoding computational consumption.

Different temporal ordering of the same video frames may represent different actions—for example, walking might become running. However, traditional attention mechanisms lack directional information. Therefore, DirecFormer [83] transforms Transformer attention mechanisms into directed temporal and spatial attention based on cosine similarity to understand human actions in the correct order.

BEVT [84] conducted BERT pre-training for AR tasks, adopting a decoupled design that first performs masked image modeling on image data, then jointly performs masked image and video modeling on image and video data through weight sharing. BEVT simplifies AR Transformer learning while preserving spatial knowledge learned from images.

Due to Transformer' s versatility across data types, multimodal AR research based on Transformer has developed. Alfasly et al. [85] used BERT to build a Semantic Audio-Video Label Dictionary (SAVLD) that maps video labels to their most relevant audio labels, then jointly estimates audio-visual modality correlations with pre-trained audio multi-label models during training. Zellers et al. [86] designed a joint encoder (Transformer) applicable to all modalities and timesteps, providing video frames and sequence-level representations of words or audio to the joint encoder to predict data content.

Since 2021, Transformer-based AR models have continuously refreshed accuracy benchmarks across datasets, demonstrating excellent long-term feature capture capabilities. However, Transformer models lack inductive bias capabilities, lacking CNN' s translation invariance and locality, thus generalizing poorly to AR when data is insufficient.

4 Deep Model Evaluation

This section introduces public video datasets in Section 4.1 and analyzes recognition accuracy and runtime efficiency of various AR models based on UCF101, HMDB51, Kinetics400, and Something-Something datasets in Sections 4.2 and 4.3.

4.1 Video Datasets

Efficient and accurate model design constitutes core AR research, but video data is equally important. Video datasets should feature balanced categories, sufficient data, correct labeling, and task relevance. Duan et al. [87] trained video recognition models using web data to overcome format barriers. Zhang et al. [88] jointly trained Transformers across different video datasets to learn better action representations. Ryoo [89] learned to mine dataset labels from visual data, with TokenLearner trained on this dataset achieving excellent recognition results. These examples demonstrate the important role of datasets in AR models. Therefore, Table 1 introduces 16 public datasets for AR tasks.

HMDB51 [90] from public databases contains 6,849 trimmed videos across 51 action categories, including facial actions, general body movements, object interactions, and human interactions. UCF101 [91] collected from YouTube contains 13,320 trimmed videos divided into 25 groups, with categories including person-object interactions, single-person actions, human-human interactions, playing instruments, and sports.

Kinetics is currently the primary dataset for AR. The first-generation Kinetics-400 [92] from YouTube contains 400 action classes with 306,245 videos. The second-generation Kinetics-600 [93] expanded to 600 classes with 482,622 videos. The third-generation Kinetics-700 [94] extended to 700 classes with 650,317 videos. Kinetics-700-2020 [95] expanded the 700 classes to at least 700 clips per class.

Something-Something [96] (Sth-Sth) contains numerous action labels emphasizing actions themselves, comprising basic actions people perform on everyday objects, with 174 action classes. Version V1 has 108,499 videos, while V2 contains 220,847 videos with durations of 2-6 seconds.

Table 1. Comparison of AR Datasets

Dataset	Year	Action Classes	Clips	Description
Hollywood2	2009	12	3,669	Movie actions
HMDB51 [90]	2011	51	6,849	Body interaction actions
UCF101 [91]	2012	101	13,320	YouTube videos
Sports-1M [98]	2014	487	1,000,000	Sports videos
ActivityNet [99]	2015	200	19,994	Untrimmed videos
Kinetics [92]	2017	400	306,245	YouTube videos
Charades [100]	2016	157	9,848	Daily activities
Moments in Time [101]	2019	339	1,020,000	Verb action labels
Sth-Sth [96]	2017	174	108,499	Daily basic actions
TITAN [102]	2020	700	700	Vehicle/pedestrian actions
20BN-JESTER [103]	2019	27	148,092	Hand gestures
MMAt [104]	2019	37	36,000	Multimodal actions
RareAct [105]	2020	122	3,000	Unusual interactions
TinyVIRAT [106]	2021	26	12,829	Low-resolution actions
UAV-Human [107]	2021	155	67,428	UAV-view actions
Action Genome [108]	2021	101	10,000	Multi-view actions

Action Genome [108] is a multi-view action dataset with multiple modalities and viewpoints, supplemented by hierarchical activity and atomic action labels along with dense scene composition labels, featuring definitions for both high-level activities and low-level actions.

4.2 Accuracy Evaluation

This section disregards parameters, computation, training iterations, data pre-processing, and hardware/software configurations, focusing instead on single-label dataset accuracy to provide reference for efficiency evaluation in Section 4.3. Table 2 cites accuracy values from original papers for various methods on UCF101 and HMDB51, arranged chronologically and by technical principle, indicating optical flow usage, architecture, and pre-training.

Table 2. Comparison of AR Model Accuracy on UCF101 and HMDB51 Datasets

Year	Model	Backbone	Pre-training	UCF101	HMDB51
2014	Slow fusion CNN [10]	AlexNet	ImageNet	65.4	-
2015	Two-stream [13]	VGG-M-2048	ImageNet	88.0	59.4

Year	Model	Backbone	Pre-training	UCF101	HMDB51
2016	Fusion two-stream [14]	VGG-M-2048	ImageNet	92.5	65.4
2016	ST-ResNet+iDT [15]	ResNet50	ImageNet	94.6	70.3
2017	Two-Stream I3D [25]	BN-Inception	ImageNet+Kinetics	93.4	66.9
2017	TSN [35]	BN-Inception	ImageNet	94.2	69.0
2018	R3D [22]	ResNet-18	Sports-1M	87.2	-
2018	ResNeXt [23]	ResNet-101	Kinetics	95.1	72.2
2018	TDD+iDT [18]	VGG-M-2048	ImageNet	90.3	63.2
2019	TRN [26]	BN-Inception	ImageNet	-	-
2019	TSM [27]	ResNet-50	ImageNet	94.5	70.7
2019	STM [32]	ResNet-50	ImageNet	96.2	72.8
2020	TEA [31]	ResNet-50	ImageNet	96.9	73.3
2020	PAN [44]	ResNet-101	ImageNet+Kinetics	96.6	75.1
2020	TDN [65]	ResNet-50	ImageNet+Kinetics	97.4	76.3
2020	BQN [57]	ResNet-101	ImageNet+Kinetics	97.6	77.7
2021	UniFormer-B [76]	I3D+Transformer	ImageNet+Kinetics	98.6	94.5

Table 2 reveals several key insights: Early attempts with Slow fusion CNN [10] were unsatisfactory due to 2D CNN' s lack of temporal feature extraction capability. Optical flow-based Two-stream CNN [13] demonstrated excellent recognition performance, confirming the positive role of optical flow temporal features. 3D CNN models like C3D [21-23, 25, 61] proved the effectiveness of 3D convolutional kernels for spatiotemporal feature extraction. Temporal modules like TSM [27, 31, 32, 57] showed powerful temporal representation capabilities for CNNs. Conversely, LSTM provided limited accuracy gains for CNN-based

AR [68, 70, 71]. Finally, pure Transformer-based AR models [81, 82] achieved recognition accuracy comparable to other representative models like SMART [38], Two-Stream I3D [25], BQN [57], and I3D-LSTM [71].

Horizontal comparisons show that in the two-stream category, TDD+IDT [18] provides slight gains over Two-stream [11], while Fusion two-stream [14] and ST-ResNet [15] demonstrate that two-stream fusion and ResNet-based depth increase are suitable methods for improving two-stream accuracy. In the 3D and temporal module categories, models like R3D [22], ResNeXt [23], Two-Stream I3D [25], TSM [27], TEA [31], and BQN [57] all use ResNet or BN-Inception architectures to deepen convolutional layers for accuracy improvement. The sampling and spatiotemporal decomposition sections show that sparse sampling or horizontal model compression combined with vertical depth extension are effective choices for improving accuracy [35, 38, 46, 47, 64].

Regarding pre-training in Table 2, two-stream and temporal module methods based on 2D CNN like Two-stream [13] and TSN [35] use ImageNet pre-training. In the 3D category, I3D [25] introduced inflation, enabling ImageNet pre-training and, for the first time, Kinetics video dataset pre-training. I3D achieved excellent recognition results, with subsequent AR methods adopting similar pre-training strategies. This underscores the importance of datasets for AR accuracy improvement, as demonstrated by Omni [87]' s large-scale data joint statistical training showing excellent recognition performance.

Table 3. AR Model Accuracy Comparison on Kinetics-400 and Something-Something V1/V2 Datasets

Model	Backbone	Kinetics-400 Top-1	Kinetics-400 Top-5	Sth-Sth V1 Top-1	Sth-Sth V1 Top-5	Sth-Sth V2 Top-1	Sth-Sth V2 Top-5
TSN [35]	ResNet-50	69.1	88.7	19.7	-	-	-
I3D [25]	BN-Inception	71.1	89.3	41.6	72.2	-	-
S3D-G [47]	BN-Inception	74.7	93.4	48.2	78.1	-	-
CSN [50]	ResNet-101	76.7	92.2	46.6	76.1	-	-
SlowFast+NL [55]	ResNet-101	79.8	93.9	61.0	86.2	63.1	87.6
TSM [27]	ResNet-50	74.7	91.4	47.3	77.1	63.4	88.5
STM [32]	ResNet-50	78.3	93.5	50.2	80.1	64.2	89.8
TEA [31]	ResNet-50	76.1	92.5	48.9	79.1	62.1	87.9

Model	Backbone	Kinetics-400 Top-1	Kinetics-400 Top-5	Sth-Sth V1 Top-1	Sth-Sth V1 Top-5	Sth-Sth V2 Top-1	Sth-Sth V2 Top-5
PAN [44]	ResNet-101	77.7	93.2	52.4	81.9	65.4	90.1
TDN [65]	ResNet-50	79.4	94.4	54.4	83.2	67.0	91.2
BQN [57]	ResNet-101	78.8	93.9	53.7	82.5	66.8	91.0
ViViT [77]	ViT-B	82.9	94.5	60.9	87.3	71.2	92.8
MViT-L [80]	ViT-B	85.8	96.5	64.7	89.2	74.1	94.1
MaskFeat	ViT-B+MViT-L	87.0	97.4	-	-	-	-

Table 3 reveals: First, TSN [35] and I3D [25] show similar performance on Kinetics-400, but TSN lags significantly behind I3D on Something-Something, indicating TSN' s sparse sampling strategy loses substantial motion information. Second, with the same ResNet-50 architecture, TSN and I3D in the first section are not superior to TSM [27], STM [32], and TEA [31] in the second section on Kinetics-400, with the gap widening on Something-Something. Third, with the same ResNet-101 architecture, CSN [50] and SlowFast [55] have slight advantages over PAN [44], TDN [65], and BQN [57] on Kinetics-400, but fall behind again on Something-Something. This demonstrates that separately designed temporal modules on CNNs extract motion features more effectively than 3D convolutional kernels and optical flow. Finally, Transformer-based AR models emerging in 2021 have continuously topped accuracy leaderboards on both Kinetics-400 and Something-Something, directly surpassing years of CNN-based model development.

In summary, for scene-related AR tasks, focus on Transformer technology, temporal module design, horizontal model compression, residual connections, and large-scale dataset pre-training. For action-related tasks with weakened scene correlation, avoid overly sparse sampling strategies and concentrate on temporal modules or Transformer technology for model design.

4.3 Efficiency Evaluation

Section 4.2 compared models' temporal modeling capabilities based on recognition accuracy, but AR model evaluation must also emphasize efficiency for practical applications. Disregarding training iterations, data preprocessing, and hardware/software configurations, Table 4 cites pre-training status, architecture usage, input frames, parameters (representing GPU memory usage), GFLOPS (representing execution time depending on GPU computing power), and accuracy metrics from original papers for benchmark model efficiency evaluation.

Table 4. Efficiency Evaluation of AR Models on Kinetics-400

<table>
<tr><th></th><th></th><th></th><th>Parameters</th><th></th><th>Kinetics-400</th></tr>
<tr><th>Model</th><th>Backbone</th><th>Frames×View</th><th>(M)</th><th>GFLOPS×View</th><th>Top-1</th></tr>
<tr><td>TSN [35]</td><td>BN-Inception</td><td>25$×10×1|11.3|53×10×1|69.1||TSN[35]|ResNet−50|8×10×1|23.7|33×10×1|71.2||I3D[25]|BN−Inception|64×N/A×N/A|12.1|108×N/A|71.1||S3D−G[47]|BN−Inception|64×10×3|11.6|71.4×10×3|74.7||ARTNet[54]|ResNet−18|16×25×10|15.4|23.7×25×10|71.8||R(2+1)D[46]|ResNet−34|32×10×1|28.1|152×10×1|74.3||MF−Net[49]|ResNet−34|16×10×5|2.9|11.1×10×5|72.8||ip−CSN[50]|ResNet−101|32×10×3|15.1|83.0×10×3|76.7||ir−CSN[50]|ResNet−101|32×10×3|15.1|73.8×10×3|75.5||SlowFast[55]|ResNet−50|(8 + 32)×10×3|34.6|65.7×10×3|79.8||SlowFast[55]|ResNet−101|(8 + 64)×10×3|53.8|106×10×3|81.8||SlowFast+NL[55]|ResNet−101 + NL|(16 + 64)×10×3|60.1|234×10×3|82.1||MoViNet−A6[60]|MobileNet|50×1×1|2.9|386×1×1|81.5||ViViT−L[77]|ViT−B|250×1×1|310.0|1059×1×1|82.9||TokenLearner[89]|ViT−B|250×1×1|28.5|1989×1×1|85.4||MViTv1−S[79]|ViT−B|32×3×4|36.0|3992×3×4|80.3||MViTv1−B[79]|ViT−B|32×3×4|52.0|4076×3×4|82.1||MViT−S[80]|ViT−B|16×1×5|25.4|70.3×1×5|83.8||MViT−B[80]|ViT−B|32×1×5|36.6|170×1×5|85.8||MViT−L[80]|ViT−B|16×1×5|69.7|64×1×5|86.3||MaskFeat|ViT−B + MViT − L|32×1×5|169.0|225×1×$5</td><td>8.7</td><td></td><td></td></tr>
</table>

Table 4 reveals several insights: First, TSN [35] shows that ResNet achieves higher accuracy than BN-Inception with smaller FLOPS (ResNet with 8 frames outperforms Inception with 25 frames). Consequently, current AR models predominantly adopt ResNet as the base architecture, though ResNet demands more parameters. Second, in 3D CNN efficiency optimization, spatiotemporal decomposition in S3D-G [47] significantly reduces operations compared to I3D while slightly improving accuracy. Decomposition in ARTNet [54] and MF-Net [49] improves efficiency at the cost of accuracy. SlowFast [55] maintains considerable parameters and computation to preserve accuracy. Recent work like X3D [59] with progressive model expansion and MoViNet [60] with neural architecture search achieve excellent efficiency-accuracy balance. Third, temporal convolution models like TSM [27] have comparable size to decomposed 3D models while maintaining stable accuracy on Kinetics-400. TDN [65] greatly improves accuracy without increasing TSN' s size, proving the high efficiency and strong temporal modeling capability of inserted temporal modules. Finally, Transformer-based AR models [77, 79, 80, 89] break through 80% accuracy without increasing computation, surpassing most CNN-based models.

In summary, for online AR tasks, focus on convolutional decomposition, temporal module design, and Transformer approaches. However, Transformer-based AR models require large-scale data to be effective, making transfer learning a suitable solution for data-scarce applications.

5 Conclusion

This paper analyzed AR models from three perspectives—temporal feature extraction, efficiency optimization, and long-term feature capture—and compared the accuracy and efficiency performance of benchmark models after introducing public video datasets. Although current AR models perform well on public datasets, gaps remain for practical application. The following are reference insights for future AR development:

a) **Few-shot learning.** Training AR models requires massive labeled videos, but annotation costs are enormous, hindering practical application of supervised learning-based AR models. Additionally, different environmental backgrounds affect models trained in different environments. Therefore, few-shot learning involving cross-domain learning, transfer learning, and unsupervised learning can alleviate annotation costs while improving generalization, such as methods that fully aggregate spatiotemporal context using limited samples [109], convert image datasets to video model pre-training sources [110], or use unlabeled videos for pre-training [111].

b) **Video semantic understanding.** Current AR methods directly extract single-action features, but actual human behaviors are complex activities involving what is happening, when, who is performing the action, and where. Recognizing composite behaviors requires not only classification models but also video content semantic understanding. Generating basic semantics from video data to understand complex semantics effectively bridges the meaning gap between low-level and high-level behaviors.

c) **Fine-grained action recognition.** Fine-grained action recognition requires attention to subtle spatiotemporal differences, such as whether a person is walking slowly or quickly. Understanding detailed execution patterns and designing AR feature extractors that represent how actions occur to better distinguish fine-grained categories is a worthwhile research direction.

d) **Multimodal action recognition.** Humans perceive environments through multiple modalities including audio, tactile, visual, and skeleton information, which differ in form yet complement each other. Beyond visual information, AR can research how to leverage complementary multimodal data during training to learn better feature extractors.

e) **Multi-view action recognition.** Current AR primarily addresses single video views, but practical applications involve cameras placed at different orientations capturing information from various angles. This multi-view data presents challenges and opportunities. Reconstructing multi-view data into comprehensive 3D information to design feature extractors for 3D video data is a future direction worth exploring.

f) **Efficient model development.** Practical AR applications require fast processing, low computational cost, and small storage space. Previous efficiency optimization methods were mostly manual. Using neural architecture search to generate efficient and diverse architectures for integration represents the future direction for AR efficiency optimization.

References

[1] Ma Lijun. Research on action recognition algorithm based on 3d convolutional neural network [D]. Beijing: China University of Geosciences, 2018.

[2] He Ming, Yu Minggang, He Hongyue, et al. The future of artificial intelligence [M]. Beijing: Publishing Science Press, 2020: 31-32.

[3] Calum Chace. Artificial intelligence and the two singularities [M]. New York: Chapman and Hall/CRC Press, 2018.

[4] Liu Yong, Xie Ruoying, Feng Yang, et al. An overview of resident' s daily action recognition in smart home [J]. Computer Engineering and Applications, 2021, 57 (04): 35-42.

[5] Liu Yun, Xue Panpan, Li Hui, et al. A review of joint behavior recognition based on deep learning [J]. Journal of Electronics & Information Technology, 2021, 43 (06): 1789-1802.

[6] Zhang Xiaoping, Ji Jiahui, Wang Li, et al. Review of video based human abnormal behavior recognition and detection methods [J]. Control and Decision, 2022, 37 (01): 14-27.

[7] Fei Lishen, Liu Shaobo, Zhao Xuezhuan. A review of human behavior recognition [J]. Journal of Frontiers of Computer Science and Technology, 2022, 16 (02): 305-322.

[8] Wang H, Schmid C. Action recognition with improved trajectories [C]// Proc of IEEE International Conference on Computer Vision. Piscataway, NJ: IEEE Press, 2013: 3551-3558.

[9] LeCun Y, Bengio Y, Hinton G. Deep learning [J]. Nature, 2015, 521 (7553): 436-444.

[10] Karpathy A, Toderici G, Shetty S, et al. Large-scale video classification with convolutional neural networks [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2014: 1725-1732.

[11] Zach C, Pock T, Bischof H. A duality based approach for realtime TV-L1 optical flow [C]// Joint Pattern Recognition Symposium. Berlin: Springer, 2007: 214-223.

[12] He M, Zhu C, Huang Q, et al. A review of monocular visual odometry [J]. The Visual Computer, 2020, 36 (5): 1053-1065.

[13] Simonyan K, Zisserman A. Two-stream convolutional networks for action recognition in videos [C]. Neural Information Processing Systems. Canada: NIPS Proceedings, 2014, 568-576.

[14] Feichtenhofer C, Pinz A, Zisserman A. Convolutional two-stream network fusion for video action recognition [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2016: 1933-1941.

[15] Christoph R, Pinz F A. Spatiotemporal residual networks for video action recognition [C]. Neural Information Processing Systems. Spain: NIPS Proceedings, 2016: 3468-3476.

[16] Feichtenhofer C, Pinz A, Wildes R P. Spatiotemporal multiplier networks for video action recognition [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2017: 7445-7454.

[17] Wang L, Xiong Y, Wang Z, et al. Towards Good Practices for Very Deep Two-Stream ConvNets [J]. Computer Science, 2015, 8 (7): 1-5.

[18] Wang L, Qiao Y, Tang X. Action recognition with trajectory-pooled deep-convolutional descriptors [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2015: 4305-4314.

[19] Ding Xueqin, Zhu Yisheng, Zhu Haohua, et al. Action recognition based on spatiotemporal heterogeneous dual-flow convolutional networks [J]. Computer Applications and Software, 2022, 39 (03): 154-158.

[20] Ji S, Xu W, Yang M, et al. 3D convolutional neural networks for human action recognition [J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 2012, 35 (1): 221-231.

[21] Tran D, Bourdev L, Fergus R, et al. Learning spatiotemporal features with 3d convolutional networks [C]// Proc of IEEE International Conference on Computer Vision. Piscataway, NJ: IEEE Press, 2015: 4489-4497.

[22] Tran D, Ray J, Shou Z, et al. Convnet architecture search for spatiotemporal feature learning [J]. Computing Research Repository, 2017, 16 (8): 1-12.

[23] Hara K, Kataoka H, Satoh Y. Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet? [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2018: 6546-6555.

[24] Diba A, Fayyaz M, Sharma V, et al. Temporal 3d convnets: New architecture and transfer learning for video classification [EB/OL]. (2017) [2022-04-04]. https://arxiv.org/abs/1711.08200.

[25] Carreira J, Zisserman A. Quo vadis, action recognition? a new model and the kinetics dataset [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2017: 4724-4733.

[26] Zhou B, Andonian A, Oliva A, et al. Temporal relational reasoning in videos [C]// Lecture Notes in Computer Science. Berlin: Springer, 2018: 831-846.

[27] Lin J, Gan C, Han S. Tsm: Temporal shift module for efficient video understanding [C]// Proc of IEEE/CVF International Conference on Computer Vision. Piscataway, NJ: IEEE Press, 2019: 7082-7092.

[28] Shao H, Qian S, Liu Y. Temporal interlacing network [C]// Proc of AAAI Conference on Artificial Intelligence, Palo Alto, CA: AAAI Press, 2020, 34 (07): 11966-11973.

[29] Liu Z, Luo D, Wang Y, et al. Teinet: Towards an efficient architecture for video recognition [C]// Proc of AAAI Conference on Artificial Intelligence, Palo Alto, CA: AAAI Press, 2020, 34 (07): 11669-11676.

[30] Liu Z, Wang L, Wu W, et al. Tam: Temporal adaptive module for video recognition [C]// Proc of IEEE/CVF International Conference on Computer Vision. Piscataway, NJ: IEEE Press, 2021: 13688-13698.

[31] Li Y, Ji B, Shi X, et al. Tea: Temporal excitation and aggregation for action recognition [C]// Proc of IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2020: 906-915.

[32] Jiang B, Wang M M, Gan W, et al. Stm: Spatiotemporal and motion encoding for action recognition [C]// Proc of IEEE/CVF International Conference on Computer Vision. Piscataway, NJ: IEEE Press, 2019: 2000-2009.

[33] Luo Huilan, Chen Han. Temporal convolution attention network for action recognition [J/OL]. Computer Engineering and Applications, 2022. (2022-03-29) [2022-04-04]. http://kns.cnki.net/kcms/detail/11.2127.TP.20220328.1758.005.html.

[34] Wu Lijun, Li Binbin, Chen Zhicong, et al. Action recognition based on 3D multi-attention mechanism [J]. Journal of Fuzhou University (Natural Science),

2022, 50 (01): 47-53.

[35] Wang L, Xiong Y, Wang Z, et al. Temporal segment networks: Towards good practices for deep action recognition [C]// Lecture Notes in Computer Science. Berlin: Springer, 2016: 20-36.

[36] Zhu W, Hu J, Sun G, et al. A key volume mining deep framework for action recognition [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2016: 1991-1999.

[37] Liu X, Pintea S L, Nejadasl F K, et al. No frame left behind: Full Video Action Recognition [C]// Proc of IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2021: 1430-1439.

[38] Gowda S N, Rohrbach M, Sevilla-Lara L. SMART Frame Selection for Action Recognition [C]// Proc of AAAI Conference on Artificial Intelligence, Palo Alto, CA: AAAI Press, 2021, 35 (2): 1451-1459.

[39] Dosovitskiy A, Fischer P, Ilg E, et al. Flownet: Learning optical flow with convolutional networks [C]// Proc of IEEE International Conference on Computer Vision. Piscataway, NJ: IEEE Press, 2015: 2758-2766.

[40] Ilg E, Mayer N, Saikia T, et al. Flownet 2.0: Evolution of optical flow estimation with deep networks [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2017: 1647-1655.

[41] Piergiovanni A J, Ryoo M S. Representation flow for action recognition [C]// Proc of IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2019: 9937-9945.

[42] Zhu Y, Lan Z, Newsam S, et al. Hidden two-stream convolutional networks for action recognition [C]// Lecture Notes in Computer Science. Berlin: Springer, 2018: 363-378.

[43] Crasto N, Weinzaepfel P, Alahari K, et al. Mars: Motion-augmented rgb stream for action recognition [C]// Proc of IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2019: 7874-7883.

[44] Zhang C, Zou Y, Chen G, et al. Pan: Towards fast action recognition via learning persistence of appearance [EB/OL]. (2020) [2022-04-04]. https://arxiv.org/abs/2008.03462.

[45] Qiu Z, Yao T, Mei T. Learning spatio-temporal representation with pseudo-3d residual networks [C]// Proc of IEEE International Conference on Computer Vision. Piscataway, NJ: IEEE Press, 2017: 5533-5541.

[46] Tran D, Wang H, Torresani L, et al. A closer look at spatiotemporal convolutions for action recognition [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2018: 6450-6459.

[47] Xie S, Sun C, Huang J, et al. Rethinking spatiotemporal feature learning:

Speed-accuracy trade-offs in video classification [C]// Lecture Notes in Computer Science. Berlin: Springer, 2018: 318-335.

[48] Sudhakaran S, Escalera S, Lanz O. Gate-shift networks for video action recognition [C]// Proc of IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2020: 1099-1108.

[49] Chen Y, Kalantidis Y, Li J, et al. Multi-fiber networks for video recognition [C]// Lecture Notes in Computer Science. Berlin: Springer, 2018: 364-380.

[50] Tran D, Wang H, Torresani L, et al. Video classification with channel-separated convolutional networks [C]// Proc of IEEE/CVF International Conference on Computer Vision. Piscataway, NJ: IEEE Press, 2019: 5551-5560.

[51] Luo C, Yuille A L. Grouped spatial-temporal aggregation for efficient action recognition [C]// Proc of IEEE/CVF International Conference on Computer Vision. Piscataway, NJ: IEEE Press, 2019: 5511-5520.

[52] Zhou Y, Sun X, Zha Z J, et al. Mict: Mixed 3d/2d convolutional tube for human action recognition [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2018: 449-458.

[53] Zolfaghari M, Singh K, Brox T. Eco: Efficient convolutional network for online video understanding [C]// Lecture Notes in Computer Science. Berlin: Springer, 2018: 713-730.

[54] Wang L, Li W, Li W, et al. Appearance-and-relation networks for video classification [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2018: 1430-1439.

[55] Feichtenhofer C, Fan H, Malik J, et al. Slowfast networks for video recognition [C]// Proc of IEEE/CVF International Conference on Computer Vision. Piscataway, NJ: IEEE Press, 2019: 6201-6210.

[56] Yang C, Xu Y, Shi J, et al. Temporal pyramid network for action recognition [C]// Proc of IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2020: 588-597.

[57] Huang G, Bors A G. Busy-Quiet Video Disentangling for Video Classification [C]// Proc of IEEE/CVF Winter Conference on Applications of Computer Vision. Piscataway, NJ: IEEE Press, 2022: 1647-1655.

[58] Liu Zhao, Yang Fan, Si Yazhong. Research on time-domain unfilled network video behavior recognition Algorithm [J/OL]. Computer Engineering and Applications, 2022. (2022-01-16) [2022-04-04]. http://kns.cnki.net/kcms/detail/11.2127.TP.20220106.1220.002.html.

[59] Feichtenhofer C. X3d: Expanding architectures for efficient video recognition [C]// Proc of IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2020: 200-210.

[60] Kondratyuk D, Yuan L, Li Y, et al. Movinets: Mobile video networks for efficient video recognition [C]// Proc of IEEE/CVF Conference on Computer

Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2021: 16015-16025.

[61] Varol G, Laptev I, Schmid C. Long-term temporal convolutions for action recognition [J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 2017, 40 (6): 1510-1517.

[62] Lan Z, Zhu Y, Hauptmann A G, et al. Deep local video feature for action recognition [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition Workshops. Piscataway, NJ: IEEE Press, 2017: 1-8.

[63] Girdhar R, Ramanan D, Gupta A, et al. Actionvlad: Learning spatio-temporal aggregation for action classification [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2017: 3165-3174.

[64] Diba A, Sharma V, Van Gool L. Deep temporal linear encoding networks [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2017: 1541-1550.

[65] Wang L, Tong Z, Ji B, et al. TDN: Temporal difference networks for efficient action recognition [C]// Proc of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2021: 1895-1904.

[66] Hochreiter S, Schmidhuber J. Long short-term memory [J]. Neural computation, 1997, 9 (8): 1735-1780.

[67] Srivastava N, Mansimov E, Salakhudinov R. Unsupervised learning of video representations using lstms [C]// Proc of the 32th International Conference on Machine Learning. Cambridge MA: JMLR, 2015: 843-852.

[68] Yue-Hei Ng J, Hausknecht M, Vijayanarasimhan S, et al. Beyond short snippets: Deep networks for video classification [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2015: 4694-4702.

[69] Donahue J, Anne Hendricks L, Guadarrama S, et al. Long-term recurrent convolutional networks for visual recognition and description [J]// IEEE Trans on Pattern Analysis and Machine Intelligence, 2017, 39 (4): 677-691.

[70] Ma C Y, Chen M H, Kira Z, et al. TS-LSTM and temporal-inception: Exploiting spatiotemporal dynamics for activity recognition [J]. Signal Processing: Image Communication, 2019, 71: 76-87.

[71] Wang X, Miao Z, Zhang R, et al. I3D-LSTM: A new model for human action recognition [C/OL]// Proc of IOP Conference Series: Materials Science and Engineering. S. l.: IOP, 2019, 569 (3): 032035.

[72] Li Z, Gavrilyuk K, Gavves E, et al. Videolstm convolves, attends and flows for action recognition [J]. Computer Vision and Image Understanding, 2018, 166: 41-50.

[73] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need [C]// Neural Information Processing Systems. USA: NIPS Proceedings, 2017: 5998-6008.

[74] Wang X, Girshick R, Gupta A, et al. Non-local neural networks [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2018: 7794-7803.

[75] Neimark D, Bar O, Zohar M, et al. Video transformer network [C]// Proc of IEEE/CVF International Conference on Computer Vision Workshops. Piscataway, NJ: IEEE Press, 2021: 3156-3165.

[76] Li K, Wang Y, Gao P, et al. Uniformer: Unified Transformer for Efficient Spatiotemporal Representation Learning [EB/OL]. (2022) [2022-04-04]. https://arxiv.org/abs/2201.04676.

[77] Arnab A, Dehghani M, Heigold G, et al. Vivit: A video vision transformer [C]// Proc of IEEE/CVF International Conference on Computer Vision. Piscataway, NJ: IEEE Press, 2021: 6816-6826.

[78] Dosovitskiy A, Beyer L, Kolesnikov A, et al. An image is worth 16x16 words: Transformers for image recognition at scale [EB/OL]. (2020) [2022-04-04]. https://arxiv.org/abs/2010.11929.

[79] Fan H, Xiong B, Mangalam K, et al. Multiscale vision transformers [C]// Proc of IEEE/CVF International Conference on Computer Vision. Piscataway, NJ: IEEE Press, 2021: 6804-6815.

[80] Li Y, Wu C Y, Fan H, et al. Improved multiscale vision transformers for classification and detection [EB/OL]. (2021) [2022-04-04]. https://arxiv.org/abs/2112.01526.

[81] Zhang Y, Li X, Liu C, et al. Vidtr: Video transformer without convolutions [C]// Proc of IEEE/CVF International Conference on Computer Vision. Piscataway, NJ: IEEE Press, 2021: 13557-13567.

[82] Sharir G, Noy A, Zelnik-Manor L. An Image is Worth 16x16 Words, What is a Video Worth? [EB/OL]. (2021) [2022-04-04]. https://arxiv.org/abs/2103.13915.

[83] Truong T D, Bui Q H, Duong C N, et al. DirecFormer: A Directed Attention in Transformer Approach to Robust Action Recognition [EB/OL]. (2022) [2022-04-04]. https://arxiv.org/abs/2203.10233.

[84] Wang R, Chen D, Wu Z, et al. Bevt: Bert pretraining of video transformers [EB/OL]. (2021) [2022-04-04]. https://arxiv.org/abs/2112.01529.

[85] Alfasly S, Lu J, Xu C, et al. Learnable Irrelevant Modality Dropout for Multimodal Action Recognition on Modality-Specific Annotated Videos [EB/OL]. (2022) [2022-04-04]. https://arxiv.org/abs/2203.03014.

[86] Zellers R, Lu J, Lu X, et al. MERLOT Reserve: Neural Script Knowledge through Vision and Language and Sound [EB/OL]. (2022) [2022-04-04].

https://arxiv.org/abs/2201.02639.

[87] Duan H, Zhao Y, Xiong Y, et al. Omni-sourced webly-supervised learning for video recognition [C]// Lecture Notes in Computer Science. Berlin: Springer, 2020: 670-688.

[88] Zhang B, Yu J, Fifty C, et al. Co-training Transformer with Videos and Images Improves Action Recognition [EB/OL]. (2021) [2022-04-04]. https://arxiv.org/abs/2112.07175.

[89] Ryoo M S, Piergiovanni A J, Arnab A, et al. TokenLearner: What Can Learned Tokens Do for Images and Videos? [EB/OL]. (2021) [2022-04-04]. https://arxiv.org/abs/2106.11297.

[90] Jhuang H, Garrote H, Poggio E, et al. HMDB: A large video database for human motion recognition [C]// Proc of IEEE International Conference on Computer Vision. Piscataway, NJ: IEEE Press, 2011: 2556-2563.

[91] Soomro K, Zamir A R, Shah M. A dataset of 101 human action classes from videos in the wild [J]. Center for Research in Computer Vision, 2012, 2 (11).

[92] Kay W, Carreira J, Simonyan K, et al. The kinetics human action video dataset [EB/OL]. (2017) [2022-04-04]. https://arxiv.org/abs/1705.06950.

[93] Carreira J, Noland E, Banki-Horvath A, et al. A short note about kinetics-600 [EB/OL]. (2018) [2022-04-04]. https://arxiv.org/abs/1808.01340.

[94] Carreira J, Noland E, Hillier C, et al. A short note on the kinetics-700 human action dataset [EB/OL]. (2019) [2022-04-04]. https://arxiv.org/abs/1907.06987.

[95] Smaira L, Carreira J, Noland E, et al. A short note on the kinetics-700-2020 human action dataset [EB/OL]. (2020) [2022-04-04]. https://arxiv.org/abs/2010.10864.

[96] Goyal R, Ebrahimi Kahou S, Michalski V, et al. The "something something" video database for learning and evaluating visual common sense [C]// Proc of IEEE International Conference on Computer Vision. Piscataway, NJ: IEEE Press, 2017: 5843-5851.

[97] Marszalek M, Laptev I, Schmid C. Actions in context [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2009: 2929-2936.

[98] Karpathy A, Toderici G, Shetty S, et al. Large-scale video classification with convolutional neural networks [C]// Proc of the Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2014: 1725-1732.

[99] Caba Heilbron F, Escorcia V, Ghanem B, et al. Activitynet: A large-scale video benchmark for human activity understanding [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2015: 961-970.

[100] Sigurdsson G A, Varol G, Wang X, et al. Hollywood in homes: Crowdsourcing data collection for activity understanding [C]// Lecture Notes in Computer

Science. Berlin: Springer, 2016: 510-526.

[101] Monfort M, Andonian A, Zhou B, et al. Moments in time dataset: one million videos for event understanding [J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 2019, 42 (2): 502-508.

[102] Kong Q, Wu Z, Deng Z, et al. Mmact: A large-scale dataset for cross modal human action understanding [C]// Proc of IEEE/CVF International Conference on Computer Vision. Piscataway, NJ: IEEE Press, 2019: 8657-8666.

[103] Materzynska J, Berger G, Bax I, et al. The jester dataset: A large-scale video dataset of human gestures [C]// Proc of IEEE/CVF International Conference on Computer Vision Workshops. Piscataway, NJ: IEEE Press, 2019: 2874-2882.

[104] Malla S, Dariush B, Choi C. Titan: Future forecast using action priors [C]// Proc of IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2020: 11183-11193.

[105] Miech A, Alayrac J B, Laptev I, et al. RareAct: A video dataset of unusual interactions [EB/OL]. (2020) [2022-04-04]. https://arxiv.org/abs/2008.01018.

[106] Demir U, Rawat Y S, Shah M. TinyVIRAT: low-resolution video action recognition [C]// Proc of the 25th International Conference on Pattern Recognition. Piscataway, NJ: IEEE Press, 2021: 7387-7394.

[107] Li T, Liu J, Zhang W, et al. UAV-Human: A Large Benchmark for Human Behavior Understanding with Unmanned Aerial Vehicles [C]// Proc of IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2021: 16261-16270.

[108] Rai N, Chen H, Ji J, et al. Home action genome: Cooperative compositional action understanding [C]// Proc of IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2021: 11179-11188.

[109] Thatipelli A, Narayan S, Khan S, et al. Spatio-temporal Relation Modeling for Few-shot Action Recognition [EB/OL]. (2021) [2022-04-04]. https://arxiv.org/abs/2112.05132.

[110] Huang Z, Zhang S, Jiang J, et al. Self-supervised motion learning from static images [C]// Proc of IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2021: 1276-1285.

[111] Wei C, Fan H, Xie S, et al. Masked Feature Prediction for Self-Supervised Visual Pre-Training [EB/OL]. (2021) [2022-04-04]. https://arxiv.org/abs/2112.09133.

Note: Figure translations are in progress. See original paper for figures.

Source: ChinaXiv —Machine translation. Verify with original.